2014

Towards a Computational Analysis of Probabilistic Argumentation Frameworks

Pierpaolo Dondio
Technological University Dublin, pierpaolo.dondio@tudublin.ie

Recommended Citation

Dondio, P. (2014) Towards a Computational Analysis of Probabilistic Argumentation Frameworks. *Cybernetics and systems* [to appear in 2014] doi:10.1080/01969722.2014.894854

Funder: School of Computing

Towards a Computational Analysis of Probabilistic Argumentation Frameworks

PIERPAOLO DONDIO[1]
[1] *School of Computing,*
Dublin Institute of Technology, Kevin Street 2, Dublin, Ireland

In this paper we analyze probabilistic argumentation frameworks (PAFs), defined as an extension of Dung abstract argumentation frameworks in which each argument n is asserted with a probability p_n. The debate around PAFs has so far centered on their theoretical definition and basic properties. This work contributes to their computational analysis by proposing a first recursive algorithm to compute the probability of acceptance of each argument under grounded and preferred semantics, and by studying the behavior of PAFs with respect to reinstatement, cycles and changes in argument structure. The computational tools proposed may provide strategic information for agents selecting the next step in an open argumentation process and they represent a contribution in the debate about gradualism in abstract argumentation.

KEYWORDS: Argumentation Theory, Probabilistic Reasoning, Abstract Argumentation, Grounded and Preferred Semantics

INTRODUCTION

An abstract argumentation framework is a direct graph where nodes represent arguments and arrows represent the attack relation. Abstract argumentation frameworks [13, 9] were introduced by Dung [2] to analyze properties of defeasible arguments, i.e. arguments whose validity can be disputed by other conflicting arguments.

Various semantics have been defined to identify the set of acceptable arguments. In this work we deal with grounded and preferred semantics and we follow the labeling approach proposed by Caminada [7], where a semantics assigns to each argument a label *in*, *out* or *undec*, meaning that the argument is considered consistently acceptable, non-acceptable or undecided (i.e. one abstains from an opinion).

In Dung's original work, arguments are treated as abstract entities that are either fully asserted or not asserted at all, and there are no degrees related to either arguments or relations of attacks. Abstract argumentation is often too strict and coarse to support a decision making process. The situation is described by Dunne et al. [11], who notice how the solution provided by abstract argumentation "*is often an empty set or several sets with nothing to distinguish between them*". Abstract argumentation has proven to be efficient in keeping the logical consistency of conflicting evidence, but there are limited extensions that can be practically deployed to handle gradualism. Some approaches have tried to marry probability calculus and argumentation semantics, defining probabilistic argumentation frameworks. In a $PAFs$ an

argumentation semantics is used to identify under which conditions a set of arguments are acceptable, while probability calculus quantifies how likely those conditions are.

The study of *PAFs* is still at an embryonic stage. Debate has centered on their correct theoretical definition and some basic properties derived from abstract argumentation. There is no computational algorithm proposed beside the brute force approach, and no study over their behavior w.r.t. to reinstatement or sensitivity to changes in the argumentation structure.

Taking stock of previous research in the area, we first modify some formal definitions of PAF concepts. However, these definitions are anchored in previous works and do not represent the major contribution of the paper. Our key contribution is represented by a set of new computational tools developed for analyzing PAF: we describe the first recursive algorithm to compute the probability of acceptance of each argument and we study the behavior of PAF w.r.t. to reinstatement, cycles and changes in the arguments structure. Our work represents a contribution to the introduction of gradualism in argumentation.

The paper is organized as follows: in the first two sections we recall the pre-requisites of abstract argumentation and *PAFs*; we describe the very first algorithm to compute the acceptance probabilities of the arguments. We then describe the behavior of PAF w.r.t. to reinstatement and cycles, we analyze the behavior of PAF in relation to changes and we propose an application of PAF, before discussing related works.

BACKGROUND DEFINITIONS

Definition 1. *(Abstract Argumentation Framework) Let U be the universe of all possible arguments. An argumentation framework is a pair (Ar, R) where Ar is a finite subset of U and $R \subseteq Ar \times Ar$ is the attack relation.*

Let us consider $\mathrm{AF} = (\mathrm{Ar}, \mathrm{R})$ and $\mathrm{Args} \subseteq \mathrm{Ar}$.

Definition 2. *(conflict-free). $Args$ is conflict-free iff $\nexists a, b \in Args \mid a\ R\ b$*

Definition 3. *(defense). $Args$ defends an argument $a \subseteq Ar$ iff $\forall b \in Ar$ such that $b\ R\ a, \exists\ c \in Args$ such that $c\ R\ b$. The set of arguments defended by $Args$ is denoted $F(Args)$.*

Definition 4. *(indirect attack/defense) Let $a, b \in Ar$ and the graph G defined by (Ar, R). Then (1) a indirectly attacks b if there is an odd-length path from a to b in the attack graph G and (2) a indirectly defends b if there is an even-length path from a to b in G.*

Two arguments a and b are rebuttals *iff* $R(a, b) \wedge R(b, a)$.

Labeling. A semantics identifies a set of arguments that can survive the conflicts encoded by the attack relation R. In this work we follow the labeling approach of Caminada et al. [7], where a semantics assigns to each argument a label *in*, *out* or *undec*.

Definition 5. *(labeling/conflict free). Let $AF = (Ar, R)$ be an argumentation framework. A labeling is a total function $L: Ar \rightarrow \{in, out, undec\}$. We write in(L) for $\{a \in Ar | L(a) = in\}$, out(L) for $\{a \in Ar | L(a) = out\}$, and undec(L) for $\{a \in Ar | L(a) = undec\}$. A labeling is conflict-free if no in-labeled argument attacks an in-labeled argument.*

Definition 6. *(complete labeling, from Definition 5 in [7]). Let (Ar, R) be an argumentation framework. A complete labeling is a labeling that for every $a \in Ar$ holds that:1. if a is labeled in then all attackers of a are labeled out; 2. if all attackers of a are labeled out*

then a is labeled in; 3. if a is labeled out then a has an attacker labeled in; 4. if a has an attacker labeled in then a is labeled out

Theorem 1. *(from [7]) Let L be a labeling of argumentation framework (Ar, R). It holds that L is a complete labeling iff for each argument $a \in Ar$ it holds that: 1. if a is labeled in then all its attackers are labeled out; 2. if a is labeled out then it has at least one attacker that is labeled in; 3. if a is labeled $undec$ then it has at least one attacker that is labeled $undec$ and it does not have an attacker that is labeled in.*

Theorem 2. *(from theorem 6 and 7 in [7]) Let $AF = (Ar, R)$ be an argumentation framework. L is the grounded labeling iff L is a complete labeling where undec(L) is maximal (w.r.t. set inclusion) among all complete labelings of AF. L is the preferred labeling iff L is a complete labeling where $in(L)$, $out(L)$ are maximal.*

In figure 1 two argumentation graphs are depicted. Grounded semantics assigns the status of *undec* to all the arguments of (A), since it represents the complete labeling with the maximal undec-set, while in (B), according to theorem 1, there is only one complete labeling (that is thus grounded and preferred), where argument a is *in* (no attackers), b is *out* and c results in *in*. Note how a reinstates c. Regarding (A), there are two complete labelings where in(L) is maximal w.r.t. to set inclusion: one with $in(L_1) = \{b\}$, $out(L_1) = \{a, c\}$, $undec(L_1) = \emptyset$ and the other with $in(L_2) = \{a, c\}$ and $out(L_2) = \{b\}$ and $undec(L_2) = \emptyset$.

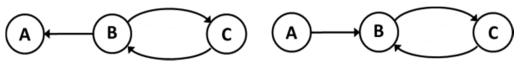


Figure 1. Two Argumentation Graphs (A) and (B)

PROBABILISTIC ARGUMENTATION FRAMEWORKS

In this section we present earlier work in probabilistic argumentation frameworks that we progress. We start from the concept of PAF and its meaning. In the first work by Li [4], a probability measure is attached to each argument and attack relation of an abstract argumentation framework. Li et al. define these probabilities as the *"likelihood of existence of an argument or attack relation"* on the argumentation graph [4]. In [6], Hunter progresses the conceptual notion of PAF. He admits that "*what is meant by the probability of an argument holding is an open question"*. He proposes a justification prospective similar to [4], where the probability indicates the degree to which the argument belongs (or is believed to belong) to the graph. Yet he also proposes an alternative view, referred to as the *premises perspective* on argument probability of being true. In this approach, the probability of each argument is based on the degree to which its premises are true, or are believed to be true. Our stance is closer to Hunter's second view. Since an argument's premises are affected by probabilistic uncertainty, we are left with an argument whose claim is affected by the same uncertainty, meaning that the claim holds with likelihood x, and does not hold with likelihood $1 - x$.

We report the definition proposed by Li [4] used as baseline reference for this paper:

A probabilistic argumentation framework PrAF is a tuple (Ar, P_A, D, P_D) where (Ar, D) is an abstract argumentation framework, $P_A : Ar \rightarrow (0 : 1]$ and $P_D : D \rightarrow (0 : 1]$.

Key elements of Li et al.'s definition are the use of a probability for both arguments and attacks and the assumption of argument and attack independence (hence P_A and P_D are scalar

numbers). Central to this is the way argument probability of acceptance is computed. The probabilistic nature of arguments, common to Li et al., Hunter and our research, implies that given an argumentation framework of n elements, 2^n different scenarios are possible, each of them obtained by assuming each argument or attack relation to exist or not. Li et al. call these scenarios *induced argumentation frameworks*, each corresponding to a subgraph of the starting argumentation framework. Each induced argumentation framework has a probability *of existing* attached to it, computed by applying the product rule using P_A and P_D, and each induced framework behaves as an abstract Dung-style framework.

Thus, given a semantics (although only grounded semantics is analysed in [4]), Li et al. define the probability of acceptance of an extension as the sum of the probabilities of all the induced frameworks where the chosen semantics produce that extension. This computation, that requires computing the chosen semantic in all the subgraphs of the original argumentation framework, is referred to by Hunter as the *constellation approach*. Hunter [6] extends some of Li et al.'s definitions and investigates the situations where arguments might not be independent and the probability P is given as a joint probability distribution.

Our Definition and its Differences from Previous Works

Definition 7. *(PAF). A probabilistic argumentation framework PAF is a couple* (A, P) *where* $A = (Ar, R)$ *is an abstract argumentation framework with a finite set of arguments* Ar *and an attack relation* R *on* $Ar \times Ar$*; and* P *is a joint probability distribution over* Ar.

Our contribution to the formal definition of PAF is minor, and our definition is an extension of the previous work of Li et al. and Hunter. However, in the next section we will introduce new definitions of argument acceptability used in our computational analysis and based on the above definition, and it is thus important to make these modifications clear and explicit. Referring to Li et al.'s definition as a baseline, our PAF differs in the following respects: probabilities are only attached to arguments and induced frameworks are only identified by subsets of nodes, the probability P is a joint probability rather than a scalar function. Moreover, as described in the next section, we define acceptability at argument level rather than at extension level, we also introduce the probability of an argument to be labeled *out* or *undec*, we extend the definitions of the probability of argument acceptance by adding the credulous and skeptical acceptance of preferred semantics.

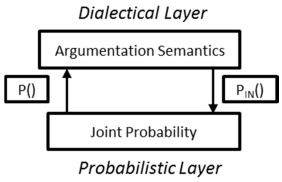
We end this section by clarifying some concepts of our PAF definition that are not discussed by Li et al. and Hunter, but that are useful to better understanding our computational analysis. In the definition of a *PAF*, given a generic argument a, $P(a)$ is the probability that a holds *on its own,* in isolation, before the dialectical process starts. It is the likelihood that the probabilistic premises of the argument are true, and thus the argument claim can be used in the argumentation process. Our aim is to compute the probabilities $P_{IN}(a), P_{OUT}(a), P_U(a)$ that a generic argument a will be labeled *in, out* or *undec* under the chosen semantics. Algorithm 1 proposes a brute force approach to computing P_{IN} (P_{OUT}, P_U are analogous).

The difference between $P(a)$ and $P_{IN}(a)$ is crucial. If $P(a)$ is the probability of argument a's claim to hold in isolation, before the argumentation process combines arguments, $P_{IN}(a)$ is the probability of a being labeled *in* by the chosen semantics. $P_{IN}(a)$ entails the effect of the argumentation process on a, i.e. the fact that a's conclusion might be invalidated by other arguments. Argument a could have a high probability of holding in isolation, but be

completely invalidated in argumentation. It may be that a: *Joe got full marks in his math test, so he is good at math*, but it might be also known that b: *Joe copied the test*. Thus $P(a) = P(b) = 1$, but since b attacks a, then $P_{IN}(a) = 0$ (the conclusion does not hold anymore) .

Algorithm 1 - Brute force approach for computing P_{IN}

```
for each sub-graph G of (Ar,R)
  use P to compute the probability P(G) of G
    for each argument a in G
      assign a label l(a) to a in G using the
      chosen semantics
      if l(a) = in add P(G) to P_IN(a)
```



Formalizing Scenarios and their Probabilities

Given an argumentation framework $AF = (Ar, R)$ with $|Ar| = n$, and the graph G identified by Ar and R, we consider the set H of all the subgraphs of G. We define specific sets of subgraphs, i.e. elements of 2^H. Given $a \in Ar$, we define:

$$A = \{g \in H \mid a \in g\} \quad ; \quad \bar{A} = \{g \in H \mid a \notin g\} \qquad (1)$$

that are respectively the set of subgraphs where argument a is present and the set of subgraphs where a is not present (note how we use $\bar{A}$ for the complementary set A^C).

We define a *scenario* as the argumentation framework identified by the subgraph g and the restriction of R to g. A scenario models the situation in which some arguments are assumed to hold and are present in the argumentation process and some arguments are assumed not to hold and are discarded from the dialectical process.

In general, we can express a set of subgraphs (and corresponding scenarios) combining some of the sets $A_1, .., A_n, \overline{A_1}, .., \overline{A_n}$. with the connectives {∪,∩}. We write AB to denote $A \cap B$ and $A + B$ for $A \cup B$. For instance, in figure 1 the single subgraph/scenario with only b and c present is denoted with $\bar{A}BC$, while the expression AB denotes a set of two subgraphs/scenarios where arguments a and b are present and c can be either present or not.

We call *clause* φ a finite intersection (or conjunction) of sets A_i , $\overline{A_i}$. We consider expressions of sets of scenarios in their *disjunctive normal form*, i.e. as a finite disjunction of clauses $\varphi_1 + \varphi_2 + .. + \varphi_m$.

It is possible to compute the probability of each subgraph/scenario starting from the probability P. If $Ar = \{a_1, .., a_n\}$, a single scenario/subgraph is a clause of length n modeling a generic situation in which a_j probabilistic arguments are assumed to hold while the other $n - j$ are not. The probability $P_s()$ of this generic scenario s is the joint probability:

$$P_s(s) = P(a_1 \wedge a_2 \wedge ... \wedge a_j \wedge \overline{a_{j+1}} \wedge ... \wedge \overline{a_n}) \qquad (2)$$

The probability of a set of scenarios $P_{ss}()$ is the sum of the probabilities of each scenario in the set. Thus, regarding the set of scenarios A, by marginalization on argument a:

$$P_{ss}(A) = \sum_{s \in A} P_s(s) = P(a) \text{ and } P_{ss}(\bar{A}) = 1 - P(a) \qquad (3)$$

Since every set of scenarios can be expressed by the conjunction of expressions containing only the sets A_i , $\overline{A_i}$, using the above equation the probability of any set of subgraphs can be expressed using P. For instance $P_{SS}(AB) = P(a \wedge b)$, $P_{SS}(A + \bar{B}) = P(a \vee \neg b)$.

Labeling Scenarios and Acceptance

Given a scenario $s \in S$ (S being the set of all the scenarios), the labeling of s follows the rules of the chosen semantics. We define a scenario labeling $\mathcal{L}$ as a total function over the Cartesian product of arguments in Ar and scenarios in S, thus $\mathcal{L}: \mathrm{Ar} \times \mathrm{S} \rightarrow \{in, out, undec\}$. When labeling a scenario, we follow this choice: an argument a is labeled *out* in all the scenarios where a does not hold (i.e. it is *out* because it is assumed not to hold *on its own*) or when it holds but it is labeled *out* by the semantics, representing the effect on a of other arguments.

Regarding grounded semantics there is only one labeling per scenario s, that we call $\mathcal{L}^g(s)$. In the case of preferred labeling there could be more than one valid labeling per scenario. Each preferred labeling for scenario s is referred to as $\mathcal{L}_i^{pr}(s)$ and the set of the preferred labelings of a scenario s as $\boldsymbol{\ell}^{pr}(s) = \{\mathcal{L}_1^{pr}(s), .., \mathcal{L}_n^{pr}(s)\}$. We call $in(\mathcal{L}^x(s))$, $out(\mathcal{L}^x(s))$, $undec(\mathcal{L}^x(s))$ the sets of argument labeled *in*, *out*, *undec* in $\mathcal{L}^x(s)$, with x denoting the semantics used (either g or pr). In order to study how an argument behaves across scenarios in S, we define the following set of scenarios. For grounded semantics:

$$A_{IN}^g = \{s \in S: a \in in(\mathcal{L}^g, s)\}\ ;\ A_{OUT}^g = \{s \in S: a \in out(\mathcal{L}^g, s)\}$$
$$A_U^g = \{s \in S: a \in undec(\mathcal{L}^g, s)\}$$

which represent all the scenarios where argument a is labeled *in*, *out* or *undec*. Regarding preferred semantics, since there could be more than one labeling for each scenario s, we define two extreme sets corresponding to skeptical and credulous attitudes. The credulous set is identified by requiring argument a to be labeled *in* at least in one of the valid preferred labelings for s. Hence we define:

$$A_{IN}^{pr+} = \left\{s \in S: \left(\exists\ \mathcal{L}^{pr}(s) \in \boldsymbol{\ell}^{pr}(s) :\ a \in in(\mathcal{L}^{pr}, s)\right)\right\}$$
$$A_{OUT}^{pr+} = \left\{s \in S: \left(\exists\ \mathcal{L}^{pr}(s) \in \boldsymbol{\ell}^{pr}(s) :\ a \in out(\mathcal{L}^{pr}, s)\right)\right\}$$
$$A_U^{pr+} = \left\{s \in S: \left(\exists\ \mathcal{L}^{pr}(s) \in \boldsymbol{\ell}^{pr}(s) :\ a \in undec(\mathcal{L}^{pr}, s)\right)\right\}$$

While the skeptical sets are:

$$A_{IN}^{pr-} = A_{IN}^{pr} \setminus \left(A_{OUT}^{pr} \cup A_U^{pr}\right)\ ;\ A_{OUT}^{pr-} = A_{OUT}^{pr} \setminus (A_{IN}^{pr} \cup A_U^{pr})\ ;\ A_U^{pr-} = A_U^{pr} \setminus (A_{OUT}^{pr} \cup A_{IN}^{pr}) \quad (4)$$

representing scenarios where argument a has the same label in all the preferred labeling of a scenario. It is $A_{IN}^{pr-} \subseteq A_{IN}^{pr+}, A_{OUT}^{pr-} \subseteq A_{OUT}^{pr+}, A_U^{pr-} \subseteq A_U^{pr+}$ and the two sets of scenarios identify an upper and lower probability level. We add a last useful notation. We write A_{out} for all the scenarios where a holds and it results labeled *out*. Note that $A_{OUT} = \bar{A} + A_{out}$.

Definition 8. *We define the probabilities of acceptance (5), of rejection (6) and undecided probability (7) of argument a for grounded and preferred semantics as follows:*

$$P_A^g = P(A_{IN}^g), P_A^+ = P(A_{IN}^{pr+}), P_A^- = P(A_{IN}^{pr-}) \qquad (5)$$
$$\overline{P_A^g} = P(A_{OUT}^g), \overline{P_A^+} = P(A_{OUT}^{pr+}), \overline{P_A^-} = P(A_{OUT}^{pr-}) \qquad (6)$$
$$U_A^g = P(A_U^g),\ U_A^- = P(A_U^{pr-}),\ U_A^+ = P(A_U^{pr+}) \qquad (7)$$

Example 1 Let us consider the graph of figure 1 (A), and let us study the properties of argument a. There are 3 arguments, thus $2^3 = 8$ scenarios. Let us presume $P(A) = P(B) = P(C) = 0.8$ and a, b, c are statistically independent. Let us start computing A_{IN}^g. Argument a is

labeled *in* in all the scenarios where it holds and b does not hold (and c becomes irrelevant). Using our notation $A = A\bar{B}$ (i.e. the set of subgraphs $\{\{a\},\{a,c\}\}$. It is *undec* when all the arguments are present, i.e. the single scenario $A_U^g = ABC$ (i.e. $\{\{a,b,c\}\}$) and it is labeled *out* when it is assumed not to hold or when b is *in* and c is *out*, i.e. $A_{OUT}^g = \bar{A} + AB\bar{C}$ (set $\{\emptyset,\{b\},\{c\},\{a,b\},\{b,c\}\}$). By inserting numerical values we have:

$$P_A^g = 0.16,\ U_A^g = 0.512,\ \overline{P_A^g} = 0.328.$$

Regarding preferred semantics, we can verify that:

$$A_{IN}^{pr+} = A(\bar{B} + BC), \qquad P\left(A_{IN}^{pr+}\right) \equiv P_A^+ = 0.672$$
$$A_{IN}^{pr-} = A\bar{B}, \qquad P\left(A_{IN}^{pr-}\right) \equiv P_A^- = 0.16$$
$$A_U^{pr+} = A_U^{pr-} = \emptyset$$
$$A_{OUT}^{pr+} = \bar{A} + AB, \qquad P(A_{OUT}^{pr+}) \equiv \overline{P_A^+} = 0.84$$
$$A_{OUT}^{pr-} = \bar{A} + AB\bar{C}, \qquad P(A_{OUT}^{pr-}) \equiv \overline{P_A^-} = 0.328.$$

COMPUTING A_{IN}: A RECURSIVE ALGORITHM

This section presents an algorithm to compute A_{IN}, A_{OUT} under grounded semantics. Given a starting argument a and a label $l \in \{in, out\}$, we need to find the set of subgraphs where argument a is legally labeled *in*. The idea is to traverse the transpose graph (a graph with reversed arrows) from a down to its attackers, propagating the constraints of the grounded labeling. While traversing the graph, the various paths correspond to a set of subgraphs. The constraints needed are listed in definition 5 and theorem 1. If argument a – attacked by *n* arguments x_n – is required to be labeled in, we impose the set A_{IN} to be:

$$A_{IN} = A \cap \left(X_{1_{OUT}} \cap X_{2_{OUT}} \cap \ldots \cap X_{n_{OUT}}\right) \qquad \text{condition (1)}$$

i.e. argument a can be labeled *in* in the subgraphs where:

1. a is present in the subgraph (i.e. the set A) and
2. all the attacking arguments x_i are labeled out (sets $X_{i_{OUT}}$).

If a is required to be labeled out, the set of subgraphs is:

$$A_{OUT} = \bar{A} \cup A \cap \left(X_{1_{IN}} \cup X_{2_{IN}} \cup \ldots \cup X_{n_{IN}}\right) \qquad \text{condition (2)}$$

i.e. a is labeled out in all the subgraphs where it is not present or at least one of the attackers is labeled in. Thus we recursively traverse the graph, finding the subgraphs that are compatible with the starting label of a. The sets $X_{n_{OUT}}$, $X_{n_{IN}}$ are found when terminal nodes are reached. When a terminal node x_T is reached the following conditions are applied:

1. if x_T is required to be in then $X_{T_{IN}} = X_T$
2. if node x_T is required to be out then $X_{T_{OUT}} = \overline{X_T}$

The way the algorithm treats cycles guarantees that only grounded complete labelings are identified. If a cycle is detected, the recursion path terminates, returning an empty set that also has the effect of discarding all the sets of subgraphs linked with a logical AND (by condition 1) to the cyclic path. We present the pseudo-code of the algorithm, while Table 1 describes the steps for computing A_{IN} in the graph of figure 2 right.

Algorithm 2 - The Recursive FindSet(A,L,P) Algorithm

```
A is a node, L a label (IN or OUT), P is the list of parent nodes, Cset holds
the partial result of the computation of conditions (1) and (2).
FindSet(A,L,P):
  if A in P:
     return empty_set // Cycle found
  if L = IN:
     if A terminal:
        return a // Terminal condition for IN Label
     else:
        add A to P
        for each child C of A
           Cset = Cset AND FindSet(C,OUT,P)
        return (a AND Cset)              // condition 1
  if L = OUT:
     if A terminal:
        return NOT(a) // Terminal condition for OUT Label
     else
        add A to P
        for each child C of A
          Cset = Cset OR FindSet(C,IN,P)
        return (NOT(a) OR (a AND Cset))      //condition 2
```

Table 1. Recursively applying Algorithm 2 on the graph of figure 3 left.

	Node, label	Constraint	Parent List	Comment
1↓	A_{IN}	$A_{IN} = A \cap B_{OUT}$	[]	*a must exist and b=OUT*
2↓	B_{OUT}	$B_{OUT} = \bar{B} \cup (B(C_{IN} \cup D_{IN}))$	$[a]$	*b is out when b does not exist or b exists and c = in or d = in*
3=	C_{IN}	$C_{IN} = C \cap A_{OUT}$	$[a, b]$	*c=IN when c exists and a=OUT. Cycle with a,* $C_{IN} = \emptyset$
4=	D_{IN}	$D_{IN} = D$	$[a, b]$	*d is initial*
5↑	B_{OUT}	$B_{OUT} = \bar{B} \cup (B \cap \mathrm{D})$		
6↑	A_{IN}	$\mathrm{A_{IN}} = \mathrm{A} \cap \left(\bar{\mathrm{B}} \cup (\mathrm{B} \cap \mathrm{D})\right) = A\bar{\mathrm{B}} + ABD$		

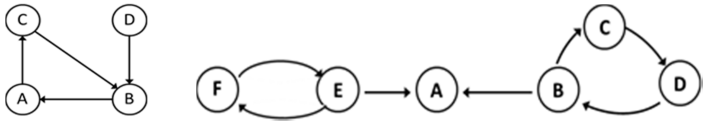


Figure 3.

Extension to Preferred Labeling

The constraints used in Algorithm 2 – argument a is *in* when all the attackers are *out* and a is *out* if one attacker is *in* – are properties of any complete labeling. The way algorithm 2 treats cycles – it always assigns the *undec* label to their arguments – guarantees that we collect only grounded complete labeling. Since a preferred labeling is complete, the extension of algorithm 2 to the case of preferred semantics requires changing only the way cycles are treated. The following lemma is useful in always assigning an argument labeled *in* in a complete labeling to the computation of A_{IN}^{pr+}.

Lemma 1. *If a is labeled in (or out) in a complete labeling of a scenario, then the scenario can be assigned to A_{IN}^{pr+} (or A_{OUT}^{pr+}).*

Proof. If a is labeled *in* in a complete labeling C of a scenario s, either C is the preferred one maximizing $in(C,s)$ w.r.t. to set inclusion, or there is another C' with $in(C,s) \subset in(C',s)$. Since $a \in in(C,s)$, then $a \in in(C',s)$ and scenario s contributes to A_{IN}^{pr+}.

We return to the treatment of cycles. When a cycle is detected, the labeling of an even-length cycle is consistent since the argument that is visited twice and identifies the cycle is required to have the same label. However, an odd-length cycle creates an inconsistent *undec* labeling not contributing to A_{IN}^{pr+} or A_{OUT}^{pr+}. Thus we assign a clause (i.e. set of scenarios) to A_{IN}^{pr+} when a consistent cycle is found, while we reject the scenario otherwise. Note how the skeptical sets A_{IN}^{pr-} and A_{OUT}^{pr-} can be derived once the credulous sets are computed using equations 4. In traversing the graph, we thus need to remember the label required for an argument to check if the cycle can be consistently labeled. It is important to bear in mind that A_{out} (small letter for the label) identifies the set of scenarios where argument a exists and it is labeled *out* (note that $A_{OUT} = \bar{A} + A_{out}$). Let us consider the graph depicted in figure 3 left. This contains both odd and even length cycles. Table 2 shows the steps in computing A_{IN}^{+}.

Table 2. Computing A_{IN}^{pr+} of figure 3 left

1	$A_{IN} = AB_{OUT}E_{OUT}$		
2a	$B_{OUT} = \bar{B} + B_{out}D_{IN}$	2b	$E_{OUT} = \bar{E} + E_{out}F_{IN}$
3a	$B_{out}D_{IN} = B_{out}DC_{OUT}$	3b	$E_{out}F_{IN} = E_{out}FE_{OUT}$
4a	$B_{out}DC_{OUT} = B_{out}D\bar{C} + B_{out}DCB_{IN} = BD\bar{C} + \emptyset$ $(inconsistent\ cycle)$	4b	$E_{out}FE_{OUT} = EF$ $(consistent\ cycle: e\ and\ f\ exists, e = in, f = out)$
5a	$B_{OUT} = \bar{B} + BD\bar{C}$	5b	$E_{OUT} = \bar{E} + EF$
6	$A_{IN}^{+} = A(\bar{B} + BD\bar{C})(\bar{E} + EF) = A(\bar{B}\bar{E} + \bar{B}EF + BD\bar{C}\bar{E} + BD\bar{C}EF)$		

Note how the 3-length cycle creates an inconsistent situation $B_{out}DCB_{IN}$ (argument b has to exist and be labeled *in* and *out* at the same time) while $E_{out}FE_{OUT}$ can be labeled consistently (the cycle is consistent when argument e is required to exist and labeled *out*). We can verify that A_{IN}^{-} differs from A_{IN}^{+} since it discards the even-length cycle EF, thus the path AEF (and any path in an *and* condition with it) are not in A_{IN}^{-} .

Notable Examples: Accrual of Attacks

Let us consider the PAF in figure 4 left. Argument a is labeled *in* iff a holds and both b and c are labeled out (satisfied only when b and c do not hold, since b and c are inital). Thus:

$$A_{IN} = A\overline{BC} \ ; \ A_{OUT} = \bar{A} + AB + A\bar{B}C$$

which represents the accrual of a probabilistic network. Note how this differs from mainstream numerical argumentation approaches where no accrual occurs [9, 1] and the effect of *n* arguments can be equated with the effect of the argument with the maximum degree. In the probabilistic accrual every argument counts, to the extent given by their joint probability.

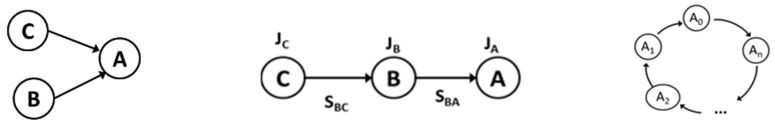


Figure 4. *Accrual (left), Reinstatement (center) and a circle of arguments (right).*

Notable Examples: Reinstatement

When an argument b attacks a, argument a is *in* iff a exists and b does not hold, thus $A_{IN} = A\bar{B},\ A_{OUT} = \bar{A} + AB$. The case of 3 isolated arguments (figure 4 center) is useful in analyzing the reinstatement property. It is easy to verify this:

$$A_{IN} = A\bar{B} + ABC\ , \qquad A_{OUT} = \bar{A} + AB\bar{C}$$

In the generic case of n arguments, A_{IN} we have:

$$A_{IN} = A\overline{B_1} + AB_1B_2\overline{B_3} + \cdots \quad ; \quad A_{OUT} = \bar{A} + AB_1\overline{B_2} + AB_1B_2B_3\overline{B_4} + \cdots$$

The statistical independence of arguments makes evident some reinstatement properties. By using the fact that $P(X) = 1 - P(\bar{X})$ the expression of $P_{IN}(A)$ can be written as follows:

$$P_{IN}(A) = P(a) - P(a)P(b_1) + P(a)P(b_1)P(b_2) - \cdots$$

Note how $P_{IN}(A)$ is the sum of the probabilities of all the paths terminating in a. The paths terminating with a defender (odd length) are positive factors contributing to $P_{IN}(A)$ and vice versa for the path ending with an attacker of a. The reinstatement effect of attackers and defenders decreases with the length of the path, but all influence $P_{IN}(A)$.

We now compare the reinstatement of a PAF to other argumentation approaches.

PAF. An argument is always reinstated but with a lower probability (note how it is fully reinstated only when $P(c) = 1$). All the arguments on the chain contribute to the reinstatement with decreasing effect.

Dung's Abstract AF. In an abstract argumentation framework, arguments in a chain are reinstated at full potential, since the chain is made up of both *in* arguments (defended by the initial argument) and *out* arguments (attacked by the initial argument).

Pollock [9]. The arguments presented in [9] have a degree of justification in $\mathbb{R}_0^+$. This degree is a subtractable cardinal quantity. Referring to figure 4 center, when an argument C of degree $\mathrm{J_C}$ attacks b of degree $\mathrm{J_B}$ then the new degree of b after the attack is $\mathrm{J'_B} = \max(\mathrm{J_B} - \mathrm{J_C}, 0)$ and $\mathrm{J'_A} = \max(\mathrm{J'_B} - \mathrm{J_A}, 0)$. Thus argument a is fully reinstated when $\mathrm{J_C} > \mathrm{J_B}$, and not reinstated (at all) if $\mathrm{J'_B} > \mathrm{J_A}$ while in all other cases it is reinstated proportionally to $\mathrm{J_B} - \mathrm{J_C}$. Note how, unlike a PAF, it is *relative comparison* of degrees that decides the reinstatement.

Cayrol and Lagasquie-Schiex's vs-defense. In [1], a strength is attached to attack relations. Argument a is fully reinstated iff $\mathrm{S_{BC}} \geq \mathrm{S_{BA}}$, otherwise it remains totally defeated. Thus it is the relative comparison of the strength of the attacks that defines reinstatement. We notice that, although attack from C to B is logically antecedent to that from B to A, it is neglected if $\mathrm{S_{CB}} < \mathrm{S_{BA}}$. Generally, apart from PAF, the distance of an argument from the first node a in the chain is not linked to its impact on a and the strength of reinstatement is usually a relative comparison of strength, with the consequence that some attacks might be neglected.

In the case of arguments forming a cycle, let us consider the case of two arguments a and b rebutting each other. In a *PAF* with grounded semantics we have:

$$A^g_{IN} = A\bar{B}\,, A^g_{OUT} = \bar{A}\,, A^g_U = AB$$

A_{IN} is unchanged compared to the case where b attacks a but not vice versa, while the not null A_U decreases A_{OUT}. The counter attack from a to b does not improve $P_{IN}(a)$. This is the expected behavior of grounded semantics: an argument cannot reinstate itself but it needs a third external argument. In the case of grounded semantics, the skeptical set A^+_{IN} neglects the attacker b ($A^+_{IN} = A$) since a fully reinstates itself; the skeptical set is equal to the grounded case ($A^-_{IN} = A\bar{B}$). It is also $A^-_U = A^+_U = \emptyset$ and $A^+_{OUT} = \bar{A} + AB$, $A^-_{OUT} = \bar{A}$

SENSITIVITY TO CHANGES IN P OR Ar

The expression of A_{IN} (A_{OUT} or A_U) allows us to study a set of properties of argument a in relation to the arguments in Ar. An interesting set of properties is the study of the sensitivity of $P_{IN}(a)$ to a change in the probability P of an argument (for instance when new evidence on its validity are found); or to the addition/removal of an argument to/from Ar (here we limit to the situation of adding an argument attacking or rebutting an argument in Ar).

The interest is mainly due to its applications: an agent might want to understand the effect of extra evidence affecting an argument, or which are the arguments that have the maximum impact on $P_{IN}(a)$. In a legal dispute, a lawyer deciding on his/her its strategy might focus on which arguments he should challenge in court. We first define two useful measurements of change.

Definition 9. *The partial differential gain of argument a w.r.t. to argument b is* $\frac{\partial P_{IN}(a)}{\partial P(b)}$

The sign of the differential gain tells whether an argument should be attacked or defended in order to increase $P(A_{IN})$, while its value quantifies the impact of the argument on a.

Definition 10. *We call argument b a dialectical defender of a iff* $\frac{\partial P_{IN}(a)}{\partial P(b)} > 0$, *and b is a dialectical attacker of a when* $\frac{\partial P_{IN}(a)}{\partial P(b)} < 0$

Let us now define a particular expression of A_{IN}, that contains information about how the a behaves in relation to changes.

Definition 11. *Given an argument a and an argument b for which there is at least a path from a to b, we call the normal form of A_{IN} w.r.t. to b the following expression of A_{IN}:*

$$A_{IN} = A(XB + Y\bar{B}) + C, \text{ with } X \cap Y = \emptyset.$$

The term AXB represents the set of scenarios contributing to A_{IN} where argument b is assumed to hold. $AY\bar{B}$ represents the set of scenarios contributing to A_{IN} but requiring b not to hold. C is the set of scenarios contributing to A_{IN} where the status of b is irrelevant.

It can be proved that argument b is always labeled *in* in the set AXB. If, *ad absurdum*, b is labeled *out* in a scenario s in AXB, then the same scenario where b does not hold has the same labeling and it would also contribute to A_{IN} and thus the status of b would have been irrelevant and the scenario would have been part of the set C and not AXB, contradicting the hypothesis. Hence we can rewrite the normal form as follows:

$$A_{IN} = A(XB_{IN} + Y\bar{B}) + C \text{ , with } X \cap Y = \emptyset.$$

Why is this form useful? The expression makes explicit the contribution of arg. b to A_{IN}.When $= Y = \emptyset$, $A_{IN} = C$ and b does not contribute to A_{IN} and thus $P_{IN}(a)$. A change in $P(b)$ does not affect a. Thus b is neither an attacker nor a defender of a, and differential gain

w.r.t. b is null. If $X = \emptyset$ then $A_{IN} = AY\bar{B} + C$. We can compute the dialectical gain w.r.t. argument b. In the rest of this section, we use P also for P_{SS} to simplify the notation (bearing in mind how the probability distribution P_{SS} defined over a set of scenarios is derived from the distribution P defined over arguments, as shown in equation 3).

We compute the dialectical gain by first evaluating the difference in $P(A_{IN})$ when $P(b)$ is increased by Δb. Since $P(b) = 1 - P(\bar{b})$, a change in $P(b)$ of Δb is a decrement of $P(\bar{b})$ by Δb, and since $P(AY\bar{B}) = P(AY \wedge \bar{B}) = P(AY|\bar{B})P(\bar{B}) = P(AY|\bar{B})P(\bar{b})$ we write:

$$P(A_{IN})_{+\Delta b} - P(A_{IN}) = P(AY|\bar{B})\big(P(\bar{b}) - \Delta b\big) - P(AY|\bar{B})P(\bar{b}) = -P(AY|\bar{B})\Delta b < 0$$

The differential gain is:

$$\frac{\partial P_{IN}(a)}{\partial P(b)} = \lim_{\Delta b \to 0} \frac{P(A_{IN})_{+\Delta b} - P(A_{IN})}{\Delta b} = -P\big(AY|\bar{b}\big) < 0$$

meaning that incrementing b will decrease the value of $P(A_{IN})$, thus making b a dialectical attacker of a. In case of statistical independence we have $\frac{\partial P_{IN}(a)}{\partial P(b)} = -P(A)P(Y)$.

Similarly, if $Y = \emptyset$ then b results a dialectical defender and the differential gain is $P(AX|b)$. We bear in mind that $P(A)$ is constant in the computation while $P_{IN}(a)$ varies. In the general case, when both X and Y are not empty, b is a defender when $P(AX|b) > P\big(AY|\bar{b}\big)$ and an attacker when $P(AX|b) < P\big(AY|\bar{b}\big)$. Thus:

Proposition 1. *Given the normal form of A_{IN} w.r.t. b $A_{IN} = A(XB_{IN} + Y\bar{B}) + C$, the differential dialectical gain of a w.r.t. to b is:*

$$\frac{\partial P_{IN}(a)}{\partial P(b)} = P(AX|b) - P\big(AY|\bar{b}\big)$$

In case of statistical independence of arguments it is: $\frac{\partial P_{IN}(a)}{\partial P(b)} = P(A)(P(X) - P(Y))$

Example 3. Let us consider figure 3 (left) again and let us presume that $P(a) = 1, P(b) = 0.8, P(c) = P(d) = 0.7$ all independent. We saw that $A_{IN} = A(BD + \bar{B})$ and thus $P_{IN}(A) = 0.2 + 0.56 = 0.76$. Table 3 reports the normal forms of a w.r.t. to argument b, d and c.

Table 3. Dialectical gains for Example 4

Normal form of a w.r.t.		X	Y	C	$\frac{\partial P_{IN}(a)}{\partial P()} = P(A)(P(X) - P(Y))$
$\boldsymbol{d}$	$A_{IN} = AB\boldsymbol{D} + A\bar{B}$	B	$\emptyset$	$A\bar{B}$	$P(A)P(B) = 0.8$
$\boldsymbol{b}$	$A_{IN} = A\bar{D}\bar{\boldsymbol{B}} + AD$	$\emptyset$	$A\bar{D}$	AD	$-P(A)P(\bar{D}) = -0.3$
$\boldsymbol{c}$	$A_{IN} = A(BD + \bar{B})$	$\emptyset$	$\emptyset$	$A(BD + \bar{B})$	0

Argument d is a defender ($Y = \emptyset$), b an attacker ($X = \emptyset$) and c is neither an attacker nor a defender. The dialectical gain of a w.r.t. b is -0.3 while for d is 0.8. Thus, if we want to increase $P(A_{IN})$ it is better to increase $P(d)$ rather than reduce $P(b)$. How much do we need to increase $P(d)$ to get $P(A_{IN}) > 0.9$? Since $P(A_{IN})$ needs to be increased by 0.14 (from 0.76 to 0.9), and since $\frac{\partial P_{IN}(a)}{\partial P(d)}$=0.8, $P(d)$ should be increased by $0.14/0.8 = 0.175$. Note that the same effect could have been obtained by decreasing $P(b)$ of a greater quantity equal to 0.47.

Adding a new argument attacking or rebutting an argument in Ar

In a dialectical process an argument is usually modified not only internally but by adding a new (maybe indirect) attack on it as depicted in figure 5. Let us study the situation in which a new argument w attacks b, but not vice-versa.

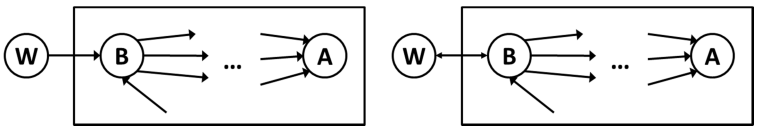


Figure 5. Attacking B via argument W

Proposition 2. *Given the normal form of A_{IN} w.r.t. b $A_{IN} = A(XB_{IN} + Y\bar{B}) + C$, if a new initial argument w is added to the argumentation framework and if b is the only argument attacked by w, then the dialectical gain of a w.r.t. to w is:*

$$\frac{\partial P_{IN}(a)}{\partial P(w)} = -\big(P(AXB|w) - P(AYB|w)\big) \text{ and}$$

$$\frac{\partial P_{IN}(a)}{\partial P(w)} = -P(b)\frac{\partial P_{IN}(a)}{\partial P(b)} \ (if\ arguments\ b\ and\ w\ are\ independent)$$

Proof. A convenient way to show how A_{IN} changes is to consider argument b attacked by w and to substitute arguments b and w with an argument b' that encompasses the effect of w on b. Argument b' will be labeled *out* in the scenarios where b is assumed not to hold or where it holds but argument w defeats it. b' is labeled *in* when argument b holds and argument w does not hold. Thus b' has the following properties: $B'_{IN} = B_{IN}\bar{W}$ and $B'_{OUT} = \bar{B} + BW$. Hence by substituting in the normal form we have:

$$A^w_{IN} = A(XB_{IN}\bar{W} + Y(\bar{B} + BW)) + C$$

The difference in $P(A_{IN})$ is:

$$P(A^w_{IN}) - P(A_{IN}) = P(AXB\bar{W}) + P(AY\bar{B}) + P(AYBW) + P(C) - P(AXB) - P(AY\bar{B}) - P(C)$$
$$= -P(AXBW) + P(AYBW) = -P(w)(P(AXB|w) - P(AYB|w))$$

And the differential gain of a w.r.t. w is $\frac{\partial P_{IN}(a)}{\partial P(w)} = -(P(AXB|w) - P(AYB|w))$. If b and w are independent we have $\frac{\partial P_{IN}(a)}{\partial P(w)} = -P(A)\big(P(X) - P(Y)\big)P(B) = -P(b)\frac{\partial P_{IN}(a)}{\partial P(b)}$

Let us now presume that w rebuts b and b is the only argument attacked by w (figure 5 right).

Proposition 3. *Given the normal form of A_{IN} w.r.t. to b $A_{IN} = A(XB_{IN} + Y\bar{B}) + C$, if a new argument w is added to the argumentation framework and if w and b are rebuttals and if b is the only argument attacked by w, then the dialectical gain of a w.r.t. to w is:*

$$\frac{\partial P_{IN}(a)}{\partial P(w)} = -P(AXB|w)$$

Proof. When w is added, the set of scenarios in C are clearly still contributing to A_{IN} since the status of argument b is irrelevant. The sets of scenarios AXB_{IN} and $AY\bar{B}$ are not affected by argument w when w is assumed not to hold (thus $AXB\bar{W}$ and $AY\bar{B}\bar{W}$contributes to A_{IN}) while when w is assumed to hold, the set of scenarios AXB_{IN} require argument b to be labeled *in*, which is no longer the case since w rebuts b and thus b cannot be labeled *in*.

Regarding the scenarios in $AY\bar{B}$, they still contribute to A_{IN} since b is required not to hold and so w is disconnected from a and therefore irrelevant. Thus:

$$A_{IN}^{W} = \bar{W}(AXB + AY\bar{B} + C) + W(AY\bar{B} + C) = AXB\bar{W} + AY\bar{B} + C$$
$$P(A_{IN}^{W}) - P(A_{IN}) = P(AXB\bar{W}) + P(AY\bar{B}) + P(C) - P(AXB) - P(AY\bar{B}) - P(C)$$
$$= -P(XBW) = -P(w)P(AXB|w)$$

And the differential gain of a w.r.t. w is $\frac{\partial P_{IN}(a)}{\partial P(w)} = -P(AXB|w)$. In the case of the statistical independence of arguments, it is: $\frac{\partial P_{IN}(a)}{\partial P(w)} = -P(A)P(X)P(b)$

We note how the dialectical gains w.r.t. b and w have opposite sign, as expected. In the case of a rebuttal, proposition 3 states that the dialectical gain is always negative or null (when $X = \emptyset$), consequence of the fact that a rebuttal under grounded semantics does not defeat the attacked argument.

Example 4. We continue example 3, where we found that $\frac{\partial P_{IN}(a)}{\partial P(d)}$=0.8 and $\frac{\partial P_{IN}(a)}{\partial P(b)} = -0.3$**.** Let us presume that we could attack argument d and we want again to bring $P(A_{IN})$ above 0.9. If we attack d we have no way to increase $P(A_{IN})$, since the dialectical gain of a w.r.t. d is positive. Let us consider argument b. The normal form is $A\bar{D}\bar{B} + AD$ and the dialectical gain w.r.t. to b is -0.3. If we attack b with a new argument w, according to prop. 2, the dialectical gain is $-0.3 * -P(b) = 0.24$. In order to increase $P(A_{IN})$ by 0.14, argument w should at least have a strength of $0.14/0.24$, about 0.583. If we rebut argument b with w, since $X = \emptyset$ in the normal form w.r.t. b, proposition 3 tells us that argument w would have no effect.

AN EXAMPLE OF APPLICATION: A LEGAL CASE

In order to make our PAF applicable, we must provide a structure for arguments and attacks. We describe a *single rule argument* model adapted from [10], that keeps the discussion simple, but is adequate for illustration. Let us consider a set of atomic propositions $F = \{a_1, .., a_n\}$ and the propositional language $\mathcal{L}$ closed under negation with atoms in F and connectives $\{\wedge, \neg\}$. We define an argument as a defeasible inference rule of the kind: $\varphi \rightarrow \phi$ where $\varphi, \phi \in \mathcal{L}$. Defeasible means that a rule admits exceptions and it can be invalidated by other arguments. Note how our definition is limits argument to a single rule (adequate for our illustrative example), instead of including derivation trees composed by chain of rules as in [10].

If we call $\mathcal{R}$ is the set of rules, we define a function of *conflict* $\bar{\ }: \mathcal{L} \rightarrow 2^{\mathcal{L}} \cup 2^{\mathcal{R}}$, that allows us to define asymmetric conflicts among propositions and rules. If $a = \bar{b}$ then when a is asserted b cannot be asserted, but not viceversa. It is $a = \overline{\neg a}$ and $\neg a = \bar{a}$. If $a = \bar{r}$ and r is a rule, this means that a is an exception to rule r, when a is asserted r is invalid. Note how this function models conflicts, but also preferences: $a = \bar{b}$ could model the fact that a is preferred to b.

Each argument has an associated probability equal to the probability $P(\varphi)$ of its premises φ (this means we know the joint probability $P()$ of all the propositions used in the premises of the arguments, representing our available evidence used to build arguments). We define three forms of attack: rebuttals, undermining and undercutting. Given two arguments $A: \varphi_A \rightarrow \phi_A$ and $B: \varphi_B \rightarrow \phi_B$, we say that A rebuts B iff $\phi_A = \overline{\phi_B}$ and $\phi_B = \overline{\phi_A}$, A undermines B iff $\phi_A = \overline{\varphi_B}$, i.e. an argument conclusion excludes a premise of another argument, and A

undercuts B iff $\phi_A = \bar{B}$, i.e. the conclusion ϕ_A invalidates the rule B. An undercutting attack model the fact that defeasible rules (such as B) might have exceptions (such as ϕ_A). Note how a rebuttal is always a symmetric attack, an undermining could be (it is iff $\phi_A = \overline{\varphi_B}$ and $\varphi_B = \overline{\phi_A}$), while an undercut is always defined as asymmetric (the exception defeats the rule but not viceversa). Given a set of arguments Ar of the kind $\varphi_i \rightarrow \phi_i$, we can represent them on a $PAF = ((Ar, R), P)$ using as P the probability of each argument $P(\varphi_i)$, and using rebuttals and undercutting attack to define the attack relation R.

We present an application of PAF to legal reasoning. Paul and John are on trial for the assassination of Sam. The following evidence is available. First it is known with certainty that John entered the room where the murder took place at 1 pm and left at 3 pm, while Paul entered at 3 pm and he was found by the police at 5 pm. A forensic test suggests that the probability that Sam died between 1 pm and 3 pm is 0.6 and between 3 pm and 5 pm is 0.4. The test used has an accuracy of 0.9. Thus we have the following arguments (in square brackets the probability of each premise):

R_J) a_1: *(John was in the room between 1 to 3* ***[1])*** $\wedge$ a_2: *(the medical test says that Sam died between 1 and 3* ***[0.6]*** *)* $\rightarrow$ a_3: *(John shot Sam)*

R_P) a_4: *(Paul was in the room between 3 and 5* ***[1])*** $\wedge$ a_5: *(the test says that Sam died between 3 and 5* ***[0.4])*** $\rightarrow$ a_6: *(Paul shot Sam)*

M_t) a_7: *(The test is void* ***[0.1])*** $\rightarrow$ $\overline{R_J} \wedge \overline{R_P}$ *(Sam's time of death cannot be estimated)*

The probability of each argument is: $R_J = 0.6, R_P = 0.4$ and $M_t = 0.1$. We also have $P(R_J \wedge R_P) = 0$, since Sam either died between 1 and 3 pm or between 3 and 5 pm. Argument M_t undercuts (invalidates) both R_J and R_P. Since $R_J = 0.6 > 0.5 > R_P$, John's lawyer asks for a fingerprint analysis of the murder weapon. The result is that with a probability of 0.7 the fingerprints are Paul's. The lawyer thus proposes a new argument:

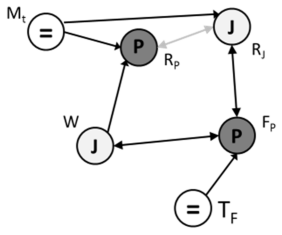
F_P) a_8: *(The test says that the fingerprints are Paul's* ***[0.7])*** $\rightarrow$ a_6: *(Paul shot Sam)*

This argument rebuts R_J (conclusions are conflicting, since it is clearly $a_3 = \overline{a_6}$ and $a_6 = \overline{a_3}$). In any case, further analysis by the police labs states that the weapon was tampered with, and the test is only 50% reliable. The new argument (with a probability of 0.5):

T_F) a_9: *(the test is void* ***[0.5])*** $\rightarrow$ $\overline{F_P}$ *(fingerprints are not valid evidence)*

undercuts the validity of F_P. Paul's lawyer counter-attacks using the testimony of a credible witness who heard a shot at 2 pm, when only John was in the room. The witness is reputable with a probability of 0.8. Thus the following argument is built by the judge:

W) a_{10}: *(A shot was heard at 2pm* ***[0.8])*** $\wedge$ a_1: *(John was in the room between 1 to 3* ***[1])*** $\rightarrow$ a_3: *(John shot Sam)* $\wedge$ $\bar{a}_5$ *(Sam died at 2 pm, not between 3 pm and 5 pm)*



Argument	Probability	
R_J	**0.6**	$P(R_J \wedge R_P) = 0$
R_P	**0.4**	
M_t	**0.1**	All arguments independent
W	**0.8**	
T_F	**0.5**	
F_P	**0.7**	

Figure 6. Argumentation Graphs for the legal case

Note how the way we wrote argument W means that the judge considers the witness' testimony a more definitive evidence than the medical test (W implies $\overline{a_5}$), and thus argument W undercuts P_R and rebuts F_P. The final graph is depicted in figure 6. A grey line indicates rebuttals between arguments with mutually exclusive premises (R_J and R_P). We marked with P the arguments whose conclusion is against Paul and with J the arguments against John. Other arguments are marked =, indicating they do not add to the conclusion but interact with P and J.

Analysis

There are 6 arguments and potentially 64 different scenarios. Let us call G_p and G_J the set of scenarios where Paul (or John) are guilty (i.e. at least one argument supporting the conclusion is labeled *in*), and $P_P = P(G_P)$ and $P_J = P(G_J)$. There are two arguments against John, R_J and W. If we apply algorithm 2 to find the $R_{J_{IN}}$ and W_{IN} sets, it is easy to verify that we obtain:

$$R_{J_{IN}} = R_J\overline{M_T}\ \left(\overline{F_P} + F_pT_F\right);\ W_{IN} = W\left(\overline{F_P} + F_pT_F\right)$$

$$G_j = R_{J_{IN}} \vee W_{IN} = R_J\overline{M_T}\ \left(\overline{F_P} + F_pT_F\right) + W\left(\overline{F_P} + F_pT_F\right) = 0.6278$$

Note that, in computing $R_{J_{IN}}$ we do not care about attacker R_P since $P\left(R_j \wedge R_p\right) = 0$. Regarding Paul, there are two arguments R_P and F_P against him and he is guilty when $R_P \vee F_P$:

$$R_{P_{IN}} = R_P\overline{M_T}\,\overline{W} \text{ and } F_{P_{IN}} = F_P\overline{T_F}\,(\overline{R_J} + R_JM_T)$$

$$G_P = R_{P_{IN}} \vee F_{P_{IN}} = R_P\overline{M_T}\,\overline{W} + F_P\overline{T_F}\,\overline{R_J} \ = R_P\overline{M_T}\,\overline{W} + F_P\overline{T_F}(\overline{R_p}M_T + R_P\overline{M_T} + R_PM_TW) = 0.284$$

John's lawyer has to find a way of decreasing the probability of the evidence against John. If we compute the dialectical gain, we find that the dialectical gain of G_j w.r.t. w is equal to 0.416, w.r.t. M_t is -0.052 and w.r.t. to T_F is 0.6104. Therefore the the best chance of minimizing John's guilt is to decrease T_F, i.e. to show that the test is valid in more than 50% of cases. In order to put the probability P_J below 0.5 we need a change in T_F of $(0.5 - P_J)/0.6104 = 0.21$, therefore T_F should go down to about 29% from 50%. An alternative is to decrease W – make the witness against Paul less credible. In that case, W should be decreased by about 0.307 (from the current 0.8 down to below 0.5) in order to bring G_J below 0.5. M_t has too minimal a dialectical gain to be used.

Regarding Paul, his lawyer wants to know if the potential moves of John's lawyer could affect Paul, i.e. could they increase G_P above 0.5. The dialectical gains for G_p are shown in table 4 left. Since $P_P = 0.284$, G_P goes above 0.5 if either W is decreased by 0.416 or if T_F is decreased by 0.4917 down to 10%, meaning that a fingerprint test should have a 90% accuracy. These values could be safe enough for the lawyer and they are greater than what is needed by John's lawyer to bring P_P below 0.5, so in this example there could be a collusion where both of the suspects are below 0.5. Nevertheless, Paul's lawyer should focus on further invalidating the fingerprint test.

Table 4. Dialectical Gain of Arguments $G_P$and G_Jw.r.t. to W, T_F, M_t

Dialectical gain of G_P w.r.t. to W, T_F, M_t		Dialectical gain of G_Jw.r.t. to W, T_F, M_t	
W	-0.519	W	0.416
M_t	-0.022	M_t	-0.052
T_F	-0.4396	T_F	0.6104

RELATED WORKS

The idea of merging probabilities and abstract argumentation was first presented by Dung et al. [3], and a more detailed formalization was provided by Li et al. [4], along with the works by Hunter [6] and Thimm [14]. In Li et al.'s definition P is not a joint probability but a scalar function $Ar \rightarrow [0,1]$ and a similar scenario-like approach (extension-based rather than argument-based) is used. Li et al.'s work is limited to fully independent arguments with grounded semantics, and no exact computation behind the brute force algorithm is analyzed ,while our paper also considers preferred semantics, providing an algorithm to compute PAF and studying the behavior of PAF w.r.t. to reinstatement, accrual, and response to changes.

Thimm in [14],and Hunter [6] in his epistemic approach, start from a complementary angle. Both authors assume that there is already an uncertainty measure – potentially not probabilistic – defined on the admissibility set of each argument (i.e. P_{IN} is given as a function $P_{IN}: Ar \rightarrow [0,1]$). Starting from P_{IN} rather than P poses the question: which P_{IN} assignments are acceptable? The authors both argue that only a subset of these measurements can be sensibly associated with an argumentation framework. They define a series of rules to identify a rationally acceptable probability distribution of P_{IN}, such as the rationality and *p-justifiability* properties. In our paper we follow a complementary approach, since our aim is to start from P (assumed to be a probability measure) and then compute P_{IN}.

Regarding other works investigating gradualism in argumentation, we first mention Pollock's work on degrees of justification [9]. Pollock rejects the use of probabilities to propagate numerical values on an argumentation framework, but he considers probabilities the only valid proxy for argument strength, and he uses the statistical syllogism as the standard comparison to measure strengths. Pollock considers the strengths of arguments as cardinal quantities that can be subtracted. The accrual of arguments is denied (except for a rebutting and an undercutting argument) and it is the argument with the maximum strength that defines the attack. In an argument chain, it is the argument with minimum strength that defines the strength of the conclusions. The model proposed by Cayrol and Dupin de Saint-Cyr [5] infers a measure of argument strengths from their position in the argumentation framework. This extrinsic strength cannot be mapped to probability or beliefs, and leads to an ordering on the arguments that does not fit our problem. The *vs-defence* model, by Cayrol and and Lagasquie-Schiex [1], is an extension of AF where attacks have a strength associated with them. Argument admissibility status is the result of the comparisons of attack strengths. We have seen two main problems: there is no description about how to compute such a strength, how to practically set a priority level and a preferral order, as Pollock wrote in [9]: *if we were to be serious about arguments' strength, there must be a way to measure it*.

In [1], the authors propose an argumentation framework with various degrees of attacks. They extend a work by Martinez & Garcia [12] that first extended Dung's argumentation framework, introducing different levels of attacks. The work contributes to the development of argumentation with attacks of different strength. [8] was the first research to suggest the use of weights both on arguments and on attacks and Dunne et al. [11] have proposed weighted argument systems in which attacks have a numeric weight, indicating how reluctant one would be to disregard the attack. They accept that attacks can have different weights, and such weights might have different interpretations: an agent-based priority voting, or a measure of how many premises of the attacked argument are compromised.

CONCLUSIONS AND FUTURE WORKS

We have analyzed probabilistic argumentation frameworks and provided a first recursive algorithm to compute the probability of argument acceptance. We also studied various properties such as sensitivity to changes and behavior in the presence of reinstatement, accruals and cycles. We showed how *PAF* can be used as a tool to argue with probabilistic information. Our results could be used by agents involved in a discussion, in order to select the best move in a dialectical process or to analyze the sensitivity of the conclusions found. We believe that this is a contribution to the debate about gradualism in argumentation to justify further research in the theoretical and applicative studies of *PAF*. Future developments may lie in the extension to other forms of uncertainty such as possibility or fuzzy/multi-value logic. Much work has to be done on the computational aspects and optimization of the recursive algorithm proposed, and an evaluation of its efficacy against a baseline brute-force approach.

References

1. C. Cayrol, C. D. Lagasquie-Schiex. "Acceptability semantics accounting for strength of attacks in argumentation,"19th ECAI, pages 995-996. Lisbon, Portugal, 2010
2. P. Dung, “On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games,” Artificial Intelligence, vol. 77, pp. 321–357, 1995
3. P. Dung, P. Thang. "Towards (Probabilistic) Argumentation for Jury-based Dispute Resolution," COMMA 2010. IOS Press, 171-182
4. Hengfei L., N. Oren, T. J. Norman. Probabilistic Argumentation Frameworks. 1st TAFA, JICAI 2011, Barcelona, Spain
5. C. Cayrol, F. Dupin de Saint-Cyr (2010) "Change in Abstract Argumentation Frameworks: Adding an Argument", Vol. 38, 49-84
6. Hunter, A. "A probabilistic approach to modelling uncertain logical arguments." International Journal of Approximate Reasoning (2012).
7. Caminada, M.W.A., and D.M. Gabbay. "A logical account of formal argumentation." Studia Logica 93.2-3 (2009): 109-145.
8. Barringer H., D. Gabbay, “Temporal dynamics of support and attack networks,” Mechanizing Mathematical Reasoning. LNAI 2605. Springer, 2005, 59–98.
9. Pollock, J., "Defeasible reasoning with variable degrees of justification,", 2001 Artificial Intelligence, Vol 133, Pg. 233-282.
10. Prakken, Henry. "An abstract framework for argumentation with structured arguments." Argument and Computation 1.2 (2010): 93-124.
11. Dunne, P.E., A. Hunter, P. McBurney, S. Parsons, “Inconsistency tolerance in weighted argument systems,” in Proc of AAMAS, 2009.
12. Martinez, D.C., A. J. Garcia, “An abstract argumentation framework with varied-strength attacks,” in Proc of KR, 2008, pp. 135–143.
13. Vreeswijk, G. "Abstract argumentation systems," Artificial Intelligence 90, 225–279, 1997
14. Thimm, M. "A Probabilistic Semantics for abstract Argumentation," ECAI. 2012.