

Towards a new framework for recursive interactions in abstract bipolar argumentation

Claudette Cayrol, M-Christine Lagasquie-Schiex

IRIT, Université Paul Sabatier,
118 route de Narbonne, 31062 Toulouse, France
{ccayrol,lagasq}@irit.fr

Andrea Cohen

Institute for Computer Science and Engineering (ICIC), CONICET-UNS,
Av. Alem 1253, Bahía Blanca (8000), Argentina
ac@cs.uns.edu.ar

Tech. Report IRIT
RR- -2016- -03- -FR

2016

Abstract

This paper proposes a new framework able to take into account recursive interactions in bipolar abstract argumentation systems. We address issues such as “How an interaction can impact another one?”, or in other words “How can the validity of an interaction be affected if this interaction is attacked or supported by another one?”.

Thus, using numerous examples, a new method for flattening such recursive bipolar abstract argumentation systems (ASAF) using meta-arguments is proposed and compared with the original framework defined in [10] for taking into account an ASAF. This comparison highlights the similarities between both frameworks and gives interesting justifications of the choices given in [10].

Contents

1	Introduction	1
2	Bipolar abstract argumentation system	2
2.1	Abstract argumentation system (AS)	2
2.2	Abstract bipolar argumentation system (BAS)	3
2.2.1	Deductive supports	3
2.2.2	Necessary supports	4
2.2.3	Duality between deductive and necessary supports	5
2.2.4	Axiomatisation of a necessary BAS	6
2.2.5	A meta-framework encoding necessary support	7
3	Recursive interactions	8
3.1	AS with recursive interactions	8
3.2	BAS with recursive interactions	9
4	Encoding recursive interactions in MAS	11
4.1	Encoding basic attacks/supports	12
4.2	Encoding labelled interactions	13
4.3	Encoding recursive interactions	15
5	Comparison with ASAFA	21
6	Conclusion and future works	23

1 Introduction

The main feature of the argumentation framework is the ability to handle contradictory pieces of information, especially for reasoning [11; 1]. Moreover, argumentation can be used to formalize dialogues between several agents by modeling the exchange of arguments in, *e.g.*, negotiation between agents [3]. An argumentation system (AS) consists of a collection of arguments interacting with each other through a relation reflecting conflicts between them, called *attack*. The issue of argumentation is then to determine “acceptable” sets of arguments (*i.e.*, sets able to defend themselves collectively while avoiding internal attacks), called “*extensions*”, and thus to reach a coherent conclusion. Another form of analysis of an AS is the study of the particular status of each argument, this status is based on membership (or non-membership) of the extensions. Formal frameworks have greatly eased the modeling and study of an AS. In particular, the framework of [11] allows to completely abstract from the “concrete” meaning of the arguments and relies only on binary interactions that may exist between them.

In this paper, we are interested in a bipolar AS (BAS), that is an AS with a second kind of interaction, the support relation. This relation represents a positive interaction between arguments and has been first introduced in [12; 21]. In [6], the support relation is left general so that the obtained framework keeps a high level of abstraction. However there is no single interpretation of the support, and a number of researchers proposed specialized variants of the support relation (deductive support [5], necessary support [15; 16], evidential support [17; 18]). Each specialization can be associated with an appropriate modelling using an appropriate complex attack. These proposals have been developed quite independently, based on different intuitions and with different formalizations. In [8], a comparative study has been done in order to restate these proposals in a common setting, the bipolar argumentation framework. Basically, the idea is to keep the original arguments, to add complex attacks defined by the combination of the original attack and the support, and to modify the classical notions of acceptability. An important contribution of [8] is the highlight of a kind of duality between the deductive and the necessary interpretations of support, which results in a duality in the modelling by complex attacks. Following the same line, different papers have recently been written: some of them give a translation between necessary supports and evidential supports [19]; others propose a justification of the necessary support using the notion of subargument [20]; an axiomatisation of necessary support leading to different frameworks has been given in [9]; an extension of the necessary support is presented in [14]. [10] extends bipolar interactions by considering recursive attacks and supports, thus defining an *Attack-Support Argumentation Framework* (ASAF for short). In other words, an argument may attack (resp. support) another argument, an attack or a support.

[10] proposes to encode an ASAF by turning it into a BAS with necessary support, and then into an AS through the addition of extended attacks. As [9] presents different frameworks for encoding necessary support, it is interesting to enrich them with recursive interactions and then to draw a comparison with the ASAF approach.

This report is organized as follows: Some background is given in Section 2 for bipolar argumentation (with deductive and necessary support) and its axiomatisation. The background about recursive interactions is presented in Section 3. Then in Section 4 we show that the MAS framework proposed in [9] is suitable for taking into account recursive interactions. This MAS-based approach is compared with ASAF in Section 5. Finally, Section 6 concludes and suggests perspectives of our work.

2 Bipolar abstract argumentation system

Bipolar argumentation systems extend Dung's argumentation systems.

2.1 Abstract argumentation system (AS)

Dung's seminal abstract framework consists of a set of arguments and only one type of interaction between them, namely attack. What really means is the way arguments are in conflict.

Def. 1 (Dung AS) A Dung's argumentation system (AS, for short) is a pair $\langle \mathbf{A}, \mathbf{R} \rangle$ where \mathbf{A} is a finite and non-empty set of arguments and \mathbf{R} is a binary relation over \mathbf{A} (a subset of $\mathbf{A} \times \mathbf{A}$), called the attack relation.

An argumentation system can be represented by a directed graph denoted \mathcal{G} , called the *interaction graph*, in which the nodes represent arguments and the edges are defined by the attack relation: $\forall a, b \in \mathbf{A}, a\mathbf{R}b$ is represented by $a \rightarrow b$.

Def. 2 (Admissibility in AS) Given $\langle \mathbf{A}, \mathbf{R} \rangle$ and $S \subseteq \mathbf{A}$,

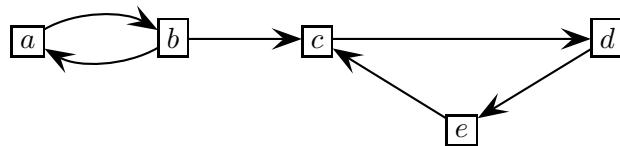
- S is conflict-free in $\langle \mathbf{A}, \mathbf{R} \rangle$ iff¹ there are no arguments $a, b \in S$, s.t.² $a\mathbf{R}b$.
- $a \in \mathbf{A}$ is acceptable in $\langle \mathbf{A}, \mathbf{R} \rangle$ wrt³ S iff $\forall b \in \mathbf{A}$ s.t. $b\mathbf{R}a$, $\exists c \in S$ s.t. $c\mathbf{R}b$.
- S is admissible in $\langle \mathbf{A}, \mathbf{R} \rangle$ iff S is conflict-free and each argument in S is acceptable wrt S .

Standard semantics introduced by Dung (preferred, stable, grounded) enable to characterize admissible sets of arguments that satisfy some form of optimality.

Def. 3 (Extensions) Given $\langle \mathbf{A}, \mathbf{R} \rangle$ and $S \subseteq \mathbf{A}$,

- S is a preferred extension of $\langle \mathbf{A}, \mathbf{R} \rangle$ iff it is a maximal (wrt \subseteq) admissible set.
- S is a stable extension of $\langle \mathbf{A}, \mathbf{R} \rangle$ iff it is conflict-free and for each $a \notin S$, there is $b \in S$ s.t. $b\mathbf{R}a$.
- S is the grounded extension of $\langle \mathbf{A}, \mathbf{R} \rangle$ iff it is the least (wrt \subseteq) admissible set X s.t. each argument acceptable wrt X belongs to X .

Ex. 1 Let AS be defined by $\mathbf{A} = \{a, b, c, d, e\}$ and $\mathbf{R} = \{(a, b), (b, a), (b, c), (c, d), (d, e), (e, c)\}$, and represented by the following graph. There are two preferred extensions ($\{a\}$ and $\{b, d\}$), one stable extension ($\{b, d\}$) and the grounded extension is the empty set.



The status of an argument is determined by its membership to the extensions of the selected semantics: e.g., an argument can be “skeptically accepted” (resp. “credulously”) if it belongs to all the extensions (resp. at least to one extension) and be “rejected” if it does not belong to any extension.

¹if and only if

²such that

³with respect to

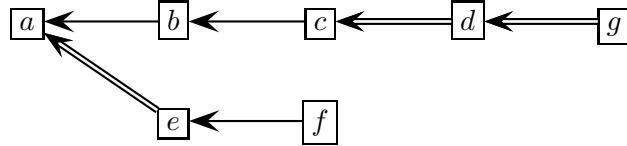
2.2 Abstract bipolar argumentation system (BAS)

The abstract bipolar argumentation framework presented in [6; 7] extends Dung's framework in order to take into account both negative interactions expressed by the attack relation and positive interactions expressed by a support relation (see [2] for a more general survey about bipolarity in argumentation).

Def. 4 (BAS) A bipolar argumentation system (BAS, for short) is a tuple $\langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ where \mathbf{A} is a finite and non-empty set of arguments, \mathbf{R}_{att} is a binary relation over \mathbf{A} called the attack relation and \mathbf{R}_{sup} is a binary relation over \mathbf{A} called the support relation.

A BAS can still be represented by a directed graph \mathcal{G}_b called the *bipolar interaction graph*, with two kinds of edges. Let a_i and $a_j \in \mathbf{A}$, $a_i \mathbf{R}_{\text{att}} a_j$ (resp. $a_i \mathbf{R}_{\text{sup}} a_j$) means that a_i attacks a_j (resp. a_i supports a_j) and it is represented by $a \rightarrow b$ (resp. by $a \Rightarrow b$).

Ex. 2 For instance, in the following graph representing a BAS, there is a support from g to d and an attack from b to a .



Handling support and attack at an abstract level has the advantage to keep genericity. An abstract bipolar framework is useful as an analytic tool for studying different notions of complex attacks, complex conflicts, and new semantics taking into account both kinds of interactions between arguments. However, the drawback is the lack of guidelines for choosing the appropriate definitions and semantics depending on the application. For solving this problem, some variants of the support relation have been proposed recently: the deductive support and the necessary support⁴.

2.2.1 Deductive supports

Among the different variants defined for taking into account a support between arguments, [5] proposed the notion of deductive support. This notion is intended to enforce the following constraint: If $b \mathbf{R}_{\text{sup}} c$ then the acceptance of b implies the acceptance of c , and as a consequence the non-acceptance of c implies the non-acceptance of b .

In order to compute semantics of a BAS, one of the main proposals is to translate the BAS into an AS expressing the new attacks due to the presence of supports. In the case of a deductive support, two kinds of attack can appear. The first one, called mediated attack, corresponds to the case when $b \mathbf{R}_{\text{sup}} c$ and $a \mathbf{R}_{\text{att}} c$: the acceptance of a implies the non-acceptance of c and so the non-acceptance of b .

Def. 5 ([5] Mediated attack)

Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. There is a mediated attack from a to b iff there is a sequence $a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_{n-1}$, and $a_n \mathbf{R}_{\text{att}} a_{n-1}$, $n \geq 3$, with $a_1 = b$, $a_n = a$. $\mathbf{M}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}}$ denotes the set of mediated attacks generated by \mathbf{R}_{sup} on \mathbf{R}_{att} .

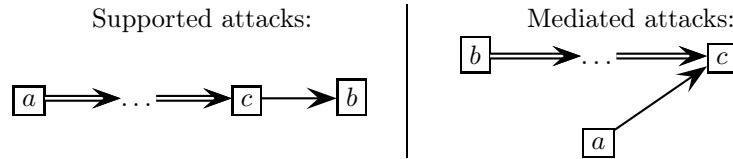
⁴A third one, the evidential support, has also been proposed in [17; 18; 19] but will not be discussed here.

Moreover, the deductive interpretation of the support justifies the introduction of another attack (called supported attacks in [7]): if $a\mathbf{R}_{\text{sup}}c$ and $c\mathbf{R}_{\text{att}}b$, the acceptance of a implies the acceptance of c and the acceptance of c implies the non-acceptance of b ; so, the acceptance of a implies the non-acceptance of b .

Def. 6 ([7] Supported attack)

Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. There is a supported attack from a to b iff there is a sequence $a_1\mathbf{R}_1 \dots \mathbf{R}_{n-1}a_n$, $n \geq 3$, with $a_1 = a$, $a_n = b$, $\forall i = 1 \dots n-2$, $\mathbf{R}_i = \mathbf{R}_{\text{sup}}$ and $\mathbf{R}_{n-1} = \mathbf{R}_{\text{att}}$. $\mathbf{S}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}}$ denotes the set of supported attacks generated by \mathbf{R}_{sup} on \mathbf{R}_{att} .

So, with the deductive interpretation of the support, new kinds of attack, from a to b , can be considered in the following cases:



Def. 7 The AS defined by $\langle \mathbf{A}, \mathbf{R}_{\text{att}} \cup \mathbf{M}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}} \cup \mathbf{S}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}} \rangle$ is called the associated Dung AS for the deductive support of BAS and denoted by AS^D .

From Definitions 5 and 6, new attacks called **d+-attacks** can be generated inductively as follows:⁵

Def. 8 ([8], d+-attacks) Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ with \mathbf{R}_{sup} being a set of deductive supports. There exists a d+-attack from a to b iff

- either $a\mathbf{R}_{\text{att}}b$, or $a\mathbf{S}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}}b$, or $a\mathbf{M}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}}b$ (**Basic case**),
- or there exists an argument c s.t. a supports c and c d+-attacks b (**Case 1**),
- or there exists an argument c s.t. a d+-attacks c and b supports c (**Case 2**).

$\mathbf{D}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}}$ denoted the set of d+-attacks generated by \mathbf{R}_{sup} on \mathbf{R}_{att} .

The AS defined by $\langle \mathbf{A}, \mathbf{D}_{\mathbf{R}_{\text{att}}}^{\mathbf{R}_{\text{sup}}} \rangle$ is called the complete associated Dung AS for the deductive support of BAS and denoted by AS^{Dc} .

BAS has been turned into a Dung's argumentation system $\text{AS}^6(\text{AS}^{Dc})$, in which the classical semantics can be considered.

2.2.2 Necessary supports

Necessary support has been initially⁷ proposed in [15; 16] with the following interpretation: If $c\mathbf{R}_{\text{sup}}b$ then the acceptance of c is necessary to get the acceptance of b , or equivalently the acceptance of b implies the acceptance of c .

⁵Our notation is different than the one used in [8]. We have modified it in order to homogenize with the notation of necessary attacks defined in [9].

⁶See in [8], some properties of AS^{Dc} .

⁷An extension of this work is presented in [14]. In this new version the support version relies a *set of arguments* to an argument (whereas, in the previous version the support relation was a binary relation between *two arguments*). In this context, the meaning of a support is not exactly the same: If $\{a_1, \dots, a_n\}\mathbf{R}_{\text{sup}}b$ then the acceptance of b implies the acceptance of at least one argument of $\{a_1, \dots, a_n\}$. This extension of the necessary support is not taken into account in the current paper.

Suppose now that $a \mathbf{R}_{\text{att}} c$. The acceptance of a implies the non-acceptance of c and so the non-acceptance of b . This constraint can be taken into account by introducing a new attack, called secondary attack in [7] and extended attack in [15].

Moreover, another kind of complex attack can be justified: If $c \mathbf{R}_{\text{sup}} a$ and $c \mathbf{R}_{\text{att}} b$, the acceptance of a implies the acceptance of c and the acceptance of c implies the non-acceptance of b . So, the acceptance of a implies the non-acceptance of b . This constraint relating a and b should be enforced by adding a new complex attack from a to b proposed in [16].

The formal definition of these two complex attacks is:

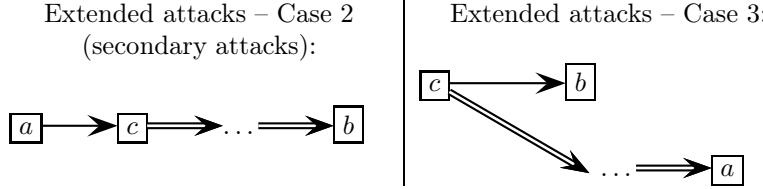
Def. 9 ([16] Extended attack) Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. There is an extended attack from a to b iff

1. either $a \mathbf{R}_{\text{att}} b$,
2. or there is a sequence $a_1 \mathbf{R}_{\text{att}} a_2 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n$, $n \geq 3$, with $a_1 = a$, $a_n = b$,
3. or there is a sequence $a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n$, and $a_1 \mathbf{R}_{\text{att}} a_p$, $n \geq 2$, with $a_n = a$, $a_p = b$.

The set of the extended attacks will be denoted by $\mathbf{R}_{\text{att}}^{\text{ext}}$.

The AS defined by $\langle \mathbf{A}, \mathbf{R}_{\text{att}}^{\text{ext}} \rangle$ is called the associated Dung AS for the necessary support of BAS and denoted by AS^N .

So, with the necessary interpretation of the support, new kinds of attack, from a to b , can be considered in the following cases:



2.2.3 Duality between deductive and necessary supports

In this section, we will use the following notation:

Nota. 1 Deductive (resp. necessary) support will be called d-support (resp. n-support) and the existence of a d-support (resp. n-support) between two arguments a and b will be denoted by $a \xrightarrow{D} b$ (resp. $a \xrightarrow{N} b$).

Deductive support and necessary support have been introduced independently. Nevertheless, they correspond to dual interpretations of the support in the following sense: $a \xrightarrow{N} b$ means that the acceptance of a is necessary to get the acceptance of b , and $a \xrightarrow{D} b$ means that the acceptance of a implies the acceptance of b . So $a \xrightarrow{N} b$ is equivalent to $b \xrightarrow{D} a$.

In [8], this duality has been used to show another kind of duality between mediated attacks and secondary attacks: the mediated attacks obtained by combining the attack relation \mathbf{R}_{att} and the support relation \mathbf{R}_{sup} exactly correspond to the secondary attacks obtained by combining the attack relation \mathbf{R}_{att} and the support relation $\mathbf{R}_{\text{sup}}^{-1}$ which is the symmetric relation of \mathbf{R}_{sup} ($\mathbf{R}_{\text{sup}}^{-1} = \{(b, a) | (a, b) \in \mathbf{R}_{\text{sup}}\}$). Similarly, the supported attacks obtained by combining the attack relation \mathbf{R}_{att} and the support relation \mathbf{R}_{sup} exactly correspond to the the third case of extended attack (Definition 9) obtained by combining the attack relation \mathbf{R}_{att} and the support relation $\mathbf{R}_{\text{sup}}^{-1}$.

Nota. 2 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ with \mathbf{R}_{sup} being a set of n-supports.

- BAS_{sym} denotes the bipolar framework defined by $\langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}}^{-1} \rangle$ ($\mathbf{R}_{\text{sup}}^{-1}$ is a set of d-supports).

- $\text{AF}_{\text{sym}}^{\text{Dc}}$ denotes the complete associated Dung AS for BAS_{sym} (obtained using the d+-attacks issued from BAS_{sym}).
- The complete associated Dung AS for the necessary support, denoted by AS^{Nc} , exactly corresponds to $\text{AF}_{\text{sym}}^{\text{Dc}}$.⁸

Using the above notations, Table 1, issued from [8], gives a synthetic view of the correspondences between the two approaches (deductive and necessary).

Deductive supports of [5] \mathbf{R}_{sup} is a d-support	Necessary supports of [15; 16] $\mathbf{R}_{\text{sup}}^{-1}$ is a n-support
supported attack	extended attack (case 3) with $\mathbf{R}_{\text{sup}}^{-1}$
mediated attack	extended attack (case 2) with $\mathbf{R}_{\text{sup}}^{-1}$
AS^{Dc}	AS^{Nc} for $\mathbf{R}_{\text{sup}}^{-1}$

Table 1: Correspondences between deductive and necessary supports

2.2.4 Axiomatisation of a necessary BAS

In [9], an axiomatic approach for handling necessary support has been proposed. Four constraints have been defined describing the desired behavior of a BAS with necessary support, and different frameworks suitable for encoding these constraints have been studied.

Transitivity (TRA) This first requirement concerns the relation \mathbf{R}_{sup} alone. It expresses transitivity of the necessary support. It is defined as:

Def. 10 (Constraint TRA [9]) $\forall a, b \in \mathbf{A}$, if $\exists n > 1$ such that $a = a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n = b$, then a supports b .

Closure (CLO) A second constraint also concerns the relation \mathbf{R}_{sup} alone and expresses the fact that if $c \mathbf{R}_{\text{sup}} b$, then “the acceptance of b implies the acceptance of c ”. So, if $c \mathbf{R}_{\text{sup}} b$, and there exists an extension S containing b , then S also contains c . This constraint can be expressed by the property of closure of an extension under $\mathbf{R}_{\text{sup}}^{-1}$.

Def. 11 (Constraint CLO [9]) Let s be a semantics and E be an extension under s . $\forall a, b \in \mathbf{A}$, if $a \mathbf{R}_{\text{sup}} b$ and $b \in E$, then $a \in E$.

Conflicting sets (CFS) Now, we consider constraints induced by the presence of both attacks and supports in a BAS. Starting from the original interpretation, if $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$, “the acceptance of a implies the non-acceptance of c ” and “the acceptance of b implies the acceptance of c ”. So, using contrapositives, “the acceptance of a implies the non-acceptance of b ”, and then “the acceptance of b implies the non-acceptance of a ”. Thus, we obtain a symmetric constraint involving a and b . However, the fact that “the acceptance of a implies the non-acceptance of b ” is not equivalent to the fact that there is an attack from a to b . We have only the sufficient condition. So, the creation of a complex attack (here a secondary attack) from a to b can be viewed in some sense too strong. Hence, faced with the case when $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$, we propose to assert a conflict between a and b , or in other words that the set $\{a, b\}$ is a conflicting set. Similarly, if $c \mathbf{R}_{\text{att}} b$ and $c \mathbf{R}_{\text{sup}} a$,

⁸ AS^{Nc} could be defined inductively from BAS and the extended attacks as done in Definition 8 for AS^{Dc} .

“the acceptance of a implies the acceptance of c ” and so “the acceptance of a implies the non-acceptance of b ”.

Def. 12 (Constraint CFS [9]) $\forall a, b, c \in \mathbf{A}$. If $(a \mathbf{R}_{\text{att}} c \text{ and } c \text{ supports } b)$ or $(c \mathbf{R}_{\text{att}} b \text{ and } c \text{ supports } a)$ then $\{a, b\}$ is a conflicting set.

Addition of new attacks (nATT, n+ATT) According to the applications and the previous works presented in literature, we may impose stronger constraints corresponding to the addition of new attacks. Two cases may be considered:

Def. 13 (Constraint nATT [9]) If $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$, then there is a new attack from a to b .

Def. 14 (Constraint n+ATT [9]) If $(a \mathbf{R}_{\text{att}} c \text{ and } c \mathbf{R}_{\text{sup}} b)$ or $(c \mathbf{R}_{\text{att}} b \text{ and } c \mathbf{R}_{\text{sup}} a)$, then there is a new attack from a to b .

nATT (resp. n+ATT) corresponds to the addition of secondary (resp. extended) attacks.

2.2.5 A meta-framework encoding necessary support

In [9], several frameworks have been proposed for handling necessary supports. In this report we focus on the framework which encodes the following interpretation of necessary support: If $c \mathbf{R}_{\text{sup}} b$, “the acceptance of c is necessary to get the acceptance of b ” because c is the *only* attacker of a particular attacker of b . So it is assumed that there exists a special argument attacking b for which c is the *only* attacker. More precisely, if $c \mathbf{R}_{\text{sup}} b$, we create a new argument N_{cb} (a meta-argument) and two attacks $c \mathbf{R}_{\text{att}} N_{cb}$ and $N_{cb} \mathbf{R}_{\text{att}} b$. As c is the unique attacker of N_{cb} , “the acceptance of b implies the acceptance of c ”. N_{cb} is the attacker of b which justifies the presence of c . A similar idea can be found in [22; 10] for the more general purpose of representing recursive and defeasible attacks and supports.

Encoding a necessary support with a meta-argument leads to the definition of a special system: the meta-argumentation system, called the MAS.

Def. 15 (The MAS associated with a BAS) Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ with \mathbf{R}_{sup} being a set of necessary supports. Let $\mathbf{A}_n = \{N_{cb} | (c, b) \in \mathbf{R}_{\text{sup}}\}$ and $\mathbf{R}_n = \{(c, N_{cb}) | (c, b) \in \mathbf{R}_{\text{sup}}\} \cup \{(N_{cb}, b) | (c, b) \in \mathbf{R}_{\text{sup}}\}$. The tuple $\text{MAS} = \langle \mathbf{A} \cup \mathbf{A}_n, \mathbf{R}_{\text{att}} \cup \mathbf{R}_n \rangle$ is the meta-argumentation system⁹ associated with BAS .

Let us check whether the minimal requirements are satisfied. Let us first consider constraint **TRA**. From $a \mathbf{R}_{\text{sup}} b$ and $b \mathbf{R}_{\text{sup}} c$, we obtain the sequence of attacks $a \mathbf{R}_{\text{att}} N_{ab} \mathbf{R}_{\text{att}} b \mathbf{R}_{\text{att}} N_{bc} \mathbf{R}_{\text{att}} c$. So, the acceptance of c implies the acceptance of b , which in turn implies the acceptance of a , as if we had directly encoded $a \mathbf{R}_{\text{sup}} c$. So, **TRA** is taken into account. The same result holds for **CLO**:

Prop. 1 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ and its associated MAS. Let $S \subseteq \mathbf{A} \cup \mathbf{A}_n$. If S is admissible in MAS , then $S \cap \mathbf{A}$ is closed under $\mathbf{R}_{\text{sup}}^{-1}$ in BAS .

Constraint **CFS** is not enforced. We only have the following property:

Prop. 2 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ and its associated MAS. Let a, b, c be arguments of \mathbf{A} . If $(a \mathbf{R}_{\text{att}} c \text{ and } c \text{ supports } b)$ or $(c \mathbf{R}_{\text{att}} b \text{ and } c \text{ supports } a)$ then no admissible set in MAS contains $\{a, b\}$.

⁹Note that it is an argumentation system in Dung’s sense.

Note that this result is weaker than **CFS** since it does not imply that $\{a, b\}$ is a conflicting set.

Obviously, stronger constraints such as **nATT** or **n+ATT** are not directly enforced. If $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$, we obtain the sequence $a \mathbf{R}_{\text{att}} c \mathbf{R}_{\text{att}} N_{cb} \mathbf{R}_{\text{att}} b$. No attack from a to b is added. However, in [9], it is shown that the meta-argumentation system associated with **BAS** enables to recover the extensions obtained when enforcing Constraint **nATT**.

3 Recursive interactions

The idea of recursive interactions has been introduced in [13] and developed in [4] for recursive attacks and in [10] for recursive supports plus attacks.

The purpose is to express the fact that the validity of an interaction may depend on another interaction (for instance because of preferences as in [13]).

3.1 AS with recursive interactions

In [4], recursive attacks are considered. An attack is defined recursively as a pair where the first part is an argument and the second part is an argument (basic case) or another attack.

Def. 16 (AFRA) *An Argumentation Framework with Recursive Attacks (AFRA) is a pair $\langle \mathbf{A}, \mathbf{R} \rangle$ where:*

- \mathbf{A} is a set of arguments,
- \mathbf{R} is a subset of $\mathbf{A} \times (\mathbf{A} \cup \mathbf{R})$. \mathbf{R} is a set of attacks, each attack being defined either between two arguments of \mathbf{A} , or between an argument of \mathbf{A} and an attack of \mathbf{R} .

Note that, in order to reason with these recursive attacks, it is mandatory to name them. Moreover, given an attack $\alpha = (a, X)$, a is called the source of α and X is called the target of α . The notion of *defeat* is defined as follows:

Def. 17 (Defeat in AFRA) *Let $\text{AFRA} = \langle \mathbf{A}, \mathbf{R} \rangle$. Let $\alpha, \beta \in \mathbf{R}$. Let $X \in \mathbf{A} \cup \mathbf{R}$.*

- α directly defeats X iff X is the target of α .
- α indirectly defeats β iff the target of α is an argument that is the source of β .

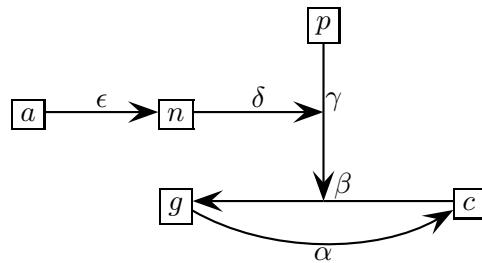
Then in [4], a translation of an AFRA into an AS is provided:

Def. 18 (AS associated with an AFRA) *Let $\text{AFRA} = \langle \mathbf{A}, \mathbf{R} \rangle$. The AS associated with AFRA is $\text{AS} = \langle \mathbf{A}', \mathbf{R}' \rangle$ defined by:*

- $\mathbf{A}' = \mathbf{A} \cup \mathbf{R}$,
- $\mathbf{R}' = \{(X, Y) \text{ s.t. } X \in \mathbf{R}, Y \in \mathbf{A} \cup \mathbf{R} \text{ and } X \text{ directly or indirectly defeats } Y\}$.

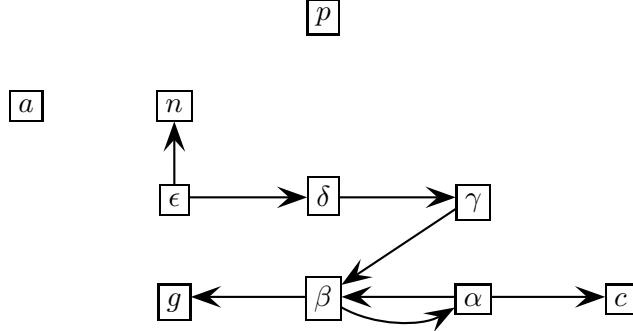
The previous notions are illustrated on the following example:

Ex. 3 Consider the AFRA represented by:



For instance, ϵ directly defeats n and indirectly defeats δ .

The AS associated with AFRA is:



The following points seem counterintuitive:

- there is no attack between a and n (more generally, no argument from \mathbf{A} can be an attacker in the associated AS of the AFRA),
- there is no link between a and ϵ (more generally, there is no link between an attack and its source); that is surprising since, without a , the attack ϵ does not exist.

3.2 BAS with recursive interactions

In [10], AFRA has been extended in order to handle supports, with the necessary interpretation. In that case, supports as well as attacks can be recursive.

Def. 19 (ASAF) An Attack-Support Argumentation Framework (ASAF) is a triple $\langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ where:

- \mathbf{A} is a set of arguments,
- \mathbf{R}_{att} is a subset of $\mathbf{A} \times (\mathbf{A} \cup \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$. \mathbf{R}_{att} is a set of attacks, each attack being defined either between two arguments of \mathbf{A} , or between an argument of \mathbf{A} and an attack of \mathbf{R}_{att} , or else between an argument of \mathbf{A} and a support of \mathbf{R}_{sup} .
- \mathbf{R}_{sup} is a subset of $\mathbf{A} \times (\mathbf{A} \cup \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$. \mathbf{R}_{sup} is a set of necessary supports, each support being defined either between two arguments of \mathbf{A} , or between an argument of \mathbf{A} and an attack of \mathbf{R}_{att} , or else between an argument of \mathbf{A} and a support of \mathbf{R}_{sup} . Note that \mathbf{R}_{sup} is assumed to be irreflexive and transitive.

We assume that $\mathbf{R}_{\text{att}} \cap \mathbf{R}_{\text{sup}} = \emptyset$.

As in the AFRA approach, a translation of an ASAF into an AS is given in [10]. This translation follows a two-steps process (see Def. 22): first, the ASAF is turned into a necessary BAS (see Def. 20), then this BAS is turned into an AS (see Def. 21) through the addition of extended attacks.

For the first step, the idea is to encode an attack $\alpha = (a, c)$, a, c being arguments, by a meta-argument α which interacts with a and c in the following way: the acceptance of the meta-argument α means that the attack α is “active” and as a is necessary for the attack it originates, there will be a necessary support from a to α . Then the fact that α defeats c is encoded by a simple attack from the meta-argument α to the argument c . So the attack $\alpha = (a, c)$ is encoded by $a \Rightarrow \alpha \rightarrow c$.

In the case of a support $\beta = (b, c)$, b, c being arguments, two meta-arguments β^+ and β^- are introduced with the following meaning: “ β^+ active” means that c is accepted (and so b is also

accepted) and “ β^- active” means that c is not accepted. So the support $\beta = (b, c)$ is encoded by $b \Rightarrow \beta^+ \rightarrow \beta^- \rightarrow c$.

The formal definition of the BAS associated with ASAFT is given below:

Def. 20 (BAS associated with ASAFT) Let $\text{ASAFT} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. The BAS associated with ASAFT is the triple $\text{BAS}' = \langle \mathbf{A}', \mathbf{R}_{\text{att}}', \mathbf{R}_{\text{sup}}' \rangle$ such that:

$$\begin{aligned} \mathbf{A}' = & \mathbf{A} \\ & \cup \{ \alpha | \alpha = (a, x) \in \mathbf{R}_{\text{att}} \} \\ & \cup \{ \beta^-, \beta^+ | \beta = (b, y) \in \mathbf{R}_{\text{sup}} \} \end{aligned}$$

$$\begin{aligned} \mathbf{R}_{\text{att}}' = & \{ (\alpha, x) | \alpha = (a, x) \in \mathbf{R}_{\text{att}} \text{ and } x \in \mathbf{A} \cup \mathbf{R}_{\text{att}} \} \\ & \cup \{ (\beta^+, \beta^-), (\beta^-, y) | \beta = (b, y) \in \mathbf{R}_{\text{sup}} \text{ and } y \in \mathbf{A} \cup \mathbf{R}_{\text{att}} \} \\ & \cup \{ (\alpha, \beta^+), (\alpha, \beta^-) | \alpha = (a, \beta) \in \mathbf{R}_{\text{att}} \text{ and } \beta \in \mathbf{R}_{\text{sup}} \} \\ & \cup \{ (\beta^+, \beta^-), (\beta^-, \gamma^+), (\beta^-, \gamma^-) | \beta = (b, \gamma) \in \mathbf{R}_{\text{sup}} \text{ and } \gamma \in \mathbf{R}_{\text{sup}} \} \end{aligned}$$

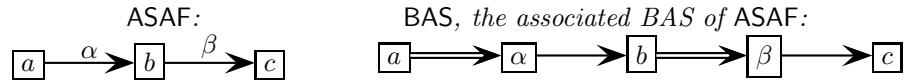
$$\begin{aligned} \mathbf{R}_{\text{sup}}' = & \{ (a, \alpha) | \alpha = (a, x) \in \mathbf{R}_{\text{att}} \text{ and } x \in \mathbf{A} \cup \mathbf{R}_{\text{att}} \} \\ & \cup \{ (b, \beta^+) | \beta = (b, y) \in \mathbf{R}_{\text{sup}} \text{ and } y \in \mathbf{A} \cup \mathbf{R}_{\text{att}} \} \\ & \cup \{ (a, \alpha) | \alpha = (a, \beta) \in \mathbf{R}_{\text{att}} \text{ and } \beta \in \mathbf{R}_{\text{sup}} \} \\ & \cup \{ (b, \beta^+) | \beta = (b, \gamma) \in \mathbf{R}_{\text{sup}} \text{ and } \gamma \in \mathbf{R}_{\text{sup}} \} \end{aligned}$$

Note that \mathbf{R}_{att} (resp. \mathbf{R}_{sup}) is not included in \mathbf{R}_{att}' (resp. \mathbf{R}_{sup}'). Nevertheless, due to the introduction of a necessary support, a link between an attack (or a support) and its source exists (this addresses one of the issues pointed in AFRA).

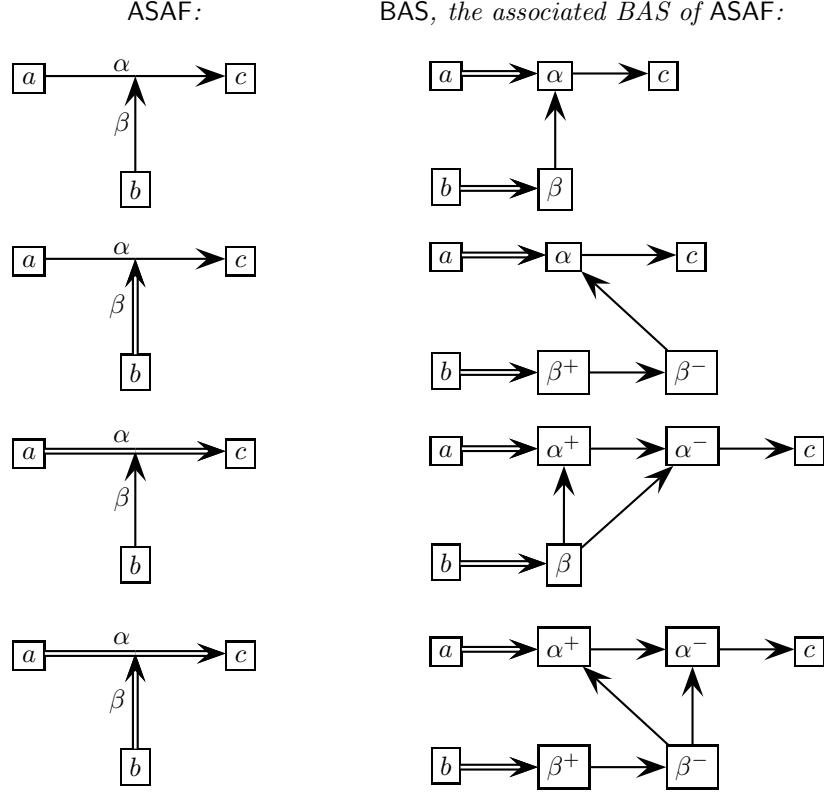
Note also that the attacks and supports of \mathbf{R}_{att}' and \mathbf{R}_{sup}' are “simple” interactions, i.e. pairs of elements of \mathbf{A}' which are not labelled with a symbol.

The following examples illustrate different cases:

Ex. 4



Ex. 5 Consider different ASAFT with recursive interactions and their associated BAS. We successively consider an attack which is attacked, an attack which is supported, a support which is attacked and a support which is supported.



After turning ASAFA into a BAS with necessary supports, the second step is to create an AS. The approach followed in [10] is to encode supports by adding complex attacks, namely the secondary attacks (or case 2 - extended attacks, see Definition 9). Formally:

Def. 21 (AS associated with BAS) Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a necessary BAS. Its associated AS is the pair $\text{AS}' = \langle \mathbf{A}', \mathbf{R}' \rangle$ such that

- $\mathbf{A}' = \mathbf{A}$,
- $\mathbf{R}' = \mathbf{R}_{\text{att}} \cup \{(a, b) | \text{ there is a sequence } a_1 \mathbf{R}_{\text{att}} a_2 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n, n \geq 3, \text{ with } a_1 = a, a_n = b\}$

Note that the AS obtained using Def. 21 is included in the AS obtained with Def. 9, since Def. 9 (corresponding to Constraint **n+ATT**) is more general than Def. 21 (corresponding only to Constraint **nATT**). Nevertheless, in terms of acceptability, the results are the same (see [9]).

Def. 22 (AS associated with ASAFA) Let $\text{ASAFA} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. The AS associated with ASAFA is the associated AS of the associated BAS of ASAFA.

For summarizing, for taking into account recursive interactions (attacks and supports), the “ASAFA approach” proposes a translation of an ASAFA into a BAS followed by a translation of this BAS into an AS.

4 Encoding recursive interactions in MAS

In Section 2.2.5, we have recalled the MAS framework proposed in [9], which enables to handle necessary support through the introduction of meta-arguments. In this section, we propose to use MAS as an alternative to ASAFA approach for encoding both supports and recursive interactions. More precisely, we will address the following issues:

- distinguishing between labelled and unlabelled interactions, *i.e.* distinguishing between interactions that may be involved in a recursion (either as a target, or as targeting another interaction) and the other interactions;
- encoding labelled interactions, *i.e.* encoding the ability to reason on the interactions;
- encoding recursive interactions, *i.e.* encoding the impact of an interaction on another interaction.

For that purpose, we need to formalize the notion of labelled interaction. So we propose a slightly modified version of the ASAFAF, which we call the labelled ASAFAF.

Def. 23 (Labelled ASAFAF) A labelled ASAFAF is a 5-uple $\langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}}, \mathcal{V}, \mathcal{L} \rangle$ where:

- \mathbf{A} is a set of arguments,
- $\mathbf{R}_{\text{att}} : \mathbf{A} \rightarrow (\mathbf{A} \cup \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ is a set of attacks,
- $\mathbf{R}_{\text{sup}} : \mathbf{A} \rightarrow (\mathbf{A} \cup \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ is a set of necessary supports,
- \mathcal{V} is a set of labels (denoted in this paper by greek letters),
- \mathcal{L} is a bijection from $\mathbf{R} \subseteq (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ to \mathcal{V} ($|\mathbf{R}| = |\mathcal{V}|$).

Moreover we assume that $\mathbf{R}_{\text{att}} \cap \mathbf{R}_{\text{sup}} = \emptyset$.

Following this definition, an interaction (attack or necessary support) can be defined either between two arguments of \mathbf{A} , or between an argument of \mathbf{A} and an attack of \mathbf{R}_{att} , or between an argument of \mathbf{A} and a support of \mathbf{R}_{sup} .¹⁰

Moreover the above definition allows for labelled interactions as well as unlabelled interactions. That will enable to distinguish those interactions which are not involved in a recursive interaction (either as a target, or as targeting another interaction). These interactions may be considered as always “valid” and will be called “basic” in the following. Since the aim of a label is to be able to reason about the interactions and to encode recursive interactions, no label is required for naming these basic interactions.¹¹ Moreover, note that each label corresponds to a unique labelled interaction (attack or necessary support) and vice versa.

The main difference between Def. 23 and Def. 19 is the fact that we explicitly integrate the labels into the definition of an ASAFAF in order to reason about the interactions.

In order to simplify the notations, a labelled interaction will be confused with its label. So considering $\alpha \in \mathcal{V}$ and $(x, y) \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ such that α is the label of (x, y) , we will write by abuse of language:

- “ $\alpha = (x, y)$ ” in place of “ $\alpha = \mathcal{L}(x, y)$ ”, and
- “ $\alpha \in \mathbf{R}_{\text{att}}$ ” (resp. \mathbf{R}_{sup}) in place of “ $\alpha = \mathcal{L}(x, y)$ and $(x, y) \in \mathbf{R}_{\text{att}}$ (resp. \mathbf{R}_{sup})”.

Following this line, \mathcal{V} will be identified with the set of labelled interactions.

In order to define the MAS associated with a labelled ASAFAF, encodings of components of this ASAFAF are given in the next sections:

- for unlabelled interactions, see Section 4.1,
- for labelled interactions, see Section 4.2,
- for recursive interactions, see Section 4.3.

Then, in Section 5, our proposal will be compared with ASAFAF approach.

4.1 Encoding basic attacks/supports

Such interactions correspond to unlabelled interactions and can be directly encoded using the MAS given in [9] (corresponding to Def. 15):

¹⁰If \mathbf{R}_{att} and \mathbf{R}_{sup} are defined from \mathbf{A} to \mathbf{A} , the labelled ASAFAF is reduced to a labelled BAS.

¹¹If all interactions are unlabelled ($\mathcal{V} = \emptyset$), the labelled ASAFAF is reduced to a simple BAS.

Def. 24 (Unlabelled interactions in a MAS) Let $\text{LASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}}, \mathcal{V}, \mathcal{L} \rangle$ be a labelled ASAF. Let MAS be the MAS associated with LASAF . Let $a, b \in \mathbf{A}$. The following schemas describe the encoding of an unlabelled attack (resp. support) between two arguments:

- the attack $a \rightarrow b$ in LASAF remains $a \rightarrow b$ in MAS and
- the support $a \Rightarrow b$ in LASAF is turned into $a \rightarrow N_{ab} \rightarrow b$ in MAS .

As said in section 2.2.5, N_{ab} codes the existence of an attacker on b that justifies the presence of a .

4.2 Encoding labelled interactions

Consider an interaction between two arguments, which might be an attacked or supported interaction. Since we want to reason about this interaction, and refer to it, this interaction must be labelled and its label will be used as a “meta-argument”.

It seems to us that a labelled interaction $\alpha = (a, b)$ encompasses two types of links.

- One link relates α to b , representing the role of α (either an attack to b or a support to b). It will be called the *effect-link* in the following for clarity.
- The other link relates α to its source, representing the grounding of α . This link from α to its source a will be called a *ground-link* in the following. The idea of “grounded” interaction is close to the notion of evidential argumentation in the work of [17; 18; 19]. It means that “an interaction makes sense only if its source argument is accepted”.

These two links suggest two kinds of validity for the interaction.

- We propose to reserve the term *validity* for the first kind of link. For instance in case of a graph containing only α attacked by β , α is not valid. Another situation would be the case of α requiring a support $\beta = (c, \alpha)$ with c being attacked by d . In that case, the interaction α is not valid.
- Concerning the “ground-link”, we use the term *grounded*. For instance, $\alpha = (a, b)$ is not grounded if a is attacked and not defended. Note that a support interaction can be valid even though its source is not accepted.

So an interaction may be for instance valid and not grounded, or grounded and not valid. We propose to call *active* an interaction which is both valid and grounded. Intuitively, if α is only attacked by an interaction which is not active (whatever the origin of this non-activity), then α should be valid. If α is necessarily supported by an interaction β which is valid but not grounded, then α should not be valid. Moreover, if β is not valid, the validity of α cannot be affected by β (even if β is not grounded). The above notions are synthetized in the following table and should be further formalized.

Let $\alpha = (a, b)$ be a labelled interaction

Type of link	Meaning of the link	Corresponding Notion
effect-link	describes the role of α wrt b (is affected by interactions on α)	validity
ground-link	describes the existence of α wrt a (takes into account only the source of α)	groundness

The interaction α is said active iff it is grounded and valid

The ground-link is a necessary support between the meta-argument and the source argument: any interaction α from a to b is “grounded” only if a is accepted (there is a necessary support between a and α). This support is basic since it is not defeasible. Moreover, if α labels an attack from a to b , we create a basic attack from α to b , and if α labels a support from a to b , we create a basic support from α to b .

Note that we encode a labelled interaction α with basic interactions, since an attack (or a support) to α will be encoded by attacks (supports) to the meta-arguments that are introduced.

So, the labelled attack $\alpha = (a, b)$ is encoded by $a \Rightarrow \alpha \rightarrow b$ and the labelled support $\beta = (b, c)$ is encoded by $b \Rightarrow \beta \Rightarrow c$.

Formally, we can define the BAS associated with a labelled ASAF as follows:

Def. 25 (Associated BAS of a labelled ASAF) Let $\text{LASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}}, \mathcal{V}, \mathcal{L} \rangle$ be a labelled ASAF. The BAS associated with LASAF is defined by $\text{BAS} = \langle \mathbf{A}', \mathbf{R}_{\text{att}}', \mathbf{R}_{\text{sup}}' \rangle$ where:

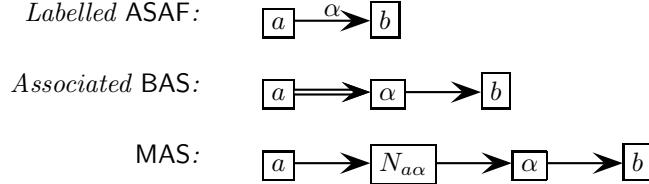
- $\mathbf{A}' = \mathbf{A} \cup \{\alpha | \alpha \in \mathbf{R}_{\text{sup}} \cup \mathbf{R}_{\text{att}}\}$,¹²
- $\mathbf{R}_{\text{att}}' = \{(a, b) | (a, b) \in \mathbf{R}_{\text{att}} \text{ and } (a, b) \notin \mathcal{V}\}$ ¹³
 $\quad \cup \{(\alpha, b) | \alpha = (a, b) \in \mathbf{R}_{\text{att}}\}$
- $\mathbf{R}_{\text{sup}}' = \{(a, b) | (a, b) \in \mathbf{R}_{\text{sup}} \text{ and } (a, b) \notin \mathcal{V}\}$
 $\quad \cup \{(a, \alpha) | \alpha = (a, b) \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}\} \cup \{(\beta, b) | \beta = (a, b) \in \mathbf{R}_{\text{sup}}\}$

Unlabelled interactions between two arguments remain unchanged in the associated BAS since we need neither to reason about them, nor to use them for recursive interactions (so they do not introduce new arguments or new interactions).

If no recursive interaction appears in the labelled ASAF, Def. 15 can be applied to its associated BAS and the supports can be encoded by the introduction of other kinds of meta-arguments and their associated attacks. So we obtain a MAS associated with a labelled ASAF without recursive interaction.

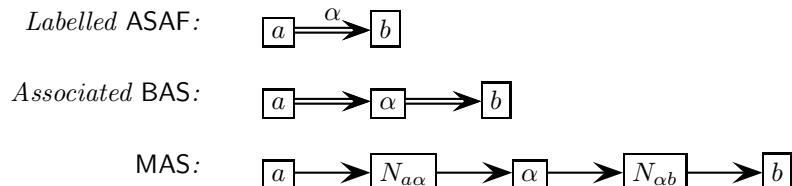
The following examples illustrate the construction of such a MAS.

Ex. 6 Encoding a labelled attack $\alpha = (a, b)$:



$N_{a\alpha}$ codes the ground-link.

Ex. 7 Encoding a labelled support $\alpha = (a, b)$:



$N_{a\alpha}$ codes the ground-link and $N_{\alpha b}$ codes the effect-link, here a support link.

Note that if an attack $\alpha = (a, b)$ is active then a and b cannot belong to the same extension. Identically if a support $\alpha = (a, b)$ is active then if b is accepted then a must be also accepted.

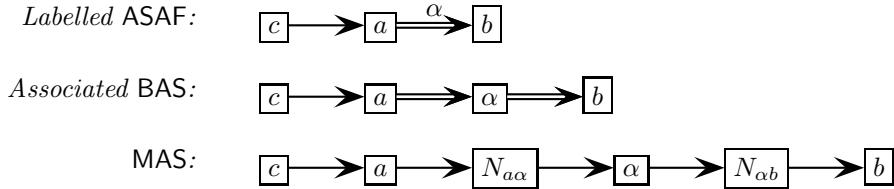
¹²Recall that $\alpha \in \mathbf{R}_{\text{sup}} \cup \mathbf{R}_{\text{att}}$ means $\exists (x, y) \in \mathbf{R}_{\text{sup}} \cup \mathbf{R}_{\text{att}}$ such that $\alpha = \mathcal{L}(x, y)$.

¹³Recall that $(a, b) \notin \mathcal{V}$ means $\nexists \alpha \in \mathcal{V}$ s.t. $\alpha = \mathcal{L}(a, b)$.

To sum up, a labelled interaction has been encoded in two steps: First, a meta-argument is introduced with a basic support from its source and then the basic supports are encoded in MAS as recalled in Section 2.2.5.

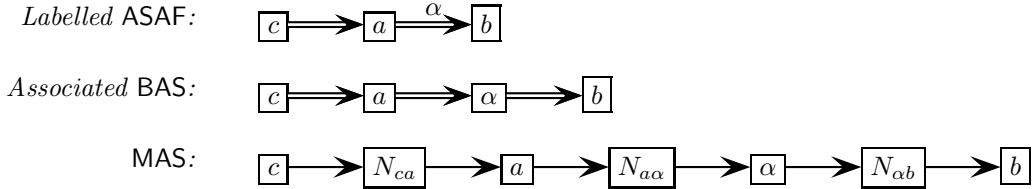
In the next examples, we illustrate the case of a labelled ASAFA without recursive interaction but containing labelled as well as unlabelled interactions.

Ex. 8 Encoding a labelled support $\alpha = (a, b)$ and a basic attack:



α is not grounded but it is valid (since there exists no interaction that can affect the validity of α).

Ex. 9 Encoding a labelled support $\alpha = (a, b)$ and a basic support:



Nevertheless, labels take on their full meaning in case of recursive interactions. So in the next section, we propose an encoding of such interactions.

4.3 Encoding recursive interactions

Following the methodology of ASAFA, our purpose is to represent an attack (resp. a support) on a labelled interaction by attacks (resp. supports) on the meta-arguments associated with this labelled interaction. However, all the meta-arguments do not play the same role and a deeper analysis is needed in order to identify the meta-arguments which will be affected by the recursive interaction.

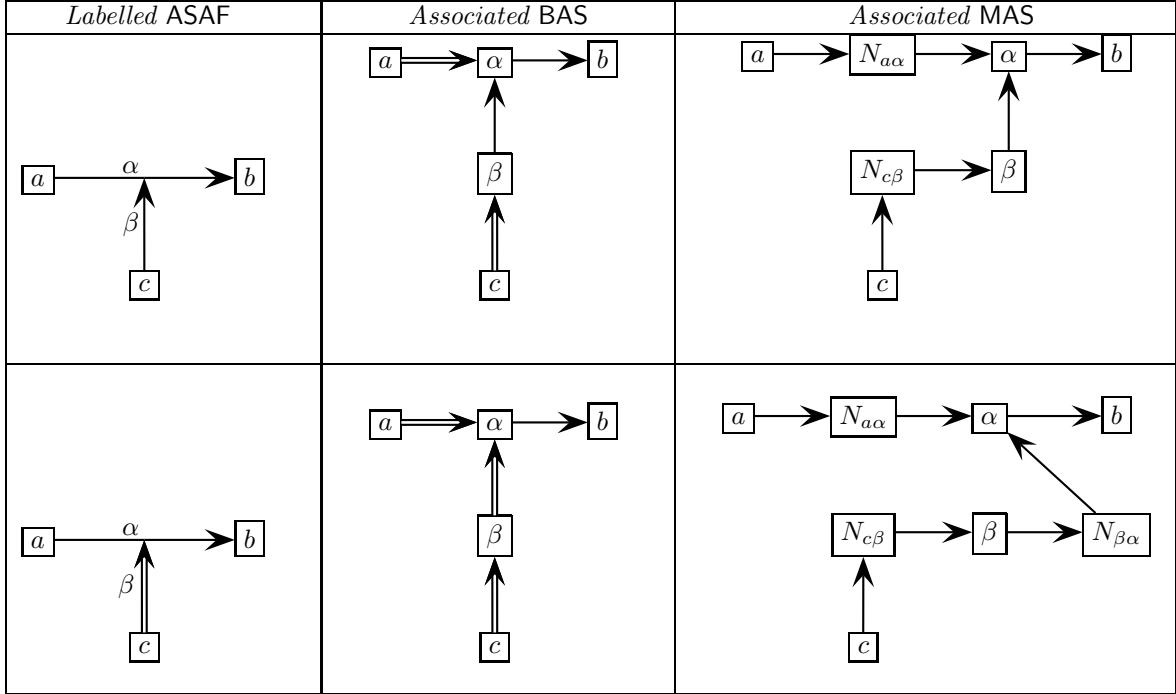
We successively consider the case of attacked (or supported) attacks and the case of attacked (or supported) supports.

Case of attacked or supported attacks Let $\alpha = (a, b)$ a labelled attack and $\beta = (c, \alpha)$ an interaction on α . In this case, it is clear that encoding the interaction β will produce an interaction on the meta-argument associated with α . More precisely, we want to enforce the following constraints:

- If α is attacked by β and β is active, then α is not valid and a and b could belong to the same extension; note that, if β is not valid, and β is the only interaction that might impact the validity of α , then α is valid (so a and b cannot belong to the same extension).
- if α is supported by β and β is valid, since this support is a necessary one, then α is valid only if β is grounded (this implies that c is accepted). This is equivalent to say that if α is supported by β , β being valid and not grounded, then α cannot be valid.

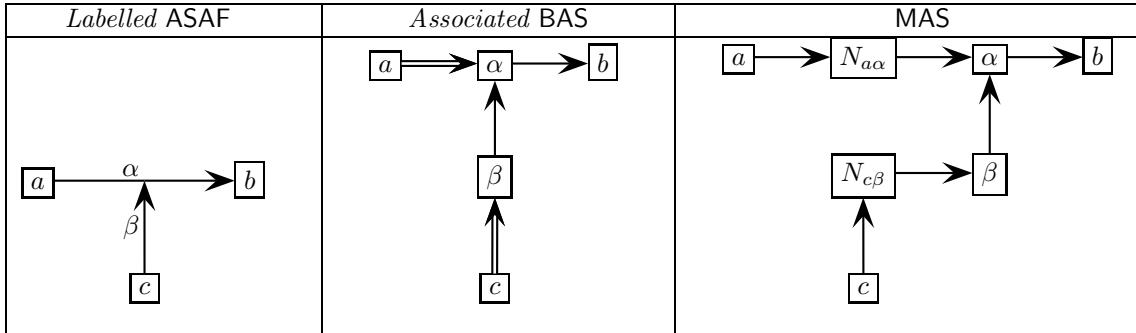
So we obtain the following translation of a labelled ASAFA with an attacked (resp. supported) attack into a MAS:

Def. 26 (Attacked or supported attacks in MAS) The following schemas describe the encoding of an attacked (resp. supported) attack in a MAS.



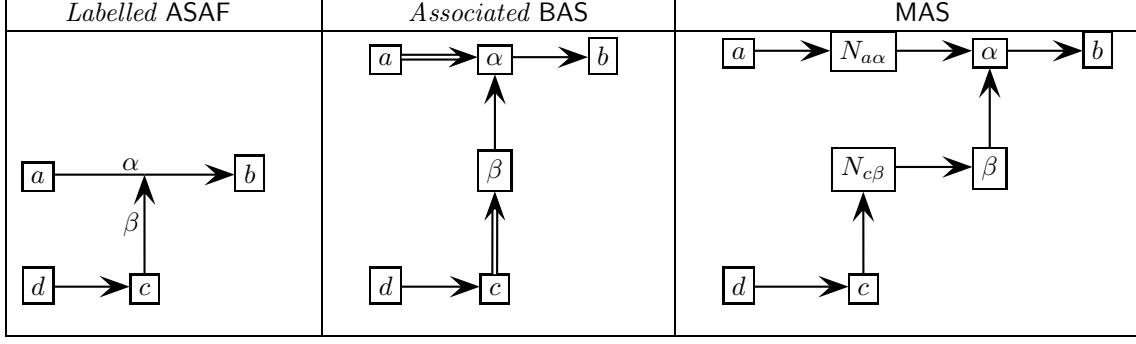
The following examples illustrate the construction of MAS presented the above.

Ex. 10 Let $\alpha = (a, b)$ and $\beta = (c, \alpha)$ two labelled attacks.



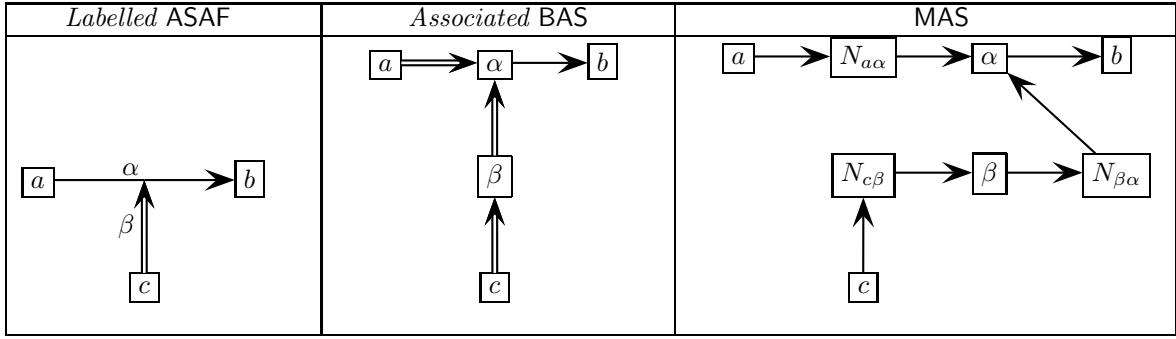
In this case, the preferred extension of MAS is $\{a, c, \beta, b\}$. So a and c are accepted, β is active, α is not active (it is grounded but not valid) and b can be accepted (it is defended by β and c in MAS). Note that this result always holds whatever the status of a .

Consider now that c is attacked by d (note that it is useless to label the attack from d to c since it is not attacked or supported by another interaction and it does not attack or support another interaction; we consider this attack as a basic attack).



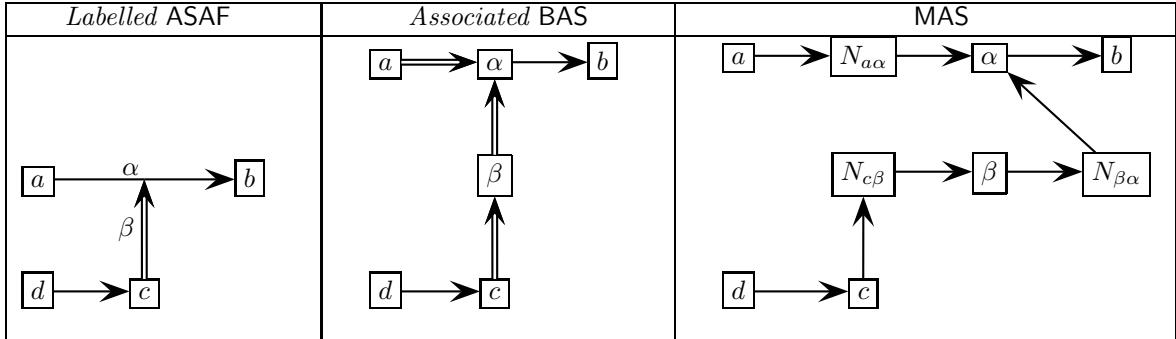
In this case, the preferred extension of MAS is $\{a, d, N_{c\beta}, \alpha\}$, which corresponds to the set $\{a, d, \alpha\}$ after removing the meta-argument $N_{c\beta}$. This set could be considered as the extension of ASAFA¹⁴. c is not accepted, and so β is not grounded. This implies that α can be valid. In consequence, a and b cannot belong to the same extension. In this case, b will be accepted only if a is attacked.

Ex. 11 Let $\alpha = (a, b)$ a labelled attack and $\beta = (c, \alpha)$ a labelled support.



In this case, the preferred extension of MAS is $\{a, c, \beta, \alpha\}$. a and c are accepted, β and α are active (valid and grounded) and in this case b cannot be accepted.

Consider now that c is attacked by d (with a basic attack).



In this case, the preferred extension of MAS is $\{a, d, N_{c\beta}, N_{\beta\alpha}, b\}$ which corresponds to the set $\{a, d, b\}$ after removing the meta-arguments $N_{c\beta}$ and $N_{\beta\alpha}$. c is not accepted, and so β is not grounded. Furthermore, since β is valid, it implies that α is not valid and thus not active. In consequence, a and b can belong to the same extension. Once again, b is defended by d , $N_{c\beta}$ and $N_{\beta\alpha}$, whatever the status of a .

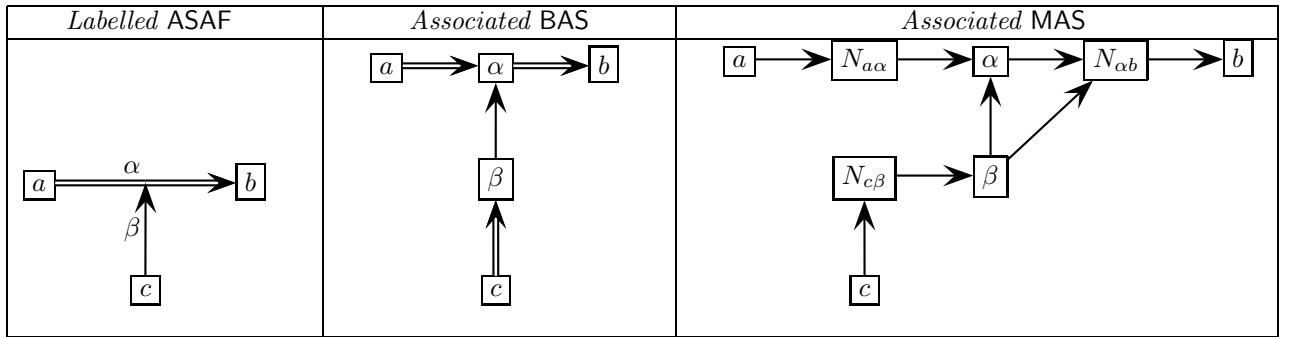
¹⁴Since this paper reports only a preliminary study, the expected outcomes of our framework are not yet defined. Nevertheless, we could consider that the extensions of the original labelled ASAFA are obtained by filtering the extensions of MAS. Indeed, only the meta-arguments associated to the labelled interactions make sense in the extensions of the labelled ASAFA.

Case of attacked or supported supports Let $\alpha = (a, b)$ a labelled support and $\beta = (c, \alpha)$ an interaction on α . In this case, encoding the interaction α on its target b produces the meta-argument $N_{\alpha b}$. We have to determine which of these meta-arguments will be affected by the interaction β . Let us consider different examples for that purpose.

Let us first consider the case when the labelled support $\alpha = (a, b)$ is attacked by $\beta = (c, \alpha)$.

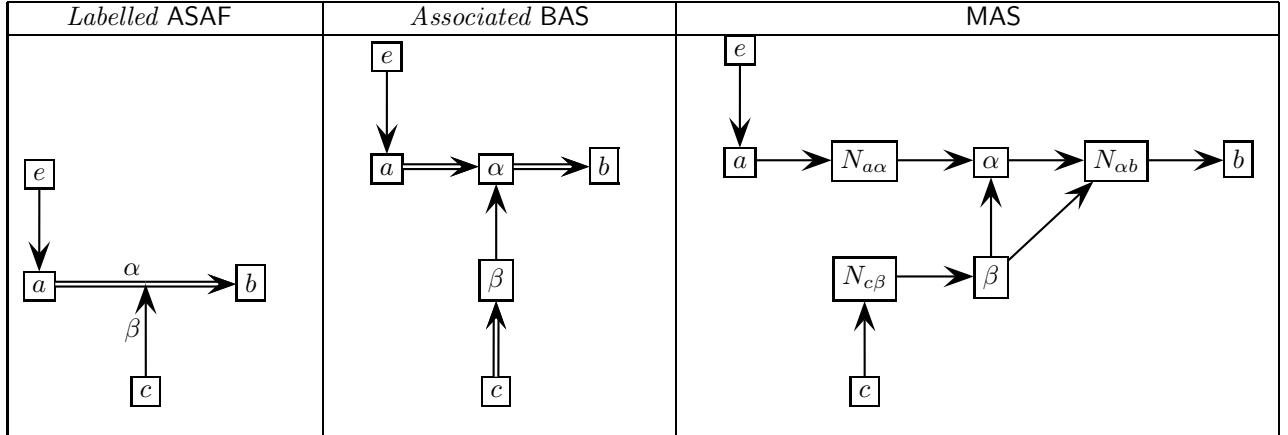
Ex. 12

- If β is active, then α is not valid. This justifies the existence of an attack from the meta-argument β to the meta-argument $N_{\alpha b}$. Moreover, α would not be active, justifying the existence of an attack from β to α in the associated MAS.
- If β is not active, for instance in situations where c would be attacked, and if β is the only interaction that impacts α , then α is valid and b is accepted implies that a is accepted (see Ex. 13.3).



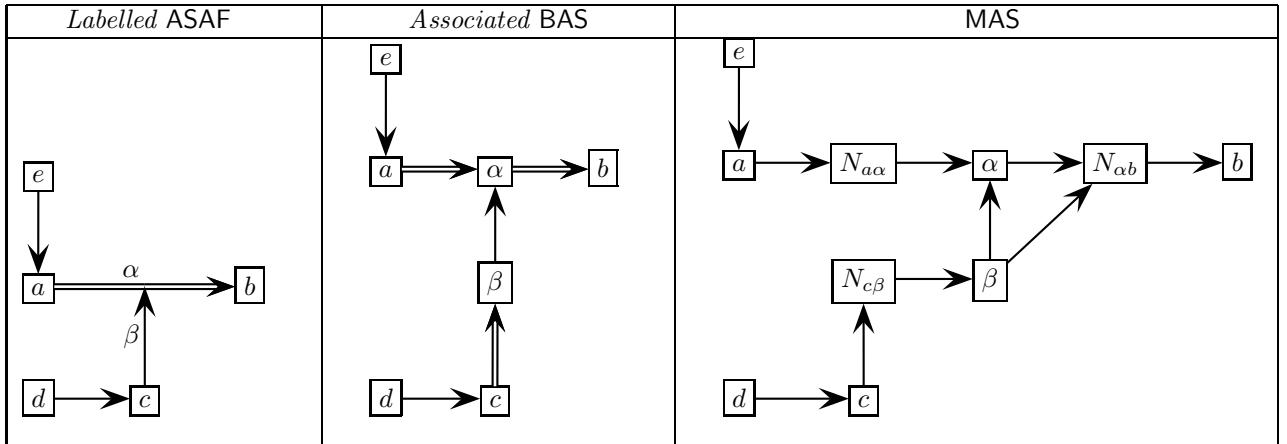
Ex. 13

1. Consider the case when a is attacked:



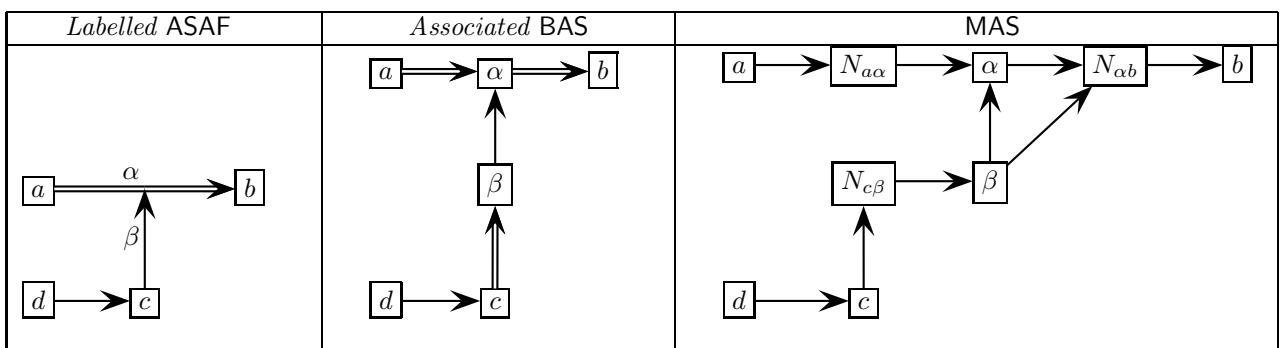
In this case, the preferred extension is $\{e, c, N_{a\alpha}, \beta, b\}$ which corresponds to the set $\{e, c, \beta, b\}$. c is accepted, and so β is grounded. Moreover, β is valid and so active. This implies that the support α from a to b is not valid, and thus not active. b being not attacked, it can be accepted even if a is not accepted (this is due to the attack from β to $N_{\alpha b}$). Note that the presence of $N_{a\alpha}$ in the extension means that α is not grounded.

2. Now, consider the case when a and c are attacked.



In this case, the preferred extension is $\{e, d, N_{a\alpha}, N_{c\beta}, N_{ab}\}$, which corresponds to the set $\{e, d\}$. c is not accepted, and so β is not grounded. Thus β is not active, implying that the support α from a to b is valid. Nevertheless, α is not grounded since a is not accepted. Therefore α is not active and b cannot be accepted.

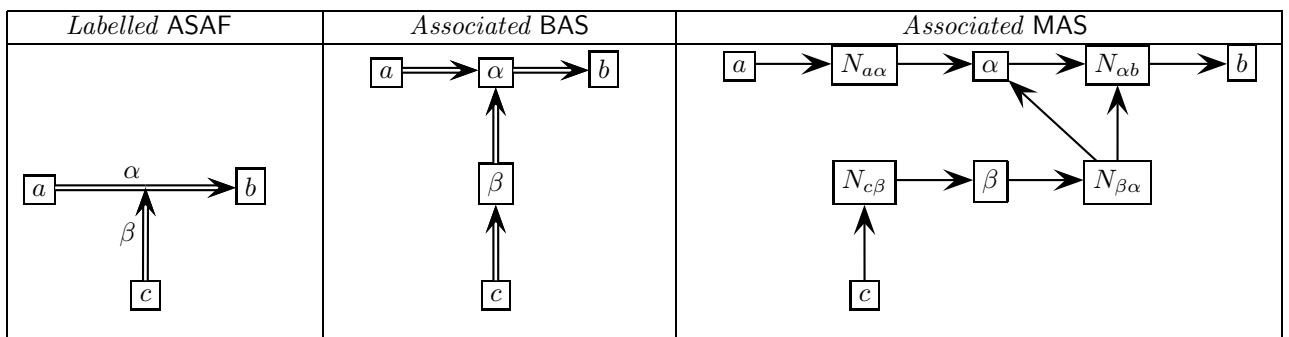
3. Lastly, consider that a is not attacked.



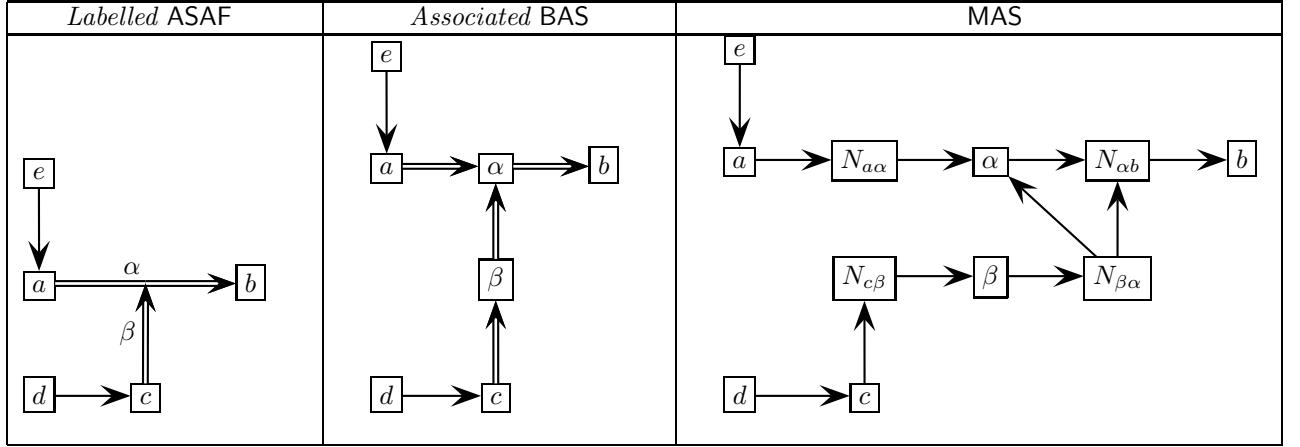
In this case, the preferred extension is $\{a, d, \alpha, N_{c\beta}, b\}$, which corresponds to the set $\{a, d, \alpha, b\}$. c is not accepted, and so β is not grounded. Thus β is not active, implying that the support α from a to b is valid. Moreover, α is grounded. So α is active and a and b are accepted together. One can say that d has defended α .

Now we consider the case when the support $\alpha = (a, b)$ is supported by $\beta = (c, \alpha)$.

Ex. 14 If β is valid and not grounded, then α is not valid. This justifies the existence of an attack from the meta-argument $N_{\beta\alpha}$ to the meta-argument N_{ab} . Moreover, α would not be active, justifying the existence of an attack from $N_{\beta\alpha}$ to α in the associated MAS.



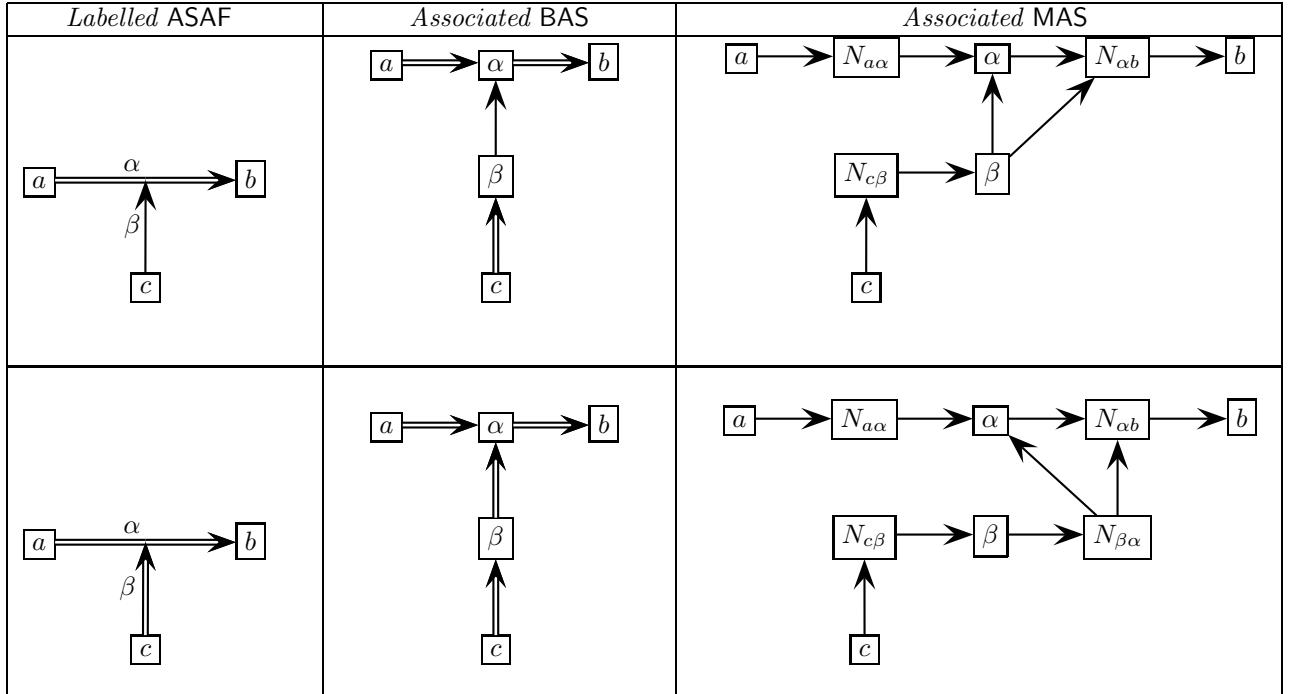
Ex. 15 Consider the case when a and c are attacked:



In this case, the preferred extension is $\{e, d, N_{a\alpha}, N_{c\beta}, N_{\beta\alpha}, b\}$, which corresponds to the set $\{e, d, b\}$. c is not accepted, and so β is not grounded and so not active. Moreover, since β is valid, it implies that the support α from a to b is not valid (thus not active). b is not attacked, so it can be accepted even if a is not accepted (this is due to the attack from $N_{\beta\alpha}$ to N_{ab}).

To sum up, we obtain the following translation of a labelled ASAFA with an attacked (resp. supported) support into a MAS:

Def. 27 (Attacked or supported supports in MAS) The following schemas describe the encoding of an attacked (resp. supported) support in a MAS.



We can summarize with the following schema: Given a support $\alpha = (a, b)$, and an extension E , four cases may occur:

- $\alpha \in E$: it means that α is active, or equivalently grounded and valid (of course in that case $N_{a\alpha} \notin E$ and $N_{ab} \notin E$);
- $\alpha \notin E$ and $N_{a\alpha} \notin E$: it means that α is grounded but not active; so it is not valid;
- $\alpha \notin E$ and $N_{a\alpha} \in E$ and $N_{ab} \in E$: it means that α is not active, not grounded and α is valid;¹⁵
- $\alpha \notin E$ and $N_{a\alpha} \in E$ and $N_{ab} \notin E$: it means that α is not active, not grounded and not valid.¹⁶

Given an attack $\alpha = (a, b)$, and an extension E , analogous links can be established, according to the three following cases:

- $\alpha \in E$: it means that α is active, or equivalently grounded and valid (of course in that case $N_{a\alpha} \notin E$);
- $\alpha \notin E$ and $N_{a\alpha} \notin E$: it means that α is grounded but not active; so it is not valid;
- $\alpha \notin E$ and $N_{a\alpha} \in E$: it means that α is not active and not grounded. However, it is not sufficient for deciding the validity of α . We need to consider supporters (resp. attackers) of α present in E .

5 Comparison with ASAFT

In this section, we compare ASAFT approach and MAS approach for encoding labelled and recursive interactions. Both approaches follow two steps. The first step produces a BAS. However, the BAS obtained by both approaches are different in case of supports. Moreover the second step is quite different.

Let us first consider the case of an attack $\alpha = (a, b)$

Prop. 3 *Let $\alpha = (a, b)$ be a labelled attack. The translation of α using Def. 20 is exactly the same that the one given by Def. 25: $a \xrightarrow{\alpha} b$ becomes $a \Rightarrow \alpha \rightarrow b$ where α denotes a meta-argument associated with the attack (a, b) .*

In the case of a support, things are different. Let $\alpha = (a, b)$ be a labelled support.

- The first step of ASAFT approach follows Def. 20: $a \xrightarrow{\alpha} b$ becomes $a \Rightarrow \alpha^+ \rightarrow \alpha^- \rightarrow b$. Two meta-arguments are created for representing the interaction α .
- The first step of MAS approach follows Def. 25: $a \xrightarrow{\alpha} b$ becomes $a \Rightarrow \alpha \Rightarrow b$. Only one meta-argument is created for representing the interaction α .
- However, during the second step of MAS approach, encoding the support $\alpha \Rightarrow b$ will produce $\alpha \rightarrow N_{ab} \rightarrow b$. So α (resp. N_{ab}) in the MAS plays the role of α^+ (resp. α^-) in the associated BAS of the ASAFT.

Indeed, the main difference lies in the encoding of the ground-link.

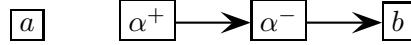
The second step produces different AS.

- In ASAFT approach, the remaining supports are handled by the addition of extended attacks. So $a \Rightarrow \alpha^+ \rightarrow \alpha^- \rightarrow b$ will be turned into (according to Def. 21 no extended attack can be added):

¹⁵Indeed, in the MAS, N_{ab} is attacked by α . Moreover, if α is the target of β in ASAFT, then, in the MAS, N_{ab} is also attacked by each attacker x related to β (*i.e.* $x = \beta$ or $N_{\beta\alpha}$ depending on the type of β); so if $N_{ab} \in E$ then

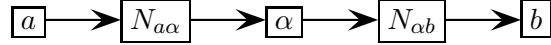
- α is valid (in ASAFT, either no β attacks α , or each attacker β of α is not active) and
- N_{ab} is defended against α (so $N_{a\alpha} \in E$, that is α is not grounded).

¹⁶Since $N_{a\alpha} \in E$, it defends N_{ab} against α . Nevertheless, since $N_{ab} \notin E$ then N_{ab} is attacked by β (or by $N_{\beta\alpha}$ depending on the type of β) and not defended and it is the same thing for α .



The preferred extension is $\{a, \alpha^+, b\}$.

- In the MAS approach, both supports are handled by creating meta-arguments (according to Def. 15). So $a \Rightarrow \alpha \Rightarrow b$ will be turned into:



The preferred extension is $\{a, \alpha, b\}$.

The following example describes the case where an extended attack can be added with the ASAFAF approach in order to obtain the associated AS.

Ex. 16 Consider the labelled ASAFAF represented by $c \xrightarrow{\beta} a \Rightarrow b$.

- With the ASAFAF approach, the associated BAS corresponding to Def. 20 is

$$c \Rightarrow \beta \rightarrow a \Rightarrow \alpha^+ \rightarrow \alpha^- \rightarrow b$$

According to Def. 21, an extended attack is introduced between β and α^+ and so the resulting AS is:



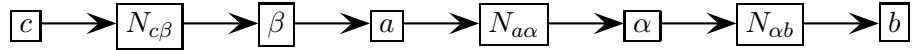
If we consider the preferred semantics, $\{c, \beta, \alpha^-\}$ is the only preferred extension.

- With the MAS approach, two cases are possible:

- either we consider that the interactions are labelled in order to reason about them (even if they are not involved in recursive interactions); in this case, the associated BAS corresponding to Def. 25 is

$$c \Rightarrow \beta \rightarrow a \Rightarrow \alpha \Rightarrow b$$

The resulting AS according to Section 4 is:



If we consider the preferred semantics, $\{c, \beta, N_{aα}, N_{ab}\}$ is the only preferred extension, which corresponds to the extension of the BAS $\{c, \beta\}$. Note that the presence of β in the extension means that the interaction β is active.

- or we consider that the labels are useless since we are not interested in reasoning about the interactions (particularly because they are not involved in recursive interactions); in that case, we can directly apply Def. 15 and obtain a simpler system:



In this case, the preferred extension is $\{c, N_{ab}\}$, which corresponds to the extension of the BAS $\{c\}$.

In the ASAFAF approach, all the supports are not taken into account in the same way. Given the labelled support $\alpha = (a, b)$, the support from the source (a) to α is handled with extended attacks, while the support from α to the target (b) is handled by the introduction of two meta-arguments. In contrast, the MAS approach provides a unified handling of all the supports, and gives also a clear meaning for the meta-arguments α^+ (resp. α^-) introduced in the ASAFAF approach.

Moreover, since in the ASAFAF approach any interaction is considered as labelled, the resulting AS can be uselessly complex.

Regarding the groundness of interactions, both ASAFAF and MAS approaches represent the ground-link by incorporating in the BAS a necessary support between the source of an interaction and a meta-argument associated with it. In particular, as mentioned before, by doing this the

ASAF approach addresses one of the issues pointed out in the AFRA. However, it could be noted that this advantage is only temporary. That is, whereas the BAS associated with an ASAF has a support link between an interaction and its source, the resulting AS has no direct link between them. This is because, as expressed before, the second step in the transformation of an ASAF into its associated AS handles this support by adding extended attacks. In contrast, in the MAS approach, given a labelled interaction $\alpha = (a, b)$, a and α are directly linked by a sequence of attacks going through the meta-argument $N_{\alpha a}$.

Another interesting point is the difference between the meaning associated with the presence of an interaction in an extension of the ASAF or the MAS, according to the terminology proposed in the paper. In the MAS, any interaction (either an attack or a support) involved in a recursion will be present in an extension only if it is active (thus, grounded and valid). In the ASAF, the presence of an attack or a support in an extension can have different meanings.

On the one hand, the presence of an attack in an extension of the ASAF means, similarly to the MAS, that the attack is active. Nevertheless, there exists a difference between the treatment of attacks in both approaches, since the MAS represents the groundness and validity of an attack α by two different meta-arguments, and uses a third meta-argument for expressing the fact that the attack is active. In contrast, the ASAF approach condenses both the groundness and validity of the attack through the meta-argument α which, in turn, represents the fact that the attack is active. Therefore, the MAS approach is more flexible, since the different meta-arguments related to a given attack handle these features separately, enabling one to easily identify whether the attack is active, grounded and/or valid.

On the other hand, the presence of a support α in an extension of the ASAF may have different meanings: that the support is active (in which case α^+ belongs to the extension of its associated AS); or that the support is valid and not grounded (in which case α^- belongs to the extension of the associated AS). Therefore, the presence of a support α in an extension of the ASAF allows one to certainly conclude that α is valid. However, the fact that α does not belong to an extension of the ASAF does not necessarily imply that the support is not valid. That is, it can either mean that the support is not valid, or that the support is valid but there is no sufficient information in order to decide whether it is grounded or not. A situation like the last one can occur, for instance, when considering the grounded semantics and a support α whose source is involved in an attack cycle. As a result, the MAS approach provides a uniform treatment for the meaning associated with the interactions belonging to its extensions, in contrast with the ASAF.

6 Conclusion and future works

This paper reports a preliminary study of a new framework that can be used for taking into account recursive interactions in bipolar abstract argumentation systems. Our aim was to address issues such as “How an interaction can impact another one?”, or in other words “How can the validity of an interaction be affected if this interaction is attacked or supported by another one?”.

In order to answer these questions, numerous examples have been studied in this paper. These examples have enabled us to identify different kinds of validity in presence of recursive interactions (namely the notions of “grounded interaction”, “valid interaction”, “active interaction”). Then, we have proposed and justified a new method for flattening recursive bipolar abstract argumentation systems using several kinds meta-arguments.

And we have initiated a comparison with the original framework defined in [10] intended for taking into account an ASAF. This comparison highlights the similarities between both frameworks and gives interesting justifications of the choices given in [10].

Our study has been essentially carried out from examples. So it opens several lines for further research:

- give a formal proof of our intuitions about the interpretation of the meta-arguments,
- define formally the expected outcomes of our framework and
- compare more deeply our proposal with the existing works, particularly in term of outcomes.

References

- [1] L. Amgoud and C. Cayrol. A reasoning model based on the production of acceptable arguments. *Annals of Mathematics and Artificial Intelligence*, 34:197–216, 2002.
- [2] L. Amgoud, C. Cayrol, M-C. Lagasquie-Schiex, and P. Livet. On bipolarity in argumentation frameworks. *Intl. Journal of Intelligent Systems*, 23:1062–1093, 2008.
- [3] L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proc. of ICMAS*, pages 31–38, 2000.
- [4] P. Baroni, F. Cerutti, M. Giacomin, and G. Guida. AFRA: Argumentation framework with recursive attacks. *Intl. Journal of Approximate Reasoning*, 52:19–37, 2011.
- [5] G. Boella, D. M. Gabbay, L. van der Torre, and S. Villata. Support in abstract argumentation. In *Proc. of COMMA*, pages 111–122. IOS Press, 2010.
- [6] C. Cayrol and M-C. Lagasquie-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Proc. of ECSQARU*, pages 378–389, 2005.
- [7] C. Cayrol and M-C. Lagasquie-Schiex. Coalitions of arguments: a tool for handling bipolar argumentation frameworks. *Intl. J. of Intelligent Systems*, 25:83–109, 2010.
- [8] C. Cayrol and M-C. Lagasquie-Schiex. Bipolarity in argumentation graphs: towards a better understanding. *IJAR*, 54(7):876–899, 2013.
- [9] C. Cayrol and M-C. Lagasquie-Schiex. An axiomatic approach to support in argumentation. In *Proc. of TAFA (LNAI 9524)*, pages 74–91, 2015.
- [10] A. Cohen, S. Gottifredi, A. J. García, and G. R. Simari. An approach to abstract argumentation with recursive attack and support. *Journal of Applied Logic*, 13(4):509–533, 2015.
- [11] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.
- [12] N. Karacapilidis and D. Papadias. Computer supported argumentation and collaborative decision making: the HERMES system. *Information systems*, 26(4):259–277, 2001.
- [13] S. Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173:901–934, 2009.
- [14] F. Nouioua. AFs with necessities: further semantics and labelling characterization. In *Proc. of SUM*, 2013.
- [15] F. Nouioua and V. Risch. Bipolar argumentation frameworks with specialized supports. In *Proc. of ICTAI*, pages 215–218. IEEE Computer Society, 2010.

- [16] F. Nouioua and V. Risch. Argumentation frameworks with necessities. In *Proc. of SUM*, pages 163–176. Springer-Verlag, 2011.
- [17] N. Oren and T. J. Norman. Semantics for evidence-based argumentation. In *Proc. of COMMA*, pages 276–284, 2008.
- [18] N. Oren, C. Reed, and M. Luck. Moving between argumentation frameworks. In *Proc. of COMMA*, pages 379–390. IOS Press, 2010.
- [19] S. Polberg and N. Oren. Revisiting support in abstract argumentation systems. In *Proc. of COMMA*, pages 369–376. IOS Press, 2014.
- [20] H. Prakken. On support relations in abstract argumentation as abstraction of inferential relations. In *Proc. of ECAI*, pages 735–740, 2014.
- [21] B. Verheij. Deflog: on the logical interpretation of *prima facie* justified assumptions. *Journal of Logic in Computation*, 13:319–346, 2003.
- [22] S. Villata, G. Boella, D. M. Gabbay, and L. van der Torre. Modelling defeasible and prioritized support in bipolar argumentation. *AMAI*, 66(1-4):163–197, 2012.