

A logical vision of abstract argumentation systems
with bipolar and recursive interactions

Claudette Cayrol,
M-Christine Lagasque-Schiex
Luis Farinas del Cerro

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

Tech. Report IRIT
RR- -2016- -02- -FR

2016

Abstract

This work is a preliminary study that proposes logical encodings for translating argumentation graphs *themselves* into logical knowledge bases. This translation will be used for identifying or redefining some properties of argumentation graphs. The graphs we consider are used to formalize abstract argumentation with at least two different kinds of interaction (attack and support) and also recursive interactions.

Contents

1	Introduction	1
2	Background of abstract argumentation systems	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	5
2.2.3	Duality between deductive and necessary supports	5
2.2.4	Axiomatisation of a necessary BAS	6
3	Recursive interactions	7
3.1	AS with recursive interactions	7
3.2	BAS with recursive interactions	9
4	Logics in biology	11
5	A logical representation of an argumentation graph	15
5.1	The case of a bipolar argumentation graph	15
5.2	The case of a recursive bipolar argumentation graph (ASAF)	20
6	Some comparison results	25
6.1	Logical representation of BAS vs axiomatisation	25
6.2	Comparing two logical representations of ASAF	26
7	Conclusion and future works	27

1 Introduction

The main feature of argumentation framework is the ability to deal with incomplete and / or contradictory information, especially for reasoning [16; 2]. Moreover, argumentation can be used to formalize dialogues between several agents by modeling the exchange of arguments in, *e.g.*, negotiation between agents [4]. 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 AS. In particular, the framework of [16] allows to completely abstract the “concrete” meaning of the arguments and relies only on binary interactions that may exist between them.

Many different extensions exist. For instance, bipolar AS (BAS) correspond to AS with a second kind of interaction, the support relation. This relation represents a positive interaction between arguments and has been first introduced by [17; 27]. In [8], the support relation is left general so that the bipolar 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 [6], necessary support [21; 22], evidential support [23; 24]). 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 [10], 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 result of [10] 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. In this context, new different papers have recently been written: some of them give a translation between necessary supports and evidential supports [25]; others propose a justification of the necessary support using the notion of subarguments [26]; an extension of the necessary support is presented in [20].

Another important extensions are AS that take into account interactions on interactions. A first version has been introduced by [19], then studied in [5] under the name of AFRA (Argumentation Framework with Recursive Attacks). This version describes abstract argumentation systems in which the interactions can be either attacks between arguments or attacks from an argument to another attack. In this case, as for the bipolar case, a translation of an AFRA into an equivalent AS can be defined by the addition of some new arguments and the attacks they produce or they receive. Recently, an extension of these AFRA has been proposed in [13] in order to take into account supports on arguments or on interactions. These systems are called ASAF (Attack-Support Argumentation Frameworks). And, once again, a translation of an ASAF into an equivalent AS is proposed by the addition of arguments and attacks.

The subject of the current paper is to propose a logical vision of an ASAF that can justify the introduction of all these new attacks. This logical vision is issued from works in bioinformatics (see [15; 1]). In this domain, we can find *metabolic networks* that describe the chemical reactions of cells; these reactions can be negative (inhibition of a protein) or positive (production of a new protein) and they can depend on other proteins or other reactions. A translation from metabolic networks to classical logic has been proposed in [1]. This translation allows to use automated

deduction methods for reasoning on these networks. We will show that ASAF are very similar to these graphs considering that inhibition is a kind of attack, and production a kind of (deductive) support.

Some background about argumentation is given in Section 2 for ASAF. Section 4 describes the logical vision of processes of inhibition and production of proteins. Section 5 presents the logical vision of the BAS and the ASAF. Then in Section 6, we give some preliminary results towards a comparative study with existing works. Finally, Section 7 concludes and suggests perspectives of our work.

2 Background of abstract argumentation systems

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 $a \longrightarrow b$.

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 .*

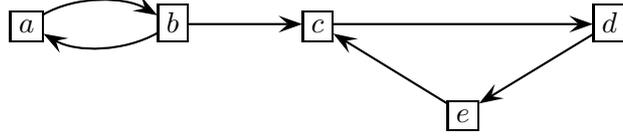
Standard semantics introduced by Dung (preferred, stable, grounded) enable to characterize admissible sets of arguments that satisfy some form of optimality.

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.*

¹if and only if
²such that
³with respect to



The status of an argument is determined by its membership of 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.

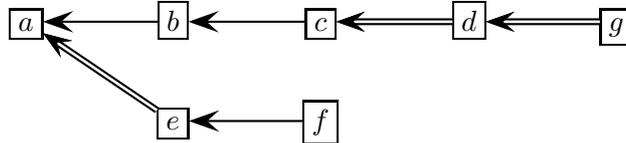
2.2 Abstract bipolar argumentation system (BAS)

The abstract bipolar argumentation framework presented in [8; 9] 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 [3] 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 $a \longrightarrow b$ (resp. by $a \Longrightarrow b$).

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, [6] 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

⁴A third one, the evidential support, has also been proposed in [23; 24; 25] but will not be discussed here.

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 ([6] Mediated attack)

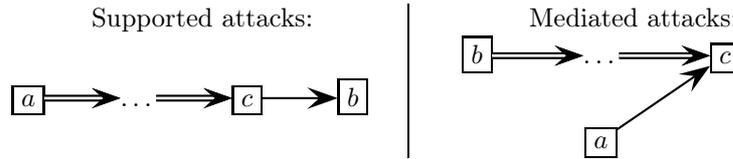
Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. There is a mediated attack from a to b iff there is a sequence $a_1\mathbf{R}_{\text{sup}}\dots\mathbf{R}_{\text{sup}}a_{n-1}$, and $a_n\mathbf{R}_{\text{att}}a_{n-1}$, $n \geq 3$, with $a_1 = b$, $a_n =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

Moreover, the deductive interpretation of the support justifies the introduction of another attack (called supported attacks in [9]): 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 ([9] 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 ([10], 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.

⁵Our notation is different that the one used in [10]. We have modified it in order to homogenize with the notation of necessary attacks defined in [11].

⁶See in [10], some properties of AS^{Dc} .

2.2.2 Necessary supports

Necessary support has been initially⁷ proposed in [21; 22] 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 .

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 [9] and extended attack in [21].

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 [22].

The formal definition of these two complex attacks is:

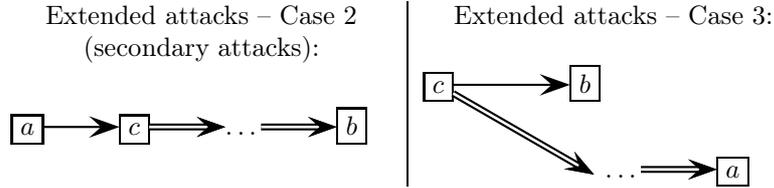
Def. 9 ([22] Extended attack) Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. There is an extended attack from a to b iff

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:



As for deductive supports, from Definition 9, new attacks called **n+-attacks** can be generated inductively as follows:

Def. 10 (n+-attacks) 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. There exists a **n+-attack** from a to b iff

- either $a\mathbf{R}_{\text{att}}b$, or there is a (case 1 or case 2) extended attack from a to b ,
- or there exists an argument c s.t. a **n+-attacks** c and c **supports** b ,
- or there exists an argument c s.t. c **supports** a and c **n+-attacks** b .

Considering the **n+-attacks** on \mathbf{A} enables to define an AS called the complete associated Dung AS for the necessary support of BAS and denoted by AS^{Nc} .

2.2.3 Duality between deductive and necessary supports

In this section, we will use the following notation:

⁷An extension of this work is presented in [20]. 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.

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 [10], 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).
- $\text{AF}_{\text{sym}}^{Dc}$ 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 $\text{AF}_{\text{sym}}^{Dc}$.

Using the above notations, Table 1, issued from [10], gives a synthetic view of the correspondences between the three approaches (abstract, deductive and necessary).

Deductive supports of [6] \mathbf{R}_{sup} is a d-support	Necessary supports of [21; 22] $\mathbf{R}_{\text{sup}}^{-1}$ 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 [11], 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. 11 (Constraint TRA [11]) $\forall a, b \in \mathbf{A}$, 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. 12 (Constraint CLO [11]) *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 $c\mathbf{R}_{\text{sup}}a$, “the acceptance of a implies the acceptance of c ” and so “the acceptance of a implies the non-acceptance of b ”.

Def. 13 (Constraint CFS [11]) $\forall a, b, c \in \mathbf{A}$. *If ($a\mathbf{R}_{\text{att}}c$ and c supports b) or ($c\mathbf{R}_{\text{att}}b$ and c 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. 14 (Constraint nATT [11]) *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. 15 (Constraint n+ATT [11]) *If ($a\mathbf{R}_{\text{att}}c$ and $c\mathbf{R}_{\text{sup}}b$) or ($c\mathbf{R}_{\text{att}}b$ 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.

3 Recursive interactions

The idea of recursive interaction has been introduced in [19] and developed in [5] for recursive attacks and in [12] 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 [19]).

3.1 AS with recursive interactions

In [5], 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 .

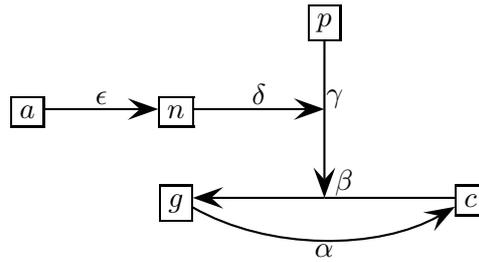
Then in [5], a translation of an AFRA into an AS is provided:

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:

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

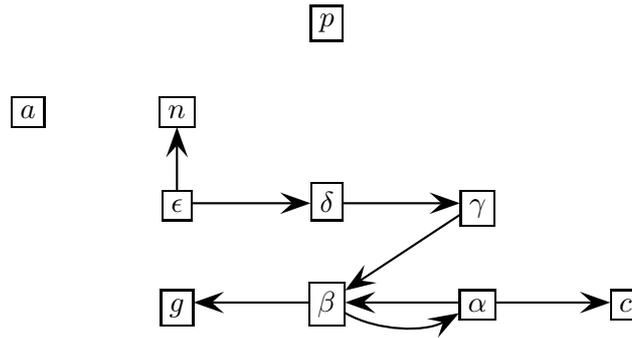
The previous notions are illustrated on the following example:

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 [12], 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 [12]. 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 ASAF is given below:

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

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

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

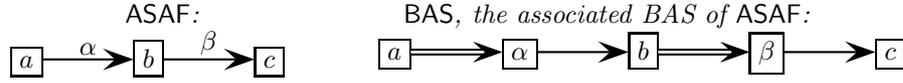
$$\begin{aligned}
\mathbf{R}_{\text{sup}}' = & \quad \{(a, \alpha) | \alpha = (a, x) \in \mathbf{R}_{\text{att}} \text{ and } x \in \mathbf{A} \cup \mathbf{R}_{\text{att}}\} \\
& \cup \{(b, \beta^+) | \beta = (b, y) \in \mathbf{R}_{\text{sup}} \text{ and } y \in \mathbf{A} \cup \mathbf{R}_{\text{att}}\}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

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 problems pointed in AFRA).

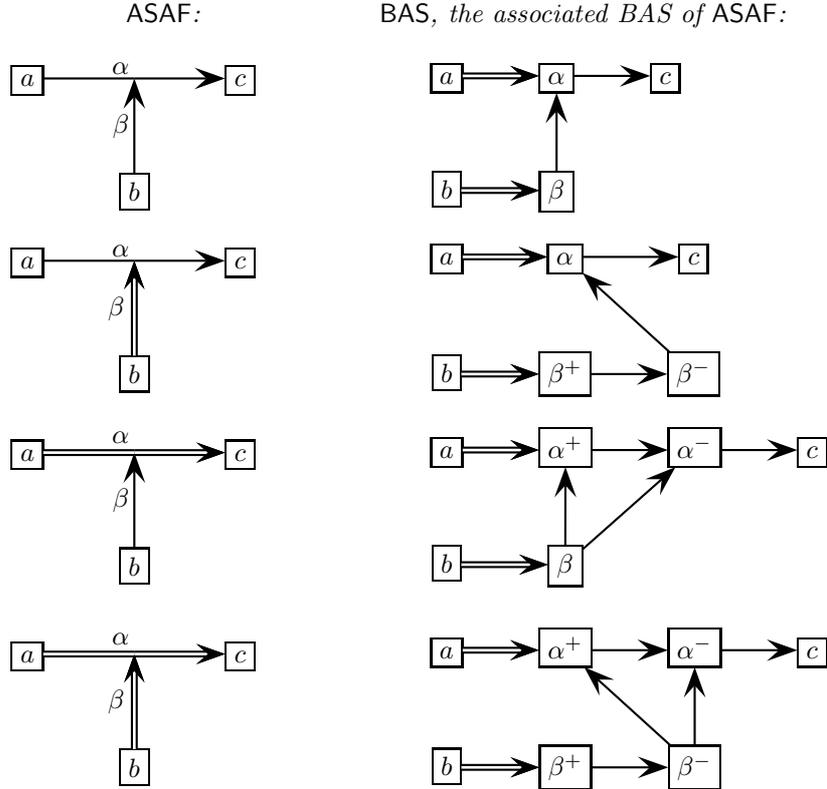
Note also that the attacks and supports of \mathbf{R}_{att}' and \mathbf{R}_{sup}' are “simple” interactions, *i.e.* pairs of elements of \mathbf{A}' which are not labelled with a symbol.

The following examples illustrate different cases:

Ex. 4



Ex. 5 Consider different ASAF 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 ASAF into a BAS with necessary supports, the second step is to create an AS. The approach followed in [12] 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 $BAS = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a necessary BAS. Its associated AS is the pair $AS' = \langle \mathbf{A}', \mathbf{R}' \rangle$ such that

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

Note that the AS obtained using Def. 21 is included in the AS obtained with Def. 9, since Def. 9 (corresponding to Constraint $\mathbf{n+ATT}$) is more general than Def. 21 (corresponding only to Constraint \mathbf{nATT}). Nevertheless, in terms of acceptability, the results are the same (see [11]).

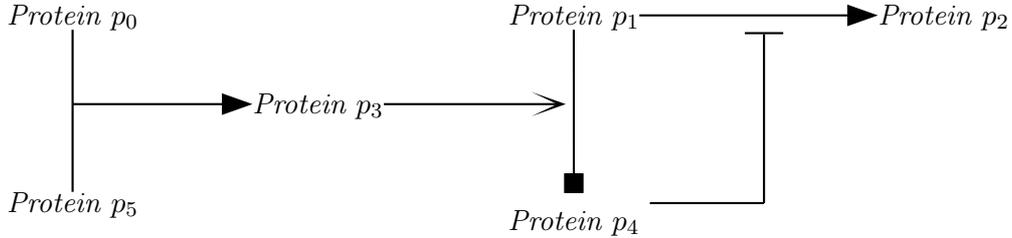
Def. 22 (AS associated with ASAF) Let $ASAF = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$. The AS associated with ASAF is the associated AS of the associated BAS of ASAF.

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

4 Logics in biology

[14; 1] present a logical approach for reasoning on *metabolic networks*. These networks describe the chemical reactions of cells; these reactions can be negative (inhibition of a protein) or positive (production of a new protein) and they can depend on other proteins and other reactions. Such a network can be graphically represented by a molecular interaction map (MIM) that is a graph whose nodes are proteins and edges are either relations between proteins (Protein p_1 can inhibit -resp. product - Protein p_2), or relations from a protein to another relation (Protein p_3 can inhibit -resp. activate - a relation, for instance the relation between p_1 and p_4 in Ex. 6). A translation from metabolic networks to classical logic has been proposed in [1]. This translation allows to use automated deduction methods for reasoning on these networks.

Ex. 6 The following graph gives a short example of a MIM:



Relations in a MIM are not restricted to binary ones: for instance a set of proteins can produce a reaction on another protein. It is the case in the graph above: the set $\{p_0, p_5\}$ produces p_3 .

Four types of edges can be found in a MIM (uppercase letters denote sets of proteins and lowercase letters denote either proteins, or reactions):

- $P \longrightarrow p$: the presence of P induces the production of p ,
- $P \longrightarrow \blacksquare p$: the presence of P inhibits the production of p (P and p cannot be present together),
- $P \longrightarrow \rightrightarrows r$: the presence of P activates the reaction r ,
- $P \longrightarrow \lrcorner r$: the presence of P inhibits the reaction r (if P is present, then the reaction r cannot be activated).

Each type of edge encodes a kind of reaction.

\longrightarrow and $\longrightarrow\blacksquare$ describe a causal link from a protein (or a set of proteins) to another protein. They are called causal reactions in the following.

By contrast, \longrightarrow and $\longrightarrow\perp$ describe the impact of a protein (or a set of proteins) on a reaction r . They are called context reactions in the following. $P\longrightarrow$ and $Q\longrightarrow\perp$ define the context \mathcal{C} in which the reaction r can occur. Moreover a context reaction \longrightarrow (resp. $\longrightarrow\perp$) may also have its own context \mathcal{C}' (resp. \mathcal{C}''). In that case, the context \mathcal{C} is denoted by a pair $\langle \mathcal{C}'P\longrightarrow, \mathcal{C}''Q\longrightarrow\perp \rangle$. Note that a context may be associated to any reaction (including the causal reactions).

Def. 23 (MIM) A MIM is a graph $\langle \mathbf{P}, \mathbf{React} \rangle$ where:

- \mathbf{P} is a finite non-empty set of proteins,
- \mathbf{React} is a set of reactions involving proteins. \mathbf{React} is partitioned into four subsets:
 - \mathbf{ProP} is a subset of $2^{\mathbf{P}} \times \mathbf{P}$ representing the reactions that produce a protein using a set of proteins,
 - \mathbf{InhP} is a subset of $2^{\mathbf{P}} \times \mathbf{P}$ representing the reactions that inhibit the production of a protein using a set of proteins,
 - \mathbf{ActR} is a subset of $2^{\mathbf{P}} \times \mathbf{React}$ representing the reactions in which a set of proteins activates a reaction,
 - \mathbf{InhR} is a subset of $2^{\mathbf{P}} \times \mathbf{React}$ representing the reactions which a set of proteins inhibits a reaction.

$\mathbf{ProP} \cup \mathbf{InhP}$ is the set of causal reactions.

$\mathbf{ActR} \cup \mathbf{InhR}$ is the set of context reactions (reactions describing a part of a context).

In the following we describe the logical translation of each element of a MIM.

Let us first consider a causal reaction without any context. The causal reaction $P \longrightarrow q$ (resp. $P \longrightarrow\blacksquare q$) is interpreted by the fact that if all the proteins of P are present then the protein q is produced (resp. inhibited). Now assume that the causal reaction $P \longrightarrow q$ has the context \mathcal{C} . In that case, the protein q is produced provided that all the proteins of P are present and the context \mathcal{C} is “active”. The fact that q is produced (resp. inhibited) is encoded with the logical symbol $\mathbf{Pr}q$ (resp. $\mathbf{In}q$) and the fact that the context \mathcal{C} is active is encoded with a logical formula $A(\mathcal{C})$. Moreover, a formula will be added for expressing that if p is produced (resp. inhibited) then p is present (resp. not present).

This leads to the following definition (note that the definition of $A(\mathcal{C})$ is given in Def. 25):

Def. 24 (Encoding of causal reactions) Let q be a protein, let $P = \{p_1, \dots, p_n\}$ be a set of proteins.

Let $P \longrightarrow q$ be a causal reaction having the context \mathcal{C} . This reaction can be translated into the two following propositional formulae:

- $(A(\mathcal{C}) \bigwedge_{p_i \in P} p_i) \rightarrow \mathbf{Pr}q$ and
- $\mathbf{Pr}q \rightarrow q$.

Let $P \longrightarrow\blacksquare q$ be a causal reaction having the context \mathcal{C} . This reaction can be translated into the two following propositional formulae:

- $(A(\mathcal{C}) \bigwedge_{p_i \in P} p_i) \rightarrow \mathbf{In}q$ and
- $\mathbf{In}q \rightarrow \neg q$.

The knowledge base Σ_1 giving the translation of all causal reactions of the MIM is defined

by:

$$\begin{aligned}\Sigma_1 = & \{(A(\mathcal{C}) \bigwedge_{p_i \in P} p_i) \rightarrow \text{Pr}q | (P \longrightarrow q) \in \mathbf{ProP}\} \\ & \cup \{(A(\mathcal{C}) \bigwedge_{p_i \in P} p_i) \rightarrow \text{In}q | (P \text{---} q) \in \mathbf{InhP}\} \\ & \cup \{\text{Pr}q \rightarrow q | q \in \mathbf{P}\} \\ & \cup \{\text{In}q \rightarrow \neg q | q \in \mathbf{P}\}\end{aligned}$$

Now let us consider a reaction r with the context $\mathcal{C} = \langle P \longrightarrow, Q \text{---} \rangle$. Let us denote $P \longrightarrow r$ by r_1 and $Q \text{---} r$ by r_2 . The fact that \mathcal{C} is active intuitively means that the context reaction r_1 is “active” and the context reaction r_2 is “inactive”. Moreover r_1 is considered as active if each protein of the set P is present, and r_2 is considered as inactive if at least one protein of the set Q is absent. That enables to define the formula $A(\mathcal{C})$ which expresses that the context is “active”. More generally, the context reactions r_1 and r_2 themselves may have a context. In that case, r_1 is considered as active if each protein of the set P is present and the context of r_1 is active. And similarly r_2 is considered as inactive if at least one protein of the set Q is absent or the context of r_2 is not active.

This leads to the following definition:

Def. 25 (Encoding context reactions) Let $\langle \mathbf{P}, \mathbf{React} \rangle$ be a MIM. Let $r \in \mathbf{React}$ be a reaction of the MIM. Let $P = \{p_1, \dots, p_n\}$, $Q = \{q_1, \dots, q_m\}$ be sets of proteins. Let $\mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \text{---} \rangle$ be the context of r , two logical expressions can be defined, the activation expression $A(\mathcal{C})$ and the inhibition expression $I(\mathcal{C})$:

$$A(\mathcal{C}) = \left(\bigwedge_{p_i \in P} p_i \right) \wedge A(\mathcal{C}') \wedge \left(\bigvee_{q_j \in Q} \neg q_j \right) \vee I(\mathcal{C}'') \quad I(\mathcal{C}) = \left(\bigvee_{p_i \in P} \neg p_i \right) \vee I(\mathcal{C}') \vee \left(\bigwedge_{q_j \in Q} q_j \right) \wedge A(\mathcal{C}'')$$

The knowledge base Σ_2 giving the translation of all context reactions of the MIM is defined by:

$$\begin{aligned}\Sigma_2 = & \{A(\mathcal{C}) \leftrightarrow ((\bigwedge_{p_i \in P} p_i) \wedge A(\mathcal{C}') \wedge ((\bigvee_{q_j \in Q} \neg q_j) \vee I(\mathcal{C}'')))) | \\ & \exists r \in \mathbf{React} \text{ such that } \mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \text{---} \rangle \text{ is the context of } r\} \\ \cup & \{I(\mathcal{C}) \leftrightarrow ((\bigvee_{p_i \in P} \neg p_i) \vee I(\mathcal{C}') \vee ((\bigwedge_{q_j \in Q} q_j) \wedge A(\mathcal{C}'')))) | \\ & \exists r \in \mathbf{React} \text{ such that } \mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \text{---} \rangle \text{ is the context of } r\}\end{aligned}$$

Note that, in the above definition, the activation expression and the inhibition expression are defined inductively. In the particular case of a simple context of the form $\mathcal{C} = \langle P \longrightarrow, Q \text{---} \rangle$, it can be easily proved that $I(\mathcal{C})$ is equivalent to $\neg A(\mathcal{C})$. Then by induction on the structure of a context it can be proved that it is also the case for a context of the form $\mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \text{---} \rangle$.

Prop. 1 Let $\mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \text{---} \rangle$ be the context of a reaction r of a given MIM. It holds that $I(\mathcal{C})$ is equivalent to $\neg A(\mathcal{C})$.

Proof:

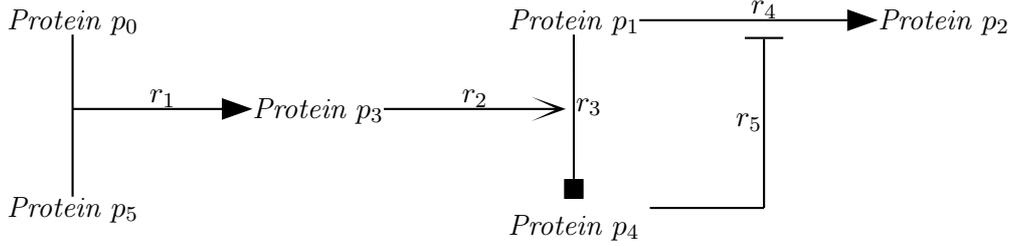
- Basic case : Let $\mathcal{C} = \langle P \longrightarrow, Q \text{---} \rangle$. $A(\mathcal{C}) = (\bigwedge_{p_i \in P} p_i) \wedge (\bigvee_{q_j \in Q} \neg q_j)$ and $I(\mathcal{C}) = (\bigvee_{p_i \in P} \neg p_i) \vee (\bigwedge_{q_j \in Q} q_j)$. Obviously, $I(\mathcal{C})$ is equivalent to $\neg A(\mathcal{C})$.

- Induction step : Let $\mathcal{C} = \langle \mathcal{C}' P \longrightarrow, \mathcal{C}'' Q \longleftarrow \rangle$ and assume that $I(\mathcal{C}') = \neg A(\mathcal{C}')$ and $I(\mathcal{C}'') = \neg A(\mathcal{C}'')$. We have $I(\mathcal{C}) = (\bigvee_{p_i \in P} \neg p_i) \vee I(\mathcal{C}') \vee ((\bigwedge_{q_j \in Q} q_j) \wedge A(\mathcal{C}''))$. So $I(\mathcal{C}) = (\bigvee_{p_i \in P} \neg p_i) \vee \neg A(\mathcal{C}') \vee ((\bigwedge_{q_j \in Q} q_j) \wedge \neg I(\mathcal{C}''))$. As $A(\mathcal{C}) = (\bigwedge_{p_i \in P} p_i) \wedge A(\mathcal{C}') \wedge ((\bigvee_{q_j \in Q} \neg q_j) \vee I(\mathcal{C}''))$, we obtain that $I(\mathcal{C})$ is equivalent to $\neg A(\mathcal{C})$.

□

Def. 26 (Encoding of MIM) Let $\langle \mathbf{P}, \mathbf{React} \rangle$ be a MIM. The knowledge base $\Sigma = \Sigma_1 \cup \Sigma_2$ is the translation of this MIM.

Ex. 6 (cont'd) In this example, we first introduce labels on edges in order to name reactions and to distinguish between causal and context reactions:



Reactions r_1 , r_3 and r_4 are causal reactions. r_2 and r_5 are context reactions.

r_1 , r_2 and r_5 have no context while r_3 and r_4 have a context.

- let $\mathcal{C} = \langle p_3 \longrightarrow, - \rangle$ be the context of r_3 (p_3 is used for activating r_3 and there is no specified inhibition of r_3).
- let $\mathcal{C}' = \langle -, p_4 \longleftarrow \rangle$ the context of r_4 is (p_4 is used for inhibiting r_4 and there is no specified activation of r_4).

Applying definitions 25 and 24 we obtain:

- $(p_5 \wedge p_0) \rightarrow \text{Pr}p_3$ (p_0 and p_5 produce p_3)
- $(A(\mathcal{C}) \wedge p_1) \rightarrow \text{In}p_4$ (p_1 produces p_4 with the context of the reaction r_3); and $A(\mathcal{C}) \leftrightarrow p_3$, we have $(p_3 \wedge p_1) \rightarrow \text{In}p_4$
- $(A(\mathcal{C}') \wedge p_1) \rightarrow \text{Pr}p_2$ (p_1 inhibits p_2 with the context of the reaction r_4); and $A(\mathcal{C}') \leftrightarrow \neg p_4$, we have $(\neg p_4 \wedge p_1) \rightarrow \text{Pr}p_2$
- and then all the formulae $\text{Pr}p_i \rightarrow p_i$ and $\text{In}p_i \rightarrow \neg p_i$

So the corresponding knowledge base is equivalent to:

$$\begin{aligned}
(p_5 \wedge p_0) &\rightarrow \text{Pr}p_3 \\
(p_3 \wedge p_1) &\rightarrow \text{In}p_4 \\
(\neg p_4 \wedge p_1) &\rightarrow \text{Pr}p_2 \\
\text{Pr}p_0 &\rightarrow p_0 \\
\text{In}p_0 &\rightarrow \neg p_0 \\
\text{Pr}p_1 &\rightarrow p_1 \\
\text{In}p_1 &\rightarrow \neg p_1 \\
\text{Pr}p_2 &\rightarrow p_2 \\
\text{In}p_2 &\rightarrow \neg p_2 \\
\text{Pr}p_3 &\rightarrow p_3 \\
\text{In}p_3 &\rightarrow \neg p_3 \\
\text{Pr}p_4 &\rightarrow p_4 \\
\text{In}p_4 &\rightarrow \neg p_4 \\
\text{Pr}p_5 &\rightarrow p_5 \\
\text{In}p_5 &\rightarrow \neg p_5
\end{aligned}$$

Then using a logical solver, we are able to deduce the presence of Protein p_2 from the presence of the proteins p_0 , p_1 and p_5 . By contrast, Protein p_1 is not enough for producing p_2 .

The logical translation of a MIM always gives a consistent knowledge base:

Prop. 2 *The knowledge base associated with a MIM is logically consistent.*

Proof: Let a consider a MIM built on the set $\{p_1, \dots, p_n\}$ of proteins. Let Σ be its associated knowledge base. In the MIM, we can find only the following elements:

- $P \longrightarrow q$ with the context \mathcal{C} , which is translated into the following formulae of Σ :
 $(A(\mathcal{C}) \wedge_{p_i \in P} p_i) \rightarrow \text{Pr}q$ and $\text{Pr}q \rightarrow q$;
- $P \longrightarrow \blacksquare q$ with the context \mathcal{C} , which is translated into the following formulae of Σ :
 $(A(\mathcal{C}) \wedge_{p_i \in P} p_i) \rightarrow \text{In}q$ and $\text{In}q \rightarrow \neg q$;
- $P \longrightarrow r$ and $Q \longrightarrow \lrcorner r$, which define the context \mathcal{C} of r and are only used for expressing $A(\mathcal{C})$ and its negation:
 $A(\mathcal{C}) \leftrightarrow (\bigwedge_{p_i \in P} p_i) \wedge A(\mathcal{C}') \wedge ((\bigvee_{q_j \in Q} \neg q_j) \vee I(\mathcal{C}''))$.

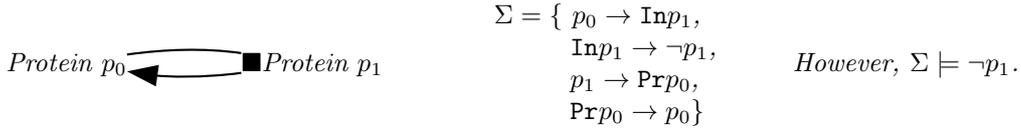
The following assignation gives a model of Σ :

- each p_i is assigned to *false*, so each formula $(A(\mathcal{C}) \wedge_{p_i \in P} p_i) \rightarrow \text{Pr}q$, each formula $(A(\mathcal{C}) \wedge_{p_i \in P} p_i) \rightarrow \text{In}q$ are assigned to *true* (whatever the value assigned to $A(\mathcal{C})$, for any \mathcal{C});
- each $\text{Pr}p_i$ is assigned to *false*, so each formula $\text{Pr}q \rightarrow q$ is assigned to it true;
- each $\text{In}p_i$ is assigned to *false*, so each formula $\text{In}q \rightarrow \neg q$ is assigned to it true.

Since Σ has a model, Σ is consistent. □

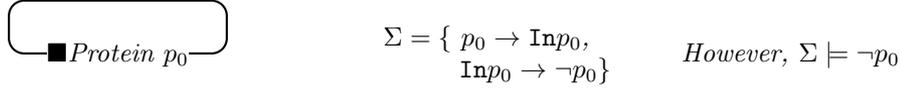
However, as soon as we introduce the presence of a protein in the MIM, the corresponding knowledge base may become inconsistent:

Ex. 7 *The following MIM is encoded by a consistent base Σ :*



So $\Sigma \cup \{p_1\}$ is inconsistent.

Ex. 8 *The following MIM is encoded by a consistent base Σ :*



So $\Sigma \cup \{p_0\}$ is inconsistent.

5 A logical representation of an argumentation graph

5.1 The case of a bipolar argumentation graph

It is easy to draw a parallel between a MIM and a bipolar argumentation graph. More precisely:

- the notion of *attack* between two arguments in a BAS corresponds to the notion of inhibition of a protein by another protein in a MIM:

the reaction $p_1 \longrightarrow \blacksquare p_2$ means “if p_1 is present then p_2 is not present”;

the attack $a_1 \longrightarrow a_2$ means “if a_1 is accepted then a_2 is not accepted”.

- the notion of *deductive support* between two arguments in a BAS corresponds to the notion of production of a protein by another protein in a MIM:

the reaction $p_1 \longrightarrow p_2$ means “if p_1 is present then p_2 is present”;

the deductive support $a_1 \Longrightarrow a_2$ means “if a_1 is accepted then a_2 is accepted”.

Note that in a BAS the interactions are always binary and no interaction can impact on another interaction.

Using the duality between deductive and necessary supports, we pursue the parallel and establish a correspondance between the notion of *necessary support* between two arguments in a BAS and the notion of production of a protein by another protein in a MIM:

the reaction $p_1 \longrightarrow p_2$ means “if p_1 is present then p_2 is present”;

the necessary support $a_2 \Longrightarrow a_1$ corresponds to “if a_1 is accepted then a_2 is accepted”.

The correspondances drawn above lead us to propose a logical encoding of a BAS. For that purpose, we consider the following propositional variables associated with a set \mathbf{A} of arguments: for each $a \in \mathbf{A}$,

- the variable a means that a is accepted,
- the variable Pra means that a is a supporter (the source of a necessary support),
- the variable Ina means that a is attacked.

In the following \mathcal{L} denotes the propositional language built on these propositional variables.

Def. 27 (Logical translation of a necessary BAS) *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system, where \mathbf{R}_{sup} is a necessary support. Let $a, b \in \mathbf{A}$.*

- *An attack $a \longrightarrow b$ is translated into the two following propositional formulae of \mathcal{L} :*
 - $a \rightarrow \text{In}b$ and
 - $\text{In}b \rightarrow \neg b$.
- *A necessary support $a \Longrightarrow b$ is translated into the two following propositional formulae of \mathcal{L} :*
 - $b \rightarrow \text{Pra}$ and
 - $\text{Pra} \rightarrow a$.

The knowledge base associated with BAS is denoted by $\Sigma(\text{BAS})$ (or Σ for short) and is defined by:

$$\begin{aligned} \Sigma = & \{b \rightarrow \text{Pra} \mid (a, b) \in \mathbf{R}_{\text{sup}}\} \\ & \bigcup \{a \rightarrow \text{In}b \mid (a, b) \in \mathbf{R}_{\text{att}}\} \\ & \bigcup \{\text{Pra} \rightarrow a \mid a \in \mathbf{A} \text{ and } \exists x \in \mathbf{A} \text{ s.t. } (a, x) \in \mathbf{R}_{\text{sup}}\} \\ & \bigcup \{\text{In}a \rightarrow \neg a \mid a \in \mathbf{A} \text{ and } \exists x \in \mathbf{A} \text{ s.t. } (x, a) \in \mathbf{R}_{\text{att}}\} \end{aligned}$$

Ex. 9 *Consider the following BAS (with necessary supports): $a \longrightarrow b \Longrightarrow c \longrightarrow d$.*

Its associated knowledge base is:

$$\begin{aligned} \Sigma = & \{a \rightarrow \text{In}b, \\ & \text{In}b, b \rightarrow, \\ & c \rightarrow \text{Pr}b, \\ & \text{Pr}b \rightarrow b, \\ & c \rightarrow \text{In}d, \\ & \text{In}d, d \rightarrow\} \end{aligned}$$

Some interesting results can be proved. The first result concerns the consistency of the associated knowledge base:

Prop. 3 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a necessary bipolar argumentation system. Let Σ be its associated knowledge base. Σ is always consistent.*

Proof: It is a direct consequence of Prop. 2. \square

The second result gives a condition for introducing an inconsistency:

Prop. 4 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system. Let Σ be its associated knowledge base. If there exist $a, b \in \mathbf{A}$ such that $a \mathbf{R}_{\text{att}} b$, then $\Sigma \cup \{a, b\}$ is inconsistent.*

Proof: Following Def. 27, Σ contains the formulae $a \rightarrow \text{In}b$ and $\text{In}b \rightarrow \neg b$. So $\Sigma \cup \{a, b\}$ is inconsistent. \square

In particular, this property can be applied to the case of a self-attacked argument, or more generally to the case of cycles of attacks whose length is upper than 1. Note also that the above inconsistency is produced by two arguments related by a direct attack. No inconsistency appears in the case of an indirect attack, as shown by the following example:

Ex. 10 *Consider the following sequence of attacks: $a_1 \mathbf{R}_{\text{att}} a_2 \mathbf{R}_{\text{att}} a_3 \mathbf{R}_{\text{att}} a_4$. Following Def. 27, the corresponding Σ contains $a_1 \rightarrow \text{In}a_2, \dots, a_3 \rightarrow \text{In}a_4$ and $\forall i = 1 \dots 4, \text{In}a_i \rightarrow \neg a_i$. We can easily see that $\Sigma \cup \{a_1\}$ infers $\neg a_2$, but it does not infer $\neg a_4$ whereas a_1 is an indirect attacker of a_4 .*

Another situation in which inconsistency may occur comes from the combination of support and attack.

Prop. 5 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. If there exist two arguments $a, b \in \mathbf{A}$ such that there is an extended attack from a to b then $\Sigma \cup \{a, b\}$ is inconsistent.*

Proof: Due to Def. 9, three cases must be considered:

1. $a \mathbf{R}_{\text{att}} b$: it follows from Prop. 4 that $\Sigma \cup \{a, b\}$ is inconsistent.
2. There is a sequence $a = a_1 \mathbf{R}_{\text{att}} a_2 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n = b, n \geq 3$: Following Def. 27, Σ contains the formulae $a_1 \rightarrow \text{In}a_2, \text{In}a_2 \rightarrow \neg a_2$, and $b = a_n \rightarrow \text{Pr}a_{n-1}, \text{Pr}a_{n-1} \rightarrow a_{n-1} \dots a_3 \rightarrow \text{Pr}a_2, \text{Pr}a_2 \rightarrow a_2$. It is easy to see that Σ infers $a \rightarrow \neg a_2$ and $b \rightarrow a_2$. So $\Sigma \cup \{a, b\}$ is inconsistent.
3. There is a sequence $a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n = a$, and $a_1 \mathbf{R}_{\text{att}} b, n \geq 2$: Following Def. 27, Σ contains the formulae $a_1 \rightarrow \text{In}b, \text{In}b \rightarrow \neg b$, and $a = a_n \rightarrow \text{Pr}a_{n-1}, \text{Pr}a_{n-1} \rightarrow a_{n-1} \dots a_2 \rightarrow \text{Pr}a_1, \text{Pr}a_1 \rightarrow a_1$. It is easy to see that Σ infers $a_1 \rightarrow \neg b$ and $a \rightarrow a_1$. So $\Sigma \cup \{a, b\}$ is inconsistent.

\square

A particular case of the above proposition occurs when $a \mathbf{R}_{\text{att}} b$ and $a \mathbf{R}_{\text{sup}} b$. In that case, there is an extended attack from b to b . So $\Sigma \cup \{b\}$ is inconsistent.

However the presence of an argument that attacks and supports another argument is not a source of inconsistency.

Ex. 11 *Assume that $a \mathbf{R}_{\text{att}} b$ and $a \mathbf{R}_{\text{sup}} b$. Following Def. 27, Σ contains the formulae $a \rightarrow \text{In}b, b \rightarrow \text{Pr}a, \text{In}b \rightarrow \neg b$ and $\text{Pr}a \rightarrow a$. Obviously $\Sigma \cup \{a\}$ is consistent.*

More generally, given a necessary BAS, conflict-freeness in the complete associated Dung AS (denoted by AS^{Nc} , see Def. 10) can be verified on the associated knowledge base Σ owing to the following result:

Prop. 6 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $S \subseteq \mathbf{A}$. S is conflict-free in AS^{Nc} iff $\Sigma \cup S$ is consistent.*

Proof: We proof the contrapositive: S is not conflict-free in AS^{Nc} iff $\Sigma \cup S$ is inconsistent.

\Rightarrow Let us assume that S is not conflict-free in AS^{Nc} . There exist $a, b \in S$ such that there is an attack from a to b in AS^{Nc} . Due to Def. 10, there exist a sequence $a_n \mathbf{R}_{\text{att}} b_p \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} b_1 = b$ and a sequence $a_n \mathbf{R}_{\text{sup}} a_{n-1} \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_1 = a$ with $n \geq 1, p \geq 1$. Following Def. 27, Σ contains the formulas $a_n \rightarrow \text{In}b_p$, $\text{In}b_p \rightarrow \neg b_p$, and $b = b_1 \rightarrow \text{Pr}b_2$, $\text{Pr}b_2 \rightarrow b_2 \dots b_{p-1} \rightarrow \text{Pr}b_p$, $\text{Pr}b_p \rightarrow b_p$ and $a = a_1 \rightarrow \text{Pr}a_2$, $\text{Pr}a_2 \rightarrow a_2 \dots a_{n-1} \rightarrow \text{Pr}a_n$, $\text{Pr}a_n \rightarrow a_n$. It is easy to see that Σ infers $a \rightarrow a_n$, $a_n \rightarrow \neg b_p$ and $b \rightarrow b_p$. So $\Sigma \cup \{a, b\}$ is inconsistent.

\Leftarrow Let us assume that $\Sigma \cup S$ is inconsistent. Note that Σ contains only binary Horn clauses. So if $\Sigma \cup S$ is inconsistent, $\Sigma \cup S$ infers two complementary literals. Moreover, as each symbol $\text{In}a$ (resp. $\text{Pr}a$) is associated with only one argument, $\Sigma \cup S$ infers two complementary literals of the form c and $\neg c$. So we have:

- $\Sigma \cup S$ infers c means that either $c \in S$ or there exist in Σ the formulae $x \rightarrow \text{Pr}c$ and $\text{Pr}c \rightarrow c$ with $\Sigma \cup S$ infers x . By induction, it is easy to prove that ($\Sigma \cup S$ infers c) is equivalent to ($c \in S$ or there is in Σ a sequence $c_k \rightarrow \text{Pr}c_{k-1}$, $\text{Pr}c_{k-1} \rightarrow c_{k-1} \dots c_2 \rightarrow \text{Pr}c_1$, $\text{Pr}c_1 \rightarrow c_1 = c$ with $c_k \in S$).
- $\Sigma \cup S$ infers $\neg c$ means that either (Case 1) there exist in Σ the formulae $x \rightarrow \text{In}c$ and $\text{In}c \rightarrow \neg c$ with $\Sigma \cup S$ infers x , or (Case 2) there exist in Σ the formulae $c \rightarrow \text{In}y$ and $\text{In}y \rightarrow \neg y$ with $\Sigma \cup S$ infers y . In Case 2, as $\Sigma \cup S$ infers c it follows that $\Sigma \cup S$ infers y and $\neg y$, with the inference of $\neg y$ falling within Case 1.

So we can assume that $\Sigma \cup S$ infers two complementary literals of the form c and $\neg c$ with a Case 1 inference for $\neg c$. Consequently, the above equivalences can be rewritten in terms of supports and attacks, producing the following results :

- $\Sigma \cup S$ infers c means that there is a sequence of supports $c = c_1 \mathbf{R}_{\text{sup}} c_2 \dots \mathbf{R}_{\text{sup}} c_k$ with $c_k \in S$ and $k \geq 1$.
- $\Sigma \cup S$ infers $\neg c$ means that there are the sequences $x \mathbf{R}_{\text{att}} c$ and $x = x_1 \mathbf{R}_{\text{sup}} x_2 \dots \mathbf{R}_{\text{sup}} x_k$ with $x_k \in S$ and $k \geq 1$.

We recover exactly the conditions under which there is an attack from x_k et c_k in AS^{Nc} . As $x_k, c_k \in S$, that proves that S is not conflict-free in AS^{Nc} .

□

The above result enables to determine the conflict-free subsets of the complete associated Dung AS, by checking Σ - consistency. Conflict-freeness is the basic requirement for extensions in standard semantics in argumentation. Now, it is worth considering other issues of argumentation in terms of logical issues. In other words, we want to find correspondences between the characteristic properties of a semantics and logical criteria. For instance, we are interested in the determination of stable sets of arguments, through direct manipulations of the logical knowledge base.

In the following, we propose some preliminary results towards that research direction.

Prop. 7 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $x \in \mathbf{A}$. There exists no argument y in \mathbf{A} s.t. $y \mathbf{R}_{\text{att}} x$ iff there is no formula in Σ containing the variable $\text{In}x$.

The proof of this proposition is obvious following the definition of Σ .

Prop. 8 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $a, b \in \mathbf{A}$. There is an attack from a to b in AS^{Nc} iff there exists $c \in \mathbf{A}$ such that Σ infers the two formulas $a \rightarrow \text{In}c$ and $b \rightarrow \neg \text{In}c$.

Proof:

\Rightarrow Assume that there is an attack from a to b in AS^{Nc} . As said in the first part of the proof of Prop. 6, there exist a sequence $a_n \mathbf{R}_{att} b_p \mathbf{R}_{sup} \dots \mathbf{R}_{sup} b_1 = b$ and a sequence $a_n \mathbf{R}_{sup} a_{n-1} \mathbf{R}_{sup} \dots \mathbf{R}_{sup} a_1 = a$ with $n \geq 1, p \geq 1$. Following Def. 27, Σ contains the formulas $a_n \rightarrow \mathbf{In}b_p$, $\mathbf{In}b_p \rightarrow \neg b_p$, and $b = b_1 \rightarrow \mathbf{Pr}b_2$, $\mathbf{Pr}b_2 \rightarrow b_2 \dots b_{p-1} \rightarrow \mathbf{Pr}b_p$, $\mathbf{Pr}b_p \rightarrow b_p$ and $a = a_1 \rightarrow \mathbf{Pr}a_2$, $\mathbf{Pr}a_2 \rightarrow a_2 \dots a_{n-1} \rightarrow \mathbf{Pr}a_n$, $\mathbf{Pr}a_n \rightarrow a_n$. It is easy to see that Σ infers $a \rightarrow a_n$, $a_n \rightarrow \mathbf{In}b_p$, $b \rightarrow b_p$ and also $\mathbf{In}b_p \rightarrow \neg b_p$. So, Σ infers the two formulas $a \rightarrow \mathbf{In}b_p$ and $b \rightarrow \neg \mathbf{In}b_p$.

\Leftarrow Assume that there exists $c \in \mathbf{A}$ such that Σ infers the two formulas $a \rightarrow \mathbf{Inc}$ and $b \rightarrow \neg \mathbf{Inc}$.

- Σ infers $a \rightarrow \mathbf{Inc}$ means that either the formula $a \rightarrow \mathbf{Inc}$ belongs to Σ , or there exists in Σ a formula $a_n \rightarrow \mathbf{Inc}$ such that Σ infers $a \rightarrow a_n$ (or equivalently $\Sigma \cup \{a\}$ infers a_n). The fact that $a_n \rightarrow \mathbf{Inc}$ belongs to Σ means that there is a direct attack from a_n to c . From the second part of the proof of Prop. 6, $\Sigma \cup \{a\}$ infers a_n means that there is a sequence of supports $a_n \mathbf{R}_{sup} a_{n-1} \dots \mathbf{R}_{sup} a$ with $n \geq 1$.
- Σ infers $b \rightarrow \neg \mathbf{Inc}$ means that either the formula $\mathbf{Inc} \rightarrow \neg b$ belongs to Σ (in that case $b = c$), or Σ contains the formula $\mathbf{Inc} \rightarrow \neg c$ and Σ infers $b \rightarrow c$ (or equivalently $\Sigma \cup \{b\}$ infers c). In the second case, we can find a sequence $c = b_p \mathbf{R}_{sup} b_{p-1} \dots \mathbf{R}_{sup} b_1 = b$ with $p \geq 1$.

Bringing together the above sequences, we obtain $a_n \mathbf{R}_{att} c = b_p \mathbf{R}_{sup} \dots \mathbf{R}_{sup} b_1 = b$ and $a_n \mathbf{R}_{sup} a_{n-1} \mathbf{R}_{sup} \dots \mathbf{R}_{sup} a_1 = a$ with $n \geq 1, p \geq 1$. That exactly corresponds to an attack from a to b in AS^{Nc} (see Def. 10). □

Corol. 1 Let $BAS = \langle \mathbf{A}, \mathbf{R}_{att}, \mathbf{R}_{sup} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $S \subseteq \mathbf{A}$. S is stable in AS^{Nc} iff $\Sigma \cup S$ is consistent, and $\forall b \in \mathbf{A} \setminus S$, there exists $c \in \mathbf{A}$ such that Σ infers $b \rightarrow \neg \mathbf{Inc}$ and $\Sigma \cup S$ infers \mathbf{Inc} .

Proof:

First of all, due to Prop. 6, S is conflict-free in AS^{Nc} iff $\Sigma \cup S$ is consistent.

\Rightarrow Assume that $\forall b \in \mathbf{A} \setminus S$, there exists $a \in S$ such that a attacks b in AS^{Nc} . Using Prop. 8, there exists $c \in \mathbf{A}$ such that Σ infers the two formulas $a \rightarrow \mathbf{Inc}$ and $b \rightarrow \neg \mathbf{Inc}$. So $\Sigma \cup S$ infers \mathbf{Inc} .

\Leftarrow Assume that $\forall b \in \mathbf{A} \setminus S$, there exists $c \in \mathbf{A}$ such that Σ infers $b \rightarrow \neg \mathbf{Inc}$ and $\Sigma \cup S$ infers \mathbf{Inc} . If $\Sigma \cup S$ infers \mathbf{Inc} , there exists $x \in \mathbf{A}$ such that Σ contains the formula $x \rightarrow \mathbf{Inc}$ and $\Sigma \cup S$ infers x . Using one again the proof of Prop. 6, there exists a sequence $x_k \rightarrow \mathbf{Pr}x_{k-1}$, $\mathbf{Pr}x_{k-1} \rightarrow x_{k-1} \dots x_2 \rightarrow \mathbf{Pr}x_1$, $\mathbf{Pr}x_1 \rightarrow x_1 = x$ with $x_k \in S$ and $k \geq 1$. Let $a = x_k$. We have $a \in S$ and Σ infers $a \rightarrow \mathbf{Inc}$. So, since Σ infers $b \rightarrow \neg \mathbf{Inc}$, following Prop. 8, we can conclude that there exists an attack from a to b in AS^{Nc} . □

As a particular case, we obtain a sufficient condition for stability:

Corol. 2 Let $BAS = \langle \mathbf{A}, \mathbf{R}_{att}, \mathbf{R}_{sup} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $S \subseteq \mathbf{A}$ such that $\Sigma \cup S$ is consistent. If $\forall b \in \mathbf{A} \setminus S$ $\Sigma \cup S$ infers $\mathbf{In}b$, then S is stable in AS^{Nc} .

Proof: If $\Sigma \cup S$ infers $\mathbf{In}b$, there exists the formula $\mathbf{In}b \rightarrow \neg b$ in Σ . So Σ infers $b \rightarrow \neg \mathbf{In}b$ and $\Sigma \cup S$ infers $\mathbf{In}b$. Cor. 1 can be applied with $c = b$. □

A particular case of Prop. 8 concerns the characterization of the extended attacks in Case 3 of Def. 9.

Prop. 9 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with necessary support. Let Σ be its associated knowledge base. Let $a, b \in \mathbf{A}$. There is a Case 3-extended attack from a to b in AS^{Nc} iff Σ infers $a \rightarrow \text{Inb}$.

Proof:

- \Rightarrow Assume that there exist a sequence $a_n \mathbf{R}_{\text{att}} b$ and a sequence $a_n \mathbf{R}_{\text{sup}} a_{n-1} \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_1 = a$ with $n \geq 1$. Following Def. 27, Σ contains the formulas $a_n \rightarrow \text{Inb}$, $\text{Inb} \rightarrow \neg b$ and $a = a_1 \rightarrow \text{Pra}_2$, $\text{Pra}_2 \rightarrow a_2 \dots a_{n-1} \rightarrow \text{Pra}_n$, $\text{Pra}_n \rightarrow a_n$. It is easy to see that Σ infers $a \rightarrow a_n$ and $a_n \rightarrow \text{Inb}$.
- \Leftarrow Assume that Σ infers $a \rightarrow \text{Inb}$. From the second part of the proof of Prop. 8, we have that there exists $a_n \in \mathbf{A}$ such that there is a direct attack from a_n to b and a sequence of supports $a_n \mathbf{R}_{\text{sup}} a_{n-1} \dots \mathbf{R}_{\text{sup}} a$ with $n \geq 1$. That is exactly a Case 3-extended attack from a to b .

□

Note that the above result is false for Case 2-extended attacks. If $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$, there is a Case 2-extended attack from a to b . The associated knowledge base Σ contains the formulas $a \rightarrow \text{Inc}$, $\text{Inc} \rightarrow \neg c$, $b \rightarrow \text{Pr}c$ and $\text{Pr}c \rightarrow c$. So Σ infers $a \rightarrow \text{Inc}$ and $b \rightarrow \neg \text{Inc}$ as stated in Prop. 8, but Σ does not infer $a \rightarrow \text{Inb}$ since the propositional variable Inb does not appear in Σ .

5.2 The case of a recursive bipolar argumentation graph (ASAF)

In this section, we extend the language defined in the previous section, in order to take into account recursive interactions. Let us start with the following remarks. In Sect. 3.2, if an attack $\alpha = a_1 \mathbf{R}_{\text{att}} a_2$ is attacked, it may be inactive. In the case of a MIM, if the reaction $r = p_1 \text{---} \blacksquare p_2$ has the context $\mathcal{C} = \langle P \text{---} \rightarrow, Q \text{---} \perp \rangle$, the fact that the context \mathcal{C} is active is expressed by the formula $A(\mathcal{C})$. Indeed, it means that the context reaction $P \text{---} \rightarrow r$ is active and the context reaction $Q \text{---} \perp r$ is inactive. So we can relate the notion of active attack to the notion of active context.

In other words, the following links can be established:

- the notion of *attacked or supported attack* between two arguments in argumentation corresponds to the notion of protein inhibition by another protein together with a given context in a MIM:

the reaction $p_1 \text{---} \blacksquare p_2$ with Context \mathcal{C} corresponds to “if p_1 is present and the context \mathcal{C} is active then p_2 is not present”;

the attack $\alpha = (a_1, a_2)$ corresponds to “if a_1 is accepted and the interaction α is active then a_2 is not accepted”.

- the notion of *attacked or supported deductive support* between two arguments in argumentation corresponds to the notion of protein production by another protein together with a given context in a MIM:

the reaction $p_1 \text{---} \blacktriangleright p_2$ with Context \mathcal{C} corresponds to “if p_1 is present and the context \mathcal{C} is active then p_2 is present”;

the support $\alpha = (a_1, a_2)$ corresponds to “if a_1 is accepted and the interaction α is active then a_2 is accepted”.

Using the duality between deductive and necessary supports, we may establish the following link between the notion of *attacked or supported necessary support* between two arguments in argumentation and the notion of protein production by another protein with a given context in a MIM:

the reaction $p_1 \longrightarrow p_2$ with Context \mathcal{C} corresponds to “if p_1 is present and the context \mathcal{C} is active then p_2 is present”;

the necessary support $\alpha = (a_2, a_1)$ corresponds to “if a_1 is accepted and the interaction α is active then a_2 is accepted”.

Owing to the above correspondences, we pursue the parallel between argumentation graphs and MIM (initiated for BAS in Sect. 5.1) by associating a context with each interaction being attacked or supported.

- Let $\alpha = (a_1, a_2)$ be an interaction which is attacked by $\beta = (x, \alpha)$. It can be considered that α has the context $\mathcal{C} = \langle -, x \longrightarrow \rangle$.
- Let $\alpha = (a_1, a_2)$ be an interaction which is supported by $\beta = (x, \alpha)$. Similarly, it could be considered that α has the context $\mathcal{C} = \langle x \implies, - \rangle$. However, in a MIM, the meaning of the first part of a context is an activation link, which corresponds to a deductive support. In contrast, in ASAF, β is a necessary support from x to α . So strictly speaking, it is not possible to use duality and to turn β into a deductive support from the interaction α to the argument x . Nevertheless, in the particular case where there is at most one support to α , it may be considered that deductive and necessary meanings coincide⁸.

That leads to the following definitions.

Def. 28 Let $\text{ASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system with recursive interactions. We define:

$$\begin{aligned} \mathcal{V} = & \{a | a \in \mathbf{A}\} \cup \{\text{Pra} | a \in \mathbf{A}\} \cup \{\text{Ina} | a \in \mathbf{A}\} \cup \\ & \{A\alpha | \alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}) \text{ and } \exists \beta \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}) \text{ s.t. } \beta = (x, \alpha)\} \cup \\ & \{I\alpha | \alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}) \text{ and } \exists \beta \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}) \text{ s.t. } \beta = (x, \alpha)\} \end{aligned}$$

\mathcal{L} is the language defined from propositional logic using \mathcal{V} as the set of propositional variables.

Note that the meaning of the chosen propositional variables corresponds to that given in the previous section completed by:

- the variable $A\alpha$ means that the interaction α is active,
- the variable $I\alpha$ means that the interaction α is inhibited.

Def. 29 (Logical translation of ASAF) Let $\text{ASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar recursive argumentation system, in which there exists at most one support to a given interaction. Let $a, b, c \in \mathbf{A}$. Let $\alpha, \beta \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}}$.

- A necessary support between two arguments $a \xrightarrow{\alpha} b$ is translated into the two following propositional formulas of \mathcal{L} :
 - $(A\alpha \wedge b) \rightarrow \text{Pra}$ and
 - $\text{Pra} \rightarrow a$.
- An attack between two arguments $a \xrightarrow{\alpha} b$ is translated into the two following propositional formulae of \mathcal{L} :
 - $(A\alpha \wedge a) \rightarrow \text{Inb}$ and
 - $\text{Inb} \rightarrow \neg b$.
- Let $\alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ be an interaction neither supported, nor attacked. In this case, $A\alpha \leftrightarrow \top$ and $I\alpha \leftrightarrow \perp$ (so $A\alpha$ is always true whereas $I\alpha$ is always false).

⁸This equivalence actually holds in a MIM, where the context is reduced to only one activation and only one inhibition. The context $\mathcal{C} = \langle P \longrightarrow, Q \longleftarrow \rangle$ of the reaction r defines all the conditions under which the reaction r can occur. So the first part may be interpreted as follows: the presence of P activates the reaction r and only the presence of P activates the reaction r .

- Let $\alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ be an interaction necessarily supported by $\beta = (c, \alpha)$. In this case:
 - $A\alpha \leftrightarrow (c \wedge A\beta)$ and
 - $I\alpha \leftrightarrow (\neg c \vee I\beta)$.
- Let $\alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ be an interaction attacked by $\beta = (c, \alpha)$. In this case:
 - $A\alpha \leftrightarrow (\neg c \vee I\beta)$ and
 - $I\alpha \leftrightarrow (c \wedge A\beta)$.
- Let $\alpha \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ be an interaction necessarily supported by $\beta = (c, \alpha)$ and attacked by $\delta = (d, \alpha)$. In this case:
 - $A\alpha \leftrightarrow ((c \wedge A\beta) \wedge (\neg d \vee I\delta))$ and
 - $I\alpha \leftrightarrow ((\neg c \vee I\beta) \vee (d \wedge A\delta))$.

Def. 30 (The knowledge base associated with an ASAF) Let $\text{ASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar recursive argumentation system, in which there exists at most one support to a given interaction.

The knowledge base associated with ASAF gives the translation of ASAF into formulae of \mathcal{L} . It is denoted by Σ and defined by:

$$\Sigma = \Sigma_1 \cup \Sigma_2$$

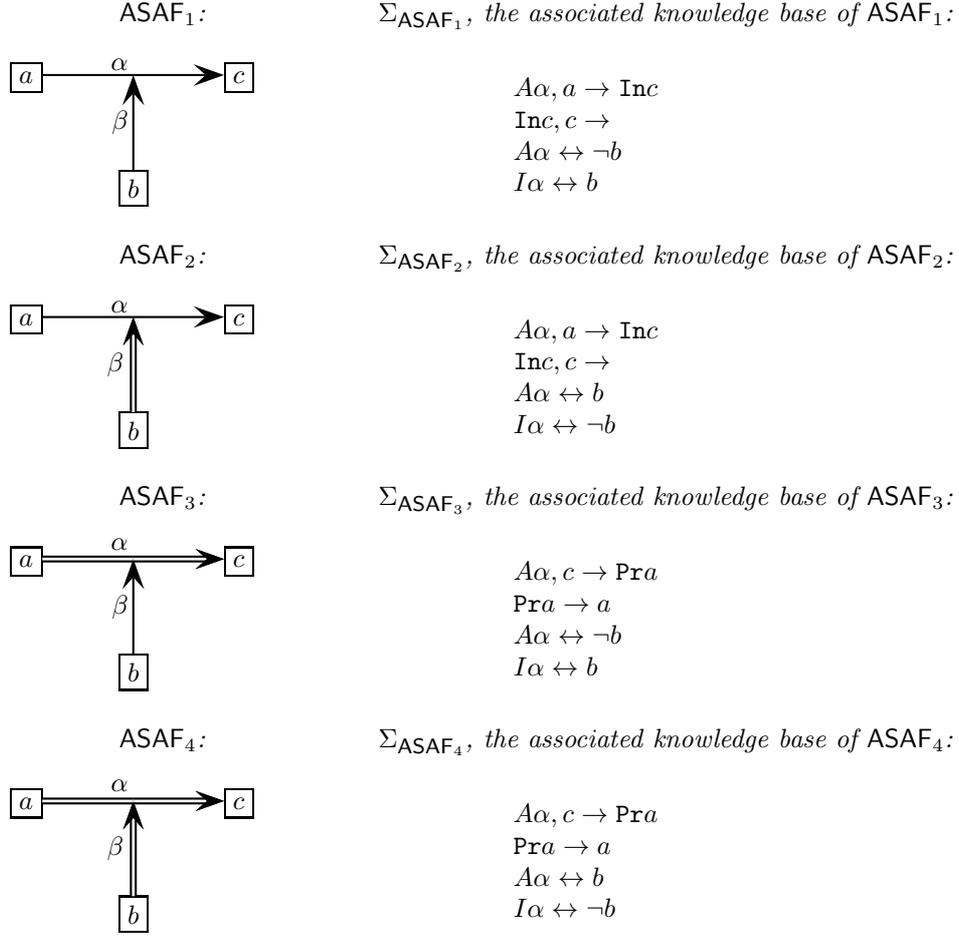
where Σ_1 gives the translation for non recursive interactions whereas Σ_2 gives the translation for recursive interactions. Σ_1 and Σ_2 are defined by:

$$\begin{aligned} \Sigma_1 = & \{(A\alpha \wedge b) \rightarrow \text{Pra} \mid \alpha = (a, b) \in \mathbf{R}_{\text{sup}}\} \\ & \cup \{(A\alpha \wedge a) \rightarrow \text{Inb} \mid \alpha = (a, b) \in \mathbf{R}_{\text{att}}\} \\ & \cup \{\text{Pra} \rightarrow a \mid a \in \mathbf{A} \text{ and } \exists x \in \mathbf{A} \text{ s.t. } (a, x) \in \mathbf{R}_{\text{sup}}\} \\ & \cup \{\text{Ina} \rightarrow \neg a \mid a \in \mathbf{A} \text{ and } \exists x \in \mathbf{A} \text{ s.t. } (x, a) \in \mathbf{R}_{\text{att}}\} \end{aligned}$$

$$\begin{aligned} \Sigma_2 = & \{(A\alpha \leftrightarrow \top) \wedge (I\alpha \leftrightarrow \perp) \mid \alpha \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}} \\ & \text{and } \nexists \beta = (c, \alpha) \in \mathbf{R}_{\text{sup}} \\ & \text{and } \nexists \delta = (d, \alpha) \in \mathbf{R}_{\text{att}}\} \\ & \cup \{(A\alpha \leftrightarrow (c \wedge A\beta)) \wedge (I\alpha \leftrightarrow (\neg c \vee I\beta)) \mid \alpha \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}} \\ & \text{and } \exists \beta = (c, \alpha) \in \mathbf{R}_{\text{sup}} \\ & \text{and } \nexists \delta = (d, \alpha) \in \mathbf{R}_{\text{att}}\} \\ & \cup \{(A\alpha \leftrightarrow (\neg d \vee I\delta)) \wedge (I\alpha \leftrightarrow (d \wedge A\delta)) \mid \alpha \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}} \\ & \text{and } \nexists \beta = (c, \alpha) \in \mathbf{R}_{\text{sup}} \\ & \text{and } \exists \delta = (d, \alpha) \in \mathbf{R}_{\text{att}}\} \\ & \cup \{(A\alpha \leftrightarrow ((c \wedge A\beta) \wedge (\neg d \vee I\delta))) \wedge (I\alpha \leftrightarrow ((\neg c \vee I\beta) \vee (d \wedge A\delta))) \mid \alpha \in \mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}} \\ & \text{and } \exists \beta = (c, \alpha) \in \mathbf{R}_{\text{sup}} \\ & \text{and } \exists \delta = (d, \alpha) \in \mathbf{R}_{\text{att}}\} \end{aligned}$$

The following example illustrates the different cases encountered in the above definition.

Ex. 12 We consider different ASAFs and their associated knowledge bases.



We have proposed a logical encoding of ASAF inspired by the logical handling of MIM. This parallel has required the restriction to ASAF containing at most one support to a given interaction. However, it could be possible to go further by generalizing the notion of context. For instance each part of the context could contain several context reactions. The activation and inhibition expressions should be defined accordingly. This is a topic for further research.

As done for bipolar argumentation graphs, we are now interested in exploiting the logical knowledge base associated with a recursive bipolar graph.

In the following, we propose some preliminary results towards that research direction.

Of course, Prop. 3 still holds in the case of an ASAF (a model of Σ can be obtained by assigning each variable p_i , each variable $\text{In}p_i$ and each variable $\text{Pr}p_i$ to *false*).

The following results describe situations in which inconsistency may appear:

Prop. 10 Let $\text{ASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be an ASAF. Let Σ be its associated knowledge base.

1. If there exists $\alpha = (a, c) \in \mathbf{R}_{\text{att}}$ and $\nexists(b, \alpha) \in (\mathbf{R}_{\text{att}} \cup \mathbf{R}_{\text{sup}})$ then $\Sigma \cup \{a, c\}$ is inconsistent.
2. If there exists $\alpha = (a, c) \in \mathbf{R}_{\text{att}}$ and $\exists(b, \alpha) \in \mathbf{R}_{\text{att}}$ and $\nexists(d, \alpha) \in \mathbf{R}_{\text{sup}}$ then $\Sigma \cup \{a, c, \neg b\}$ is inconsistent.
3. If there exists $\alpha = (a, c) \in \mathbf{R}_{\text{att}}$ and $\nexists(b, \alpha) \in \mathbf{R}_{\text{att}}$ and $\exists(d, \alpha) \in \mathbf{R}_{\text{sup}}$ then $\Sigma \cup \{a, c, d\}$ is inconsistent.
4. If there exists $\alpha = (a, c) \in \mathbf{R}_{\text{att}}$ and $\exists(b, \alpha) \in \mathbf{R}_{\text{att}}$ and $\exists(d, \alpha) \in \mathbf{R}_{\text{sup}}$ then $\Sigma \cup \{a, c, \neg b, d\}$ is inconsistent.

Proof:

1. Σ contains the formulas $a \rightarrow \text{Inc}$, $\text{Inc}, c \rightarrow$. So $\Sigma \cup \{a, c\}$ is inconsistent.
2. Σ contains the formulas $A\alpha, a \rightarrow \text{Inc}$, $\text{Inc}, c \rightarrow$, $A\alpha \leftrightarrow \neg b$. So $\Sigma \cup \{a, c, \neg b\}$ is inconsistent.
3. Σ contains the formulas $A\alpha, a \rightarrow \text{Inc}$, $\text{Inc}, c \rightarrow$, $A\alpha \leftrightarrow d$. So $\Sigma \cup \{a, c, d\}$ is inconsistent.
4. Σ contains the formulas $A\alpha, a \rightarrow \text{Inc}$, $\text{Inc}, c \rightarrow$, $A\alpha \leftrightarrow (\neg b \wedge d)$. So $\Sigma \cup \{a, c, \neg b, d\}$ is inconsistent.

□

Unfortunately, there is no simple characterization of subsets S such that $\Sigma \cup S$ is inconsistent (as stated in Prop. 6).

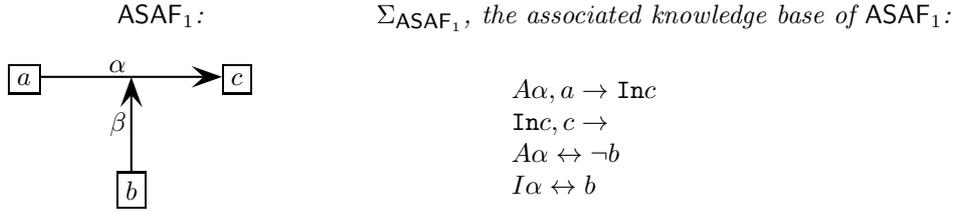
However interesting results can be derived by considering the **A**-prime implicates of the knowledge base Σ . Let us recall the definition of **A**-prime implicate of Σ (for instance, see [18]).

Def. 31 Let Σ be a propositional knowledge base built over a set \mathcal{V} of propositional variables. Let W be a subset of \mathcal{V} . Let π be a clause built on W (it is called a W -clause). π is a W -prime implicate of Σ iff

1. Σ infers π (π is called an implicate of Σ)
2. If π' is a W -clause such that π' is an implicate of Σ and π' infers π , then π infers π' .

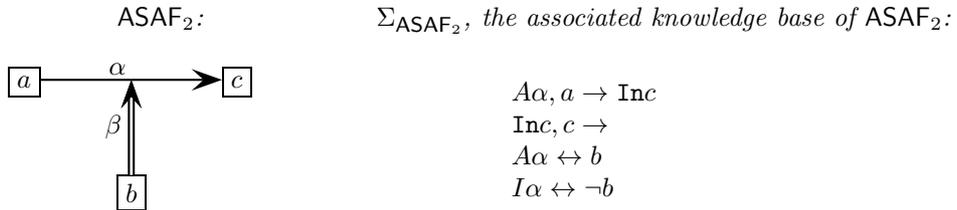
The above results can be illustrated on Ex. 12.

Ex. 12 (cont'd) We consider different ASAF and their associated knowledge base.



We have:

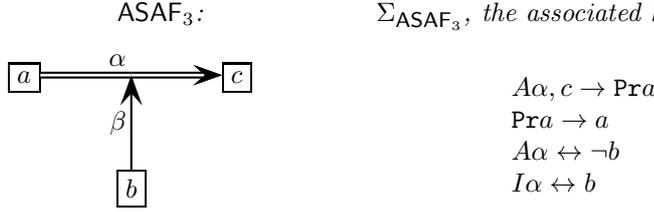
- Any interpretation in which the truth value of b is true (β is active) or a is false (a is not accepted) is a model of Σ_{ASAF_1} (in these cases, a and c can be accepted together since α is not active).
- The unique **A**-prime implicate of Σ_{ASAF_1} is $\neg a \vee b \vee \neg c$.
- As a consequence, the interpretation in which a is true (α is active) and b is false (β is not active) is a model of Σ_{ASAF_1} only if c is false (c is not accepted). In other words $\Sigma_{\text{ASAF}_1} \cup \{a, c, \neg b\}$ is inconsistent (second case of Prop. 10).



We have:

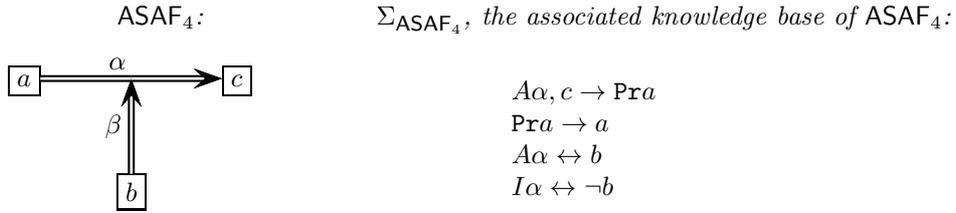
- Any interpretation in which the truth value of b is false (β is not active) or a is false (a is not accepted) is a model of Σ_{ASAF_2} (in these cases, a and c can be accepted together since α is not active).
- The unique **A**-prime implicate of Σ_{ASAF_2} is $\neg a \vee \neg b \vee \neg c$.

- As a consequence, the interpretation in which a and b are true (α is active and β is also active) is a model of Σ_{ASAF_2} only if the truth value of c is false (c is not accepted). In other words $\Sigma_{\text{ASAF}_2} \cup \{a, c, b\}$ is inconsistent (third case of Prop. 10).



We have:

- Any interpretation in which the truth value of b is true (β is active) or c is false (c is not accepted) is a model of Σ_{ASAF_3} (in these cases, a can be accepted even if c is not accepted since α is not active).
- The unique **A**-prime implicate of Σ_{ASAF_3} is $a \vee b \vee \neg c$.
- As a consequence, the interpretation in which the truth value of c is true and the truth value of b is false (α is active and β is not active) is a model of Σ_{ASAF_3} only if the truth value of a is true (a is accepted). In other words, $\Sigma_{\text{ASAF}_3} \cup \{\neg a, c, \neg b\}$ is inconsistent.



We have:

- Any interpretation in which the truth value of b is false (β is not active) or c is false (c is not accepted) is a model of Σ_{ASAF_4} (in these cases, a can be accepted even if c is not accepted since α is not active).
- The unique **A**-prime implicate of Σ_{ASAF_4} is $a \vee \neg b \vee \neg c$.
- As a consequence, the interpretation in which the truth values of c and b are true (α and β are valid) is a model of Σ_{ASAF_4} only if the truth value of a is true (a is accepted). In other words, $\Sigma_{\text{ASAF}_4} \cup \{\neg a, c, b\}$ is inconsistent.

6 Some comparison results

6.1 Logical representation of BAS vs axiomatisation

Consider a necessary BAS and the constraints proposed in [11] and recalled in Section 2.2. These constraints aim at describing the desired behavior of a necessary BAS. An important issue is to check whether these constraints can be satisfied in the logical representation of the BAS.

Transitivity (TRA) Transitivity of the necessary support is satisfied as shown by the following result:

Prop. 11 Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system. Let Σ be its associated knowledge base. $\forall a, b \in \mathbf{A}$, if $\exists n > 1$ such that $a = a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n = b$ then Σ infers $(b \rightarrow \text{Pra})$.

Proof: Consider $a, b \in \mathbf{A}$, such that $\exists n > 1$ such that $a = a_1 \mathbf{R}_{\text{sup}} \dots \mathbf{R}_{\text{sup}} a_n = b$. So in Σ , we have the following formulae: $b = a_n \rightarrow \text{Pr} a_{n-1}, \dots, a_2 \rightarrow \text{Pr} a$ and $\forall a_i, \text{Pr} a_i \rightarrow a_i$. So Σ infers $b \rightarrow \text{Pr} a$ and $\text{Pr} a \rightarrow a$ which encodes a necessary support from a to b . \square

Closure (CLO) The property of closure concerns extensions (under a given semantics). There is no equivalent notion in the logical representation. However, as discussed in Sect. 5.1, the basic requirement for extensions, conflict-freeness, exactly corresponds to Σ -consistency. So, the following result ensures a kind of closure.

Prop. 12 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system. Let Σ be its associated knowledge base. $\forall a, b \in \mathbf{A}$, if $a \mathbf{R}_{\text{sup}} b$ then $\Sigma \cup \{b\}$ infers a .*

Proof: Consider $a, b \in \mathbf{A}$, such that $a \mathbf{R}_{\text{sup}} b$. So in Σ , we have the following formulae: $b \rightarrow \text{Pr} a$ and $\text{Pr} a \rightarrow a$. So $\Sigma \cup \{b\}$ infers a . \square

Indeed, this result enables to prove that each model of Σ satisfying b also satisfies a .

Conflicting sets (CFS) As a particular case of Prop. 5, we obtain the following result showing that the constraint is satisfied:

Prop. 13 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system. Let Σ be its associated knowledge base. $\forall a, b, c \in \mathbf{A}$, if $(a \mathbf{R}_{\text{att}} c$ and c supports $b)$ or $(c \mathbf{R}_{\text{att}} b$ and c supports $a)$ then $\Sigma \cup \{a, b\}$ is inconsistent.*

Addition of new attacks (nATT, n+ATT) Due to results obtained in Sect. 5.1, especially in Prop. 8 and Prop. 9, these constraints are not satisfied. More precisely, we have:

Prop. 14 *Let $\text{BAS} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$ be a bipolar argumentation system. Let Σ be its associated knowledge base. $\forall a, b, c \in \mathbf{A}$,*

- *if $a \mathbf{R}_{\text{att}} c$ and $c \mathbf{R}_{\text{sup}} b$ then Σ does not infer $(a \rightarrow \text{In} b)$ (i.e. the attack (a, b) is not inferred by Σ);*
- *if $c \mathbf{R}_{\text{att}} b$ and $c \mathbf{R}_{\text{sup}} a$ then Σ infers $(a \rightarrow \text{In} b)$ (i.e. the attack (a, b) is inferred by Σ).*

Note that the logical representation of BAS differentiates the Case 2-extended attacks (namely the secondary attacks), since they are more difficult to determine.

6.2 Comparing two logical representations of ASAF

In Sect. 5.1, we can find a logical encoding of a BAS and a direct logical encoding of an ASAF. In Sect. 3.2, a translation of an ASAF into a necessary BAS has been described. So, there are two ways for obtaining a logical representation of an ASAF. The purpose of this section is to compare these two ways.

Consider $\text{ASAF} = \langle \mathbf{A}, \mathbf{R}_{\text{att}}, \mathbf{R}_{\text{sup}} \rangle$, the associated BAS will be denoted by $\text{BAS}' = \langle \mathbf{A}', \mathbf{R}_{\text{att}}', \mathbf{R}_{\text{sup}}' \rangle$ (Def. 20). Let Σ denote the knowledge base associated with ASAF (Def. 29) and Σ' denote the base associated with BAS' (Def. 27). First, we compare the models of Σ with those of Σ' .

Ex. 12 (cont'd) *Case of ASAF_1*

$$\Sigma_{\text{ASAF}_1} = \begin{cases} \{A\alpha, a \rightarrow \text{Inc}, \\ \text{Inc}, c \rightarrow, \\ A\alpha \leftrightarrow \neg b, \\ I\alpha \leftrightarrow b\} \end{cases} \quad \Sigma'_{\text{ASAF}_1} = \begin{cases} \{\alpha \rightarrow \text{Pr} a, & \beta \rightarrow \text{Pr} b, \\ \text{Pr} a \rightarrow a, & \text{Pr} b \rightarrow b, \\ \alpha \rightarrow \text{Inc}, & \beta \rightarrow \text{In} \alpha, \\ \text{Inc}, c \rightarrow, & \text{In} \alpha, \alpha \rightarrow\} \end{cases}$$

The unique \mathbf{A} -prime implicate of Σ_{ASAF_1} is $\neg a \vee b \vee \neg c$.

We want to compare the models of Σ_{ASAF_1} and those of Σ'_{ASAF_1} . Let $\mathcal{V}_{\text{ASAF}_1}$ (resp. $\mathcal{V}'_{\text{ASAF}_1}$) be the set of propositional variables appearing in Σ_{ASAF_1} (resp. Σ'_{ASAF_1}).

A model $\mathcal{I}_{\text{ASAF}_1}$ of Σ_{ASAF_1} can be easily “extended into” a model $\mathcal{I}'_{\text{ASAF}_1}$ of Σ'_{ASAF_1} considering that:

- $\forall x \in (\mathcal{V}_{\text{ASAF}_1} \cap \mathcal{V}'_{\text{ASAF}_1}), \mathcal{I}'_{\text{ASAF}_1}(x) = \mathcal{I}_{\text{ASAF}_1}(x)$ and
- $\forall x \in \{\text{Pra}, \text{Prb}, \alpha, \beta, \text{In}\alpha\}, \mathcal{I}'_{\text{ASAF}_1}(x) = \text{false}$.

However, the converse does not hold. Indeed, consider $\mathcal{I}'_{\text{ASAF}_1}$ a model of Σ'_{ASAF_1} such that $\mathcal{I}'_{\text{ASAF}_1}(a) = \mathcal{I}'_{\text{ASAF}_1}(c) = \text{true}$ and $\mathcal{I}'_{\text{ASAF}_1}(b) = \mathcal{I}'_{\text{ASAF}_1}(\text{In}c) = \text{false}$ (all the other variables being assigned to false). Obviously, this model cannot be extended to any model of Σ_{ASAF_1} since Σ_{ASAF_1} infers $\neg a \vee b \vee \neg c$. This models corresponds to the fact that the attack α is inactive whereas its source argument is accepted and the interaction β that attacks α is also inactive. It is surprising since, in this case, α should be active. The fact that b is false implies that β is false (β is inactive). However, it does not imply that α is true, as would be expected. Indeed, there could be another attack against α . The logical encoding of the attack from β to α does not completely capture the meaning of “being active” in that particular case where there is only one attack against α (the one from β).

The above example raises several issues which deserve further investigation:

1. With the first logical encoding producing Σ_{ASAF_1} , we have encoded the equivalence $A\alpha \leftrightarrow \neg b$. The part $A\alpha \rightarrow \neg b$ of this equivalence is used for finding the unique \mathbf{A} -prime implicate of Σ_{ASAF_1} and for restricting the models. The equivalence is justified by the fact that there exists at most one attack against α (following the MIM approach). Whereas in the second encoding producing Σ'_{ASAF_1} , we just encode the implication $\beta \rightarrow \text{In}\alpha$ and we do not encode the fact that if β is inactive then α is active (or equivalently if α is inactive then β is active). Indeed, we have used the translation of an ASAF into a BAS. So we have replaced a context by a meta-argument (α) before encoding the attack (β, α). However, in argumentation semantics, the meaning of an attack (a, b) is only that if a is accepted then b is not accepted. If a is not accepted, b could be attacked by another argument, so it might be not accepted.
2. The semantics of argumentation graphs encoded in logic cannot be completely captured by the notion of model. So a comparison between two logical bases needs not only to compare the models, but also to find a mapping from particular models and to compare particular subsets of the bases.
3. In a recent work about recursive interactions in bipolar argumentation (see [7]), it has been shown that two notions should be considered for an interaction : validity and groundness. Roughly speaking, an interaction $\alpha = (a, b)$ is active if it is grounded and valid. It is grounded if its source a is accepted. It is valid if any attacker is inactive and any (necessary) supporter is active. It would be interesting to study a logical encoding for all these notions.

7 Conclusion and future works

In this paper, we have presented a preliminary study about logical encodings of argumentation graphs *themselves*. This work is inspired by a similar approach in bioinformatics that consists in logically encoding molecular networks (called MIM). Indeed, some correspondences can be established between these maps and argumentation graphs where two different interactions appear (attacks and supports) and where some interactions are recursive (*i.e.* an interaction can be defined between either two arguments, or an argument and another interaction).

We have also given some properties that allow for a comparison between different approaches for taking into account bipolar and recursive interactions.

As future works, we propose to:

- Complete the study of the proposed translations.
- Restate the main concepts used in abstract argumentation (particularly, the different notions of acceptability) in terms of logical issues and vice-versa.
- Propose efficient algorithms for encoding these concepts.

References

- [1] J. M. Alliot, R. Demolombe, L. Fariñas del Cerro, M. Diéguez, and N. Obeid. Reasoning on molecular interaction maps. In *7th European Symposium on Computational Intelligence and Mathematics (ESCIM'15)*, pages 263–269, Cádiz, Spain, 2015. Available at: <http://escim2015.uca.es/wp-content/uploads/2015/02/ESCIM2015-FINAL.pdf>.
- [2] 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.
- [3] L. Amgoud, C. Cayrol, M-C. Lagasque-Schiex, and P. Livet. On bipolarity in argumentation frameworks. *Intl. Journal of Intelligent Systems*, 23:1062–1093, 2008.
- [4] L. Amgoud, N. Maudet, and S. Parsons. Modelling dialogues using argumentation. In *Proc. of ICMAS*, pages 31–38, 2000.
- [5] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. AFRA: Argumentation framework with recursive attacks. *International Journal of Approximate Reasoning*, 52:19–37, 2011.
- [6] 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.
- [7] C. Cayrol, A. Cohen, and M-C. Lagasque-Schiex. Towards a new framework for recursive interactions in abstract bipolar argumentation. Technical Report RR- -2016- -03- -FR, IRIT, 2016.
- [8] C. Cayrol and M-C. Lagasque-Schiex. On the acceptability of arguments in bipolar argumentation frameworks. In *Proc. of ECSQARU*, pages 378–389, 2005.
- [9] C. Cayrol and M-C. Lagasque-Schiex. Coalitions of arguments: a tool for handling bipolar argumentation frameworks. *Intl. J. of Intelligent Systems*, 25:83–109, 2010.
- [10] C. Cayrol and M-C. Lagasque-Schiex. Bipolarity in argumentation graphs: towards a better understanding. *IJAR*, 54(7):876–899, 2013.
- [11] Claudette Cayrol and Marie-Christine Lagasque-Schiex. An axiomatic approach to support in argumentation. In *Proceedings of the workshop TAFA (LNAI 9524)*, pages 74–91, 2015.
- [12] 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*, 2014.
- [13] Andrea Cohen, Sebastian Gottifredi, Alejandro J García, and Guillermo R Simari. An approach to abstract argumentation with recursive attack and support. *Journal of Applied Logic*, 2014.

- [14] Robert Demolombe, Luis Fariñas del Cerro, and Najj Obeid. Automated Reasoning in Metabolic Networks with Inhibition (regular paper). In *International Conference of the Italian Association for Artificial Intelligence (AI*IA), Turin, Italy, 04/12/2013-06/12/2013*, volume 8249 of *Lecture Notes in Computer Science*, pages 37–47, <http://www.springerlink.com>, décembre 2013. Springer.
- [15] Robert Demolombe, Luis Fariñas del Cerro, and Najj Obeid. Molecular Interaction Automated Maps (regular paper). In *International Workshop on Learning and Non Monotonic Reasoning (LNMR), A Corunna, Spain, 15/09/2013*, volume abs/1311.4639, pages 31–42, <http://arxiv.org/corr>, septembre 2013. Computing Research Repository.
- [16] 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.
- [17] N. Karacapilidis and D. Papadias. Computer supported argumentation and collaborative decision making: the HERMES system. *Information systems*, 26(4):259–277, 2001.
- [18] P. Marquis. Consequence finding algorithms. In *Handbook on Defeasible Reasoning and Uncertainty Management Systems (Vol. 5: Algorithms for Defeasible and Uncertain Reasoning)*, pages 41–145. Kluwer Academic, 2000.
- [19] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artificial Intelligence*, 173:901–934, 2009.
- [20] F. Nouioua. AFs with necessities: further semantics and labelling characterization. In *Proc. of SUM*, 2013.
- [21] F. Nouioua and V. Risch. Bipolar argumentation frameworks with specialized supports. In *Proc. of ICTAI*, pages 215–218. IEEE Computer Society, 2010.
- [22] F. Nouioua and V. Risch. Argumentation frameworks with necessities. In *Proc. of SUM*, pages 163–176. Springer-Verlag, 2011.
- [23] N. Oren and T. J. Norman. Semantics for evidence-based argumentation. In *Proc. of COMMA*, pages 276–284, 2008.
- [24] N. Oren, C. Reed, and M. Luck. Moving between argumentation frameworks. In *Proc. of COMMA*, pages 379–390. IOS Press, 2010.
- [25] S. Polberg and N. Oren. Revisiting support in abstract argumentation systems. In *Proc. of COMMA*, pages 369–376. IOS Press, 2014.
- [26] H. Prakken. On support relations in abstract argumentation as abstraction of inferential relations. In *Proc. of ECAI*, pages 735–740, 2014.
- [27] B. Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *Journal of Logic in Computation*, 13:319–346, 2003.