On Input/Output Argumentation Frameworks

Pietro BARONI [a,1], Guido BOELLA [b] Federico CERUTTI [a], Massimiliano GIACOMIN [a] Leendert VAN DER TORRE [c] and Serena VILLATA [d]

[a] *University of Brescia, Italy*
[b] *University of Torino, Italy*
[c] *University of Luxembourg, Luxembourg*
[d] *INRIA Sophia Antipolis, France*

Abstract. This paper introduces Input/Output Argumentation Frameworks, a novel approach to characterize the behavior of an argumentation framework as a sort of black box exposing a well-defined external interface. As a starting point, we define the novel notion of semantics decomposability and analyze complete, stable, grounded and preferred semantics in this respect. Then we show as a main result that, under grounded, stable and credulous preferred semantics, Input/Output Argumentation Frameworks with the same behavior can be interchanged without affecting the result of semantics evaluation of other arguments interacting with them.

Keywords. Argumentation frameworks, Decomposability, Equivalence

Introduction

Dung's argumentation framework [6] is a widely adopted formalism to represent "reasoning situations" where conflicts are present and to analyze their properties at an essential and abstract level. In the original treatment argumentation frameworks are regarded as monolithic entities and, as such, are suitable to deal with static and complete situations only. To go beyond this basic setting, approaches [5,7,3] considering portions of argumentation frameworks and interactions among them are receiving increasing attention.

This paper provides a systematic treatment of "partial" argumentation frameworks and the relevant semantic issues, by introducing the notion of *input/output (I/O) argumentation framework*, namely a formal approach to characterize the behavior of an argumentation framework in terms of a well-defined interface connecting it with other "similar devices". In this perspective, two I/O argumentation frameworks are *equivalent* if given the same input they produce the same output. The fundamental question is then whether equivalence is sufficient for interchangeability of I/O argumentation frameworks, or, in other terms, whether an I/O argumentation framework can be seen as a reusable "black box" which can be freely combined with other ones. We show that the

[1] Corresponding Author: Pietro Baroni, Dept Information Engineering, University of Brescia, Via Branze 38, 25123 Brescia, Italy; E-mail: pietro.baroni@ing.unibs.it.

equivalence relation needs to be semantics specific and that, under stable, grounded and credulous preferred semantics, the above question has a positive answer.

Among the applications of this research idea we mention summarization and combination. In a nutshell, summarization involves the ability to replace some articulated set of arguments (and attacks involving them) with a simpler pattern which is equivalent as far as the interaction with other arguments is concerned: this will help the user to focus on the set of arguments she is interested in, without being cluttered with unnecessary details. With some similarity, combination involves putting together several "argumentation patterns" [10], i.e. modular components able to synthesize prototypical reasoning schemes to be reused in articulated contexts.

The paper is organised as follows. After recalling the necessary background in Section 1, in Section 2 we define the key notion of *decomposability* of a semantics, and analyze decomposability of complete, stable, grounded and preferred semantics. In Section 3 we introduce input/output (I/O) argumentation frameworks and their equivalence, and show that equivalent I/O argumentation frameworks are interchangeable wrt grounded, stable and credulous preferred semantics. Section 4 draws some comparisons with related works and concludes the paper. Proofs are omitted due to space limitations.

1. Background

We follow the traditional definition of argumentation framework introduced in [6] and define its restriction to a subset of its arguments.

Definition 1 *An* argumentation framework *is a pair* $AF = (Ar, att)$ *in which* Ar *is a finite set of arguments and* $att \subseteq Ar \times Ar$. *Given a set* $Args \subseteq Ar$, *the restriction of* AF *to* $Args$, *denoted as* $AF{\downarrow}_{Args}$ *is the argumentation framework* $(Args, att \cap (Args \times Args))$.

In this paper we use the labelling-based approach to the definition of argumentation semantics [4,1]. For technical reasons, we define labellings both for argumentation frameworks and for arbitrary sets of arguments.

Definition 2 *Let* $AF = (Ar, att)$ *be an argumentation framework. A* labelling *of* AF *is a total function* $Lab : Ar \longrightarrow \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$. *The set of all* labellings *of* AF *will be denoted as* $\mathfrak{L}(AF)$. *Given a set of arguments* $Args$, *a* labelling *of* $Args$ *is a total function* $Lab : Args \longrightarrow \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\}$. *The set of all* labellings *of* $Args$ *will be denoted as* $\mathfrak{L}_{Args}$. *Given a labelling* Lab, *we write* $\mathtt{in}(Lab)$ *for* $\{A \mid Lab(A) = \mathtt{in}\}$, $\mathtt{out}(Lab)$ *for* $\{A \mid Lab(A) = \mathtt{out}\}$ *and* $\mathtt{undec}(Lab)$ *for* $\{A \mid Lab(A) = \mathtt{undec}\}$. *The restriction of* Lab *to a set of arguments* $Args \subseteq Ar$, *denoted as* $Lab{\downarrow}_{Args}$ *is defined as* $Lab \cap (Args \times \{\mathtt{in}, \mathtt{out}, \mathtt{undec}\})$.

Definition 3 *Given an argumentation framework* $AF = (Ar, att)$, *a labelling-based semantics* $\mathbf{S}$ *associates with* AF *a subset of* $\mathfrak{L}(AF)$, *denoted as* $\mathbf{L_S}(AF)$. *An argument* A *is* skeptically justified *under* $\mathbf{S}$ *if* $\forall Lab \in \mathbf{L_S}(AF)\ Lab(A) = \mathtt{in}$; *an argument* A *is* credulously justified *under* $\mathbf{S}$ *if* $\exists Lab \in \mathbf{L_S}(AF) : Lab(A) = \mathtt{in}$.

In the labelling-based approach, semantics definition relies on some basic *legality* constraints relating the label of an argument to those of its attackers.

Definition 4 *Let Lab be a labelling of the argumentation framework (Ar, att). An* `in`*-labelled argument is said to be* legally `in` *iff all its attackers are labelled* `out`*. An* `out`*-labelled argument is said to be* legally `out` *iff it has at least one attacker that is labelled* `in`*. An* `undec`*-labelled argument is said to be* legally `undec` *iff not all its attackers are labelled* `out` *and it does not have an attacker that is labelled* `in`*.*

We can now introduce the definitions of labellings corresponding to traditional complete and stable semantics.

Definition 5 *A* complete labelling *is a labelling where every* `in`*-labelled argument is legally* `in`*, every* `out`*-labelled argument is legally* `out` *and every* `undec` *labelled argument is legally* `undec`*. A* stable labelling *is a complete labelling without* `undec`*-labelled arguments.*

To simplify the technical treatment in the following, we introduce the labelling-based definition of grounded and preferred semantics by referring to the commitment relation between labellings [1].

Definition 6 *Let Lab_1 and Lab_2 be two labellings: we say that Lab_2 is* more or equally committed *than Lab_1 ($Lab_1 \sqsubseteq Lab_2$) iff $\mathtt{in}(Lab_1) \subseteq \mathtt{in}(Lab_2)$ and $\mathtt{out}(Lab_1) \subseteq \mathtt{out}(Lab_2)$. Let $AF = (Ar, att)$ be an argumentation framework. The grounded labelling of AF is the minimal (w.r.t. $\sqsubseteq$) labelling among all complete labellings. A preferred labelling of AF is a maximal (w.r.t. $\sqsubseteq$) labelling among all complete labellings.*

Complete, stable, grounded and preferred semantics will be denoted in the following as **CO**, **ST**, **GR**, **PR**, respectively.

2. Semantics Decomposability

In order to define the notion of semantics decomposability, we need to introduce a formal setting for arbitrary partitions of argumentation frameworks and their labellings. Given $AF = (Ar, att)$ and any subset $Args$ of its arguments, the two partial frameworks induced by $Args$ and $Ar \setminus Args$ mutually interact and affect each other. We need therefore to formalize the elements directly affecting a set of arguments $Args$: these consist of the arguments attacking $Args$ from outside, called *input* arguments, and of the attack relation from the input arguments to $Args$, called *conditioning relation*.

Definition 7 *Given $AF = (Ar, att)$ and a set $Args \subseteq Ar$, the* input *of $Args$, denoted as $Args^{in}$, is the set $\{B \in Ar \setminus Args \mid \exists A \in Args, (B, A) \in att\}$, the* conditioning relation *of $Args$, denoted as R_{Args}, is defined as $att \cap (Args^{in} \times Args)$.*

We now consider the issue of labelling a (partial) argumentation framework affected by a set of external input arguments and the relevant attacks from them. The basic idea is that for each labelling of the input arguments a set of labellings of the affected argumentation framework is determined. This is formalized through the notion of *base function*.

Definition 8 *Given a tuple including an argumentation framework $AF = (Ar, att)$, a set of arguments I such that $I \cap Ar = \emptyset$, a labelling $L_I \in \mathfrak{L}_I$ and a relation $R_I \subseteq I \times Ar$, a* base function *associates with the tuple a set of labellings of AF, i.e. $F(AF, I, L_I, R_I) \in 2^{\mathfrak{L}(AF)}$.*

Semantics decomposability can then be defined with respect to an arbitrary partition of an argumentation framework induced by a set of arguments P: this gives rise to two restricted argumentation frameworks, namely $AF{\downarrow_P}$ and $AF{\downarrow_{Ar \setminus P}}$, affecting each other with the relevant input arguments and conditioning relations as stated in Definition 8. Decomposability of a semantics $\mathbf{S}$ requires that $\mathbf{S}$ can be put in correspondence with a base function $F_{\mathbf{S}}$ such that: i) every labelling prescribed by $\mathbf{S}$ on AF, namely every element of $\mathbf{L_S}(AF)$, corresponds to the union of two "compatible" labellings L_P and $L_{Ar \setminus P}$ of the restricted argumentation frameworks, both obtained applying $F_{\mathbf{S}}$; ii) in turn, each union of two "compatible" labellings L_P and $L_{Ar \setminus P}$ obtained applying $F_{\mathbf{S}}$ to the restricted frameworks gives rise to a labelling of AF. The "compatibility" constraint mentioned above reflects the fact that the labelling of the restricted framework is used by $F_{\mathbf{S}}$ for computing the other one. This means that L_P and $L_{Ar \setminus P}$ are "compatible" if L_P is produced by $F_{\mathbf{S}}$ for $AF{\downarrow_P}$ with the input arguments P^{in} labelled according to $L_{Ar \setminus P}$ and viceversa if $L_{Ar \setminus P}$ is produced by $F_{\mathbf{S}}$ for $AF{\downarrow_{Ar \setminus P}}$ with the input arguments $(Ar \setminus P)^{\mathrm{in}}$ labelled according to L_P. Definition 9 synthesizes these considerations.

Definition 9 *A semantics* $\mathbf{S}$ *is* decomposable *iff there is a base function* $F_{\mathbf{S}}$ *such that for any argumentation framework* $AF = (Ar, att)$ *and any set* $P \subseteq Ar$, $\mathbf{L_S}(AF) = \{L_P \cup L_{Ar \setminus P} \mid L_P \in F_{\mathbf{S}}(AF{\downarrow_P}, P^{in}, L_{Ar \setminus P}{\downarrow_{P^{in}}}, R_P), L_{Ar \setminus P} \in F_{\mathbf{S}}(AF{\downarrow_{Ar \setminus P}}, (Ar \setminus P)^{in}, L_P{\downarrow_{(Ar \setminus P)^{in}}}, R_{Ar \setminus P})\}$

The notion of decomposability refers to partitions into just two subframeworks: this is appropriate for the purposes of this paper. In particular, as far as replaceability of partial argumentation frameworks is concerned, one can treat a multiple replacement of several subframeworks with other equivalent subframeworks as a sequence of replacements each involving just one subframework. Extending the notion of decomposability to partitions of arbitrary cardinality is quite straightforward and will be dealt with in future works.

To analyze decomposability of a specific semantics $\mathbf{S}$ it is necessary to "guess" a base function $F_{\mathbf{S}}$ for $\mathbf{S}$ and then prove that it satisfies Definition 9. "Guessing" a base function for complete semantics amounts to extend the standard definition of complete labelling in order to account for "external" input arguments in the obvious way.

Definition 10 *Given an argumentation framework* $AF = (Ar, att)$*, a set of arguments* I *such that* $I \cap Ar = \emptyset$*, a labelling* $L_I \in \mathfrak{L}_I$ *and a relation* $R_I \subseteq I \times Ar$, $F_{\mathbf{CO}}(AF, I, L_I, R_I) \equiv \{Lab \in \mathfrak{L}(AF) \mid Lab(A) = \mathtt{in} \rightarrow ((\forall B \in Ar : (B,A) \in att, Lab(B) = \mathtt{out}) \wedge (\forall B \in I : (B,A) \in R_I, L_I(B) = \mathtt{out})), Lab(A) = \mathtt{out} \rightarrow ((\exists B \in Ar : (B,A) \in att \wedge Lab(B) = \mathtt{in}) \vee (\exists B \in I : (B,A) \in R_I \wedge L_I(B) = \mathtt{in})), Lab(A) = \mathtt{undec} \rightarrow (((\forall B \in Ar : (B,A) \in att, Lab(B) \neq \mathtt{in}) \wedge (\forall B \in I : (B,A) \in R_I, L_I(B) \neq \mathtt{in})) \wedge ((\exists B \in Ar : (B,A) \in att \wedge Lab(B) = \mathtt{undec}) \vee (\exists B \in I : (B,A) \in R_I \wedge L_I(B) = \mathtt{undec})))\}$

It turns out that this guess is successful, as proved by Theorem 1[2], and a similar result holds for stable semantics.

Theorem 1 *Complete semantics* **CO** *is decomposable.*

[2]Proposition 3 of [9] proves a weaker property of complete semantics for arbitrary partitions in the extension based context. It can be rephrased as follows: if all extensions of the restricted subframeworks are "locally complete" then their union is a complete extension of the whole framework. Theorem 1 provides a significant strengthening of this result though, by now, restricted to partitions into two subframeworks.

Theorem 2 *Stable semantics* **ST** *is decomposable, with the relevant base function* $F_{\mathbf{ST}}(AF, I, L_I, R_I) \equiv \{Lab \in F_{\mathbf{CO}}(AF, I, L_I, R_I) \mid \forall A \in Ar, Lab(A) \neq \texttt{undec}\}$.

Grounded semantics and preferred semantics are not decomposable[3] as shown by the counterexample of $AF = (\{A,B,C,D\}, \{(A,B),(B,C),(C,A),(A,D),(D,A)\})$ with reference to the subframeworks induced by the sets $\{A,B,C\}$ and $\{D\}$. Note in particular that $\{A,B,C\}^{\text{in}} = \{D\}$ and $\{D\}^{\text{in}} = \{A\}$. Consider the application of $F_{\mathbf{CO}}$ to the two subframeworks and, for each of them, to any of the possible labellings of its (unique) input argument. If D is labelled $\texttt{in}$, then $F_{\mathbf{CO}} : \{(A,\texttt{out}),(B,\texttt{in}),(C,\texttt{out})\}$; if D is either $\texttt{out}$ or $\texttt{undec}$, $F_{\mathbf{CO}} : \{(A,\texttt{undec}),(B,\texttt{undec}),(C,\texttt{undec})\}$. On the other hand, if A is $\texttt{in}$, $F_{\mathbf{CO}}$ prescribes that D is labelled $\texttt{out}$, while if A is labelled $\texttt{out}$ we get D labelled $\texttt{in}$, and if A is $\texttt{undec}$ we get D labelled $\texttt{undec}$. In any case, the outcome is a unique labelling and from Theorem 1 both the grounded and the preferred labellings of AF derive from the outcomes of $F_{\mathbf{CO}}$, and in particular the following two labellings are obtained: $\{(A,\texttt{undec}),(B,\texttt{undec}),(C,\texttt{undec}),(D,\texttt{undec})\}$ and $\{(A,\texttt{out}),(B,\texttt{in}),(C,\texttt{out}),(D,\texttt{in})\}$. The former is the grounded labelling, the latter is the preferred labelling: this shows that, for grounded and preferred semantics, putting together labellings of the restricted subframeworks does not necessarily give rise to a global labelling of the whole framework. Of course the question arises whether the set of labellings prescribed by grounded/preferred semantics for AF is a subset of those obtained by combining the labellings prescribed for the restricted subframeworks. We show that this property, which we call *top-down decomposability*, holds in general for these semantics. First we need to "guess" the relevant base functions.

Definition 11 *Given an argumentation framework* $AF = (Ar, att)$*, a set of arguments* I *such that* $I \cap Ar = \emptyset$*, a labelling* $L_I \in \mathfrak{L}_I$ *and a relation* $R_I \subseteq I \times Ar$*:*

- $F_{\mathbf{GR}}(AF, I, L_I, R_I) \equiv \{\mathbf{L}^*\}$*, where* $\mathbf{L}^*$ *is the minimal (w.r.t.* $\sqsubseteq$*) labelling in* $F_{\mathbf{CO}}(AF, I, L_I, R_I)$
- $F_{\mathbf{PR}}(AF, I, L_I, R_I) \equiv \{\mathbf{L} \mid \mathbf{L}$ *is a maximal (w.r.t.* $\sqsubseteq$*) labelling among those of* $F_{\mathbf{CO}}(AF, I, L_I, R_I)\}$

Proposition 1 shows a sort of monotonicity property of $F_{\mathbf{CO}}$ with respect to $\sqsubseteq$.

Proposition 1 *Given an argumentation framework* $AF = (Ar, att)$*, a set of arguments* I *such that* $I \cap Ar = \emptyset$ *and a relation* $R_I \subseteq I \times Ar$*, let* ${L_I}^1, {L_I}^2 \in \mathfrak{L}_I$ *be two labellings of* I *such that* ${L_I}^1 \sqsubseteq {L_I}^2$*. Then it holds that*

- $\forall \mathbf{L}_1 \in F_{\mathbf{CO}}(AF, I, {L_I}^1, R_I), \exists \mathbf{L}_2 \in F_{\mathbf{CO}}(AF, I, {L_I}^2, R_I)$ *such that* $\mathbf{L}_1 \sqsubseteq \mathbf{L}_2$*; and*
- $\forall \mathbf{L}_2 \in F_{\mathbf{CO}}(AF, I, {L_I}^2, R_I), \exists \mathbf{L}_1 \in F_{\mathbf{CO}}(AF, I, {L_I}^1, R_I)$ *such that* $\mathbf{L}_1 \sqsubseteq \mathbf{L}_2$.

Building on the above results, we are now in a position to prove in Theorems 3 and 4 that grounded and preferred semantics are top-down decomposable.

Theorem 3 *Given an argumentation framework* $AF = (Ar, att)$*, let* $\mathbf{L}$ *be the grounded labelling of* AF*. For any set* $P \subseteq Ar$*,* $\mathbf{L}\downarrow_P \in F_{\mathbf{GR}}(AF\downarrow_P, P^{in}, \mathbf{L}\downarrow_{P^{in}}, R_P)$*.*

[3] A counterexample to decomposability of grounded semantics is provided also in [9].

Theorem 4 *Given an argumentation framework $AF = (Ar, att)$, let $\mathbf{L}$ be a preferred labelling of AF. For any set $P \subseteq Ar$, $\mathbf{L}{\downarrow}_P \in F_{\mathbf{PR}}(AF{\downarrow}_P, P^{in}, \mathbf{L}{\downarrow}_{P^{in}}, R_P)$.*

3. I/O Argumentation Frameworks

Results on semantics decomposability provide a basis both to introduce I/O Argumentation Frameworks and to investigate the notions of equivalence and interchangeability. Let us exemplify these ideas by introducing a couple of simple examples.

First, consider a finite chain of n arguments $A_1, \ldots, A_n$ such that A_i attacks A_{i+1} with $1 \leq i < n$, and suppose that only A_1 can receive further attacks from other arguments and only A_n can attack other arguments. According to all the semantics considered in this paper (and to most semantics available in the literature), the "black-box behavior" of a sequence of arguments of this kind whose external "terminals" are A_1 and A_n only depends on whether n is odd or even. In fact, the behavior of any odd-length sequence is the same as the one of A_1 alone (with n odd, A_n gets necessarily the same label as A_1), while for any even-length sequence the behavior is the same as in the case $n = 2$ (if A_1 is `in` then A_n is `out`, if A_1 is `out` then A_n is `in`, if A_1 is `undec` then A_n is `undec`).

There are however examples where different semantics give rise to different summarizations. Consider the argumentation framework $AF = (\{A, B, C, D\}, \{(A, B), (B, A), (A, C), (B, C), (C, D)\})$ and assume that no argument in $\{A, B, C, D\}$ may receive further attacks and only D can attack arguments outside. Then, according to preferred semantics, the "black-box behavior" of this argumentation framework is equivalent to the one of a single unattacked argument D (in fact D is included in all preferred extensions), while according to grounded semantics the "black-box behavior" of this argumentation framework is equivalent to the one of a single argument D attacking itself (D is labelled `undec` in either case).

Let us now undertake the formalization of the intuitive notions introduced above. First, in order to characterize an argumentation framework as a "black box" we need to identify the arguments playing the role of its "input and output terminals". This gives rise to the definition of I/O argumentation framework.

Definition 12 *An I/O argumentation framework $\mathscr{P}$ is a tuple (AF, C, O) where $AF = (Ar, att)$ and $C, O \subseteq Ar$.*

When an I/O argumentation framework is "connected to the external world" it will "receive" some input from outside (through the conditioned arguments) and produce an output (in terms of labellings of the output arguments). Technically speaking, the relation between input and output is determined by a (semantics specific) base function. Thus equivalence between two I/O argumentation frameworks having the same "input and output terminals" (i.e. the same conditioned and output arguments) has to be referred to a given base function F: for any possible input, F has to produce the same labellings of output arguments in the two frameworks.

Definition 13 *Two I/O argumentation frameworks $\mathscr{P}_1 = (AF_1, C, O)$ and $\mathscr{P}_2 = (AF_2, C, O)$ are* equivalent under a base function F *iff for any set I, any relation $R_I \subseteq I \times C$ and any labelling $L_I \in \mathfrak{L}_I$, it holds that $\{\mathbf{L}_1{\downarrow}_O \mid \mathbf{L}_1 \in F(AF_1, I, L_I, R_I)\} = \{\mathbf{L}_2{\downarrow}_O \mid \mathbf{L}_2 \in F(AF_2, I, L_I, R_I)\}$.*

While Definition 13 applies to any base function, we will only use base functions associated with a semantics **S**: in this case, we will indicate equivalence under the base function as equivalence under **S**. It is straightforward to see that the informal examples of equivalent frameworks introduced above satisfy the requirements of Definition 13.

The fact that two I/O frameworks are equivalent according to a given semantics **S** is a necessary but not sufficient condition for them being interchangeable wrt **S** in the context of a bigger framework: interchangeability depends also on the decomposability properties of **S**. The desired property actually holds for stable and grounded semantics: equivalence of I/O frameworks entails their interchangeability, as shown in Theorem 5.

Theorem 5 *Let $\mathscr{P}_1 = (AF_1', C, O)$ and $\mathscr{P}_2 = (AF_2', C, O)$ be two equivalent I/O argumentation frameworks under grounded (stable) semantics. Let $AF_1 = (E \cup D_1, att_1)$ and $AF_2 = (E \cup D_2, att_2)$ be two argumentation frameworks coinciding in a (possibly empty) set E, i.e. $AF_1{\downarrow_E} = AF_2{\downarrow_E}$, such that $AF_1{\downarrow_{D_1}} = AF_1'$, $AF_2{\downarrow_{D_2}} = AF_2'$, $att_1 \cap (E \times D_1) = att_2 \cap (E \times D_2) \subseteq (E \times C)$, $att_1 \cap (D_1 \times E) = att_2 \cap (D_2 \times E) \subseteq (O \times E)$. Then, letting $\mathbf{L}_1$ and $\mathbf{L}_2$ be the grounded labellings of AF_1 and AF_2, respectively, it holds that $\mathbf{L}_1{\downarrow_E} = \mathbf{L}_2{\downarrow_E}$ (for stable semantics, it holds that $\{\mathbf{L}_1{\downarrow_E} \mid \mathbf{L}_1 \in \mathbf{L}_{\mathbf{ST}}(AF_1)\} = \{\mathbf{L}_2{\downarrow_E} \mid \mathbf{L}_2 \in \mathbf{L}_{\mathbf{ST}}(AF_2)\}$).*

We are also able to prove that, for preferred semantics, equivalence of I/O frameworks entails their interchangeability as far as credulous justification is concerned.

Theorem 6 *Let $\mathscr{P}_1 = (AF_1', C, O)$ and $\mathscr{P}_2 = (AF_2', C, O)$ be two equivalent I/O argumentation frameworks under preferred semantics. Let $AF_1 = (E \cup D_1, att_1)$ and $AF_2 = (E \cup D_2, att_2)$ be two argumentation frameworks coinciding in a (possibly empty) set E, i.e. $AF_1{\downarrow_E} = AF_2{\downarrow_E}$, such that $AF_1{\downarrow_{D_1}} = AF_1'$, $AF_2{\downarrow_{D_2}} = AF_2'$, $att_1 \cap (E \times D_1) = att_2 \cap (E \times D_2) \subseteq (E \times C)$, $att_1 \cap (D_1 \times E) = att_2 \cap (D_2 \times E) \subseteq (O \times E)$. Then, for any argument $A \in E$, A is credulously justified according to preferred semantics in AF_1 if and only if it is credulously justified according to preferred semantics in AF_2.*

4. Conclusions

This paper provides a contribution to the growing corpus of results on partial argumentation frameworks and their properties in several respects, among the others concerning semantics decomposability. In particular, semantics decomposability finds its roots in the SCC-recursiveness property [2] and has relationships with the division-based method [7] and the splitting of argumentation frameworks [3] to deal with changes in order to restrict computation on the affected part. Both these approaches are based on the directionality principle and consider two subframeworks such that one has an output, without having an input, and the other has an input (from the former) without having an output. Our approach provides a reference framework for evaluating incrementally argumentation frameworks by considering generic mutually interacting parts of a framework. Comparing the results in [7,3] with those presented in this paper it emerges (i) that the notions we introduce of semantics decomposability, input/output argumentation frameworks, and their equivalence have no counterpart on the cited works; and (ii) that only complete and stable semantics preserves decomposability when removing the requirements on partitioning the framework. Identifying the minimal set of requirements to recover decomposability for other semantics represents an interesting research question.

Another related work is [9], which deals with the problem of evaluating different parts of a framework with different semantics. A sort of bottom-up decomposability is proved in [9] for all semantics satisfying the properties of conflict-freeness, admissibility, and completeness, and in this paper we generalize these results. Moreover, the notion of decomposability we introduce is close to the property of uniform case extension equivalence defined in [9], where, however, no further results are provided. Extending our results to the case of arbitrary partitions with different semantics involved as in multi-sorted argumentation is an important future development.

The notion of equivalence of input/output frameworks can be compared with other notions of equivalence of argumentation frameworks, namely strong equivalence [8] and weak expansion equivalence [3]. Strong equivalence (wrt a given semantics) of two argumentation frameworks AF_1 and AF_2 requires that applying the same (arbitrary) addition of arguments and attacks to both AF_1 and AF_2 one gets the same semantics extensions. Weak expansion equivalence enfeebles the strong equivalence notion by imposing some constraints on the arguments and attacks that can be added, but keeps the requirement that semantic extensions are equal in the two resulting frameworks. Equivalence of I/O frameworks is less demanding as it concerns only the behavior of frameworks as "black boxes": extensions of equivalent I/O frameworks can be different, provided that these differences do not manifest themselves outside.

To conclude, I/O frameworks equivalence provides a formal setting for the study of replacement and/or combination of (parts of) argumentation frameworks while preserving semantics properties on the remaining part. In particular, we provide results ensuring interchangeability of equivalent frameworks under stable, grounded and credulous preferred semantics: the analysis of further semantics is left for future work.

References

[1] P. Baroni, M. Caminada, and M. Giacomin. An introduction to argumentation semantics. *The Knowledge Engineering Review*, 26:4:365–410, 2011.

[2] P. Baroni, M. Giacomin, and G. Guida. SCC-recursiveness: a general schema for argumentation semantics. *Artificial Intelligence Journal*, 168(1-2):165–210, 2005.

[3] Ringo Baumann. Splitting an argumentation framework. In *Proc. of LPNMR 2011 11th Int. Conf. on Logic Programming and Nonmonotonic Reasoning*, pages 40–53, 2011.

[4] M. W. A. Caminada. On the issue of reinstatement in argumentation. In *Proc. of the 10th European Conference on Logics in Artificial Intelligence (JELIA 2006)*, pages 111–123, 2006.

[5] C. Cayrol, F. Dupin de Saint-Cyr, and M. Lagasquie-Schiex. Change in abstract argumentation frameworks: adding an argument. *Journal of Artificial Intelligence Research*, 38:49–84, 2010.

[6] 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.

[7] B. Liao, L. Jin, and R. C. Koons. Dynamics of argumentation systems: A division-based method. *Artificial Intelligence*, 175:1790–1814, 2011.

[8] E. Oikarinen and S. Woltran. Characterizing strong equivalence for argumentation frameworks. *Artificial Intelligence*, 175:1985–2009, 2011.

[9] T. Rienstra, A. Perotti, S. Villata, D. Gabbay, L. van Der Torre, and G. Boella. Multi-sorted argumentation frameworks. In *Proc. of TAFA 2011 First Int. Workshop on the Theory and Applications of Formal Argumentation*, pages 231–245, 2011.

[10] S. Villata, G. Boella, and L. van Der Torre. Argumengtation patterns. In *Proc. of ARGMAS 2011 8th Int. Workshop on Argumentation in Multi-Agent Systems*, pages 133–150, 2011.