

Attack-defense semantics of argumentation

Beishui LIAO ^{a,1}, Leendert VAN DER TORRE ^b

^a*Zhejiang University*

^b*University of Luxembourg*

Abstract. Abstract argumentation is an important research area in AI. It is mainly about the acceptability of arguments in an argumentation framework. Classical notion of defense has not fully reflected some useful information implicitly encoded by the interaction relation between arguments. In this paper, instead of using arguments and attacks as first citizens, a novel notion of attack-defense is adopted as a first citizen, based on which a theory of attack-defense framework and attack-defense semantics are established, where an attack-defense is a triple (x, y, z) , meaning that: argument x defends argument z against attacker y . Attack-defense semantics can be used not only to identify the impact of arguments in some odd cycles, and remove some “useless” defenses, but also to capture new types of equivalence that cannot be represented by the existing notions of equivalence of argumentation frameworks. In addition, it shows that attack-defense framework and attack-defense semantics can represent some knowledge that cannot be represented in Dung’s argumentation, e.g., some context-sensitive knowledge in a dialogue.

Keywords. Dung’s abstract argumentation, equivalence, expressiveness, odd cycles

1. Introduction

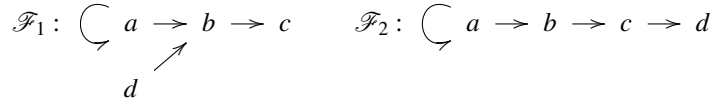
In the field of AI, abstract argumentation is mainly about the acceptability of arguments in an argumentation framework (AF) [9,12]. A set of arguments that is collectively acceptable according to some criteria is called an extension. There are two basic criteria for defining all kinds of extensions that are based on the notion of admissible set, called conflict-freeness and defense. An argument is defended by a set of arguments if every attacker of this argument is attacked by at least one argument in this set. Obviously, the notion of defense plays an important role in evaluating the status of arguments. However, this usage of the classical notion of defense has not fully reflected some useful information implicitly encoded by the interaction relation between arguments. The latter can be used to deal with some important problems in formal argumentation.

The first problem is about the expressivity of Dung’s argumentation. In many situations, the combination of arguments and attacks to form defenses is contextual and may not refer to an AF. For instance, in a dialogue, when a proponent says c and an opponent says b , the proponent uses a as a counterargument. Then, the proponent says e and the opponent says b again. This time, the proponent uses d as a counterargument, rather than

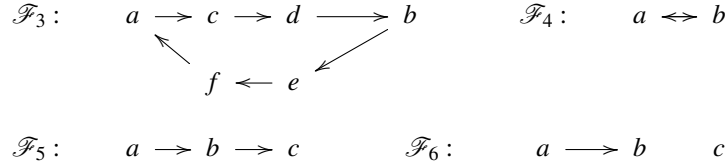
¹Corresponding Author: Beishui Liao.

a , due to some reason, say, rhetoric. It is interesting to note that such context-sensitive knowledge cannot be represented by Dung's argumentation.

The second problem is about the treatment of odd cycles. Intuitively, arguments in some odd cycles cannot provide support for the acceptance of some other arguments. In \mathcal{F}_1 , argument a is not useful for the acceptance of c , while in \mathcal{F}_2 , argument a makes all other argument not acceptable. Is it possible to explicitly represent the impact of such arguments on other arguments, and remove some "useless" impact?



The third problem is about the equivalence between AFs. Intuitively, some AFs are equivalent in terms of some interpretations, but cannot be represented by the existing notions of equivalence. Some of AFs are not equivalence, but cannot be well captured by the existing notions of equivalence. For instance, \mathcal{F}_3 and \mathcal{F}_4 are equivalent w.r.t. the acceptance of a and b , in the sense that in both AFs, a is the root reason to accept a , and b is the root reason to accept b . \mathcal{F}_5 and \mathcal{F}_6 are equivalent according to the notion of standard equivalence, but the reasons for accepting c in the two AFs are different.



To address the above problems, instead of using arguments and attacks as first citizens, we adopt a novel notion of attack-defense as a first citizen and establish a theory of attack-defense framework and attack-defense semantics, where an attack-defense is a triple (x, y, z) , meaning that: argument x defends argument z against attacker y . Since a successful attack-defense contains not only accepted arguments but also the information about why arguments are accepted, an attack-defense extension contains more information than that of the corresponding argument extension.

The structure of this paper is as follows. In Section 2, we define attack-defense framework and attack-defense semantics. In Section 3, we formulate some important properties of attack-defense semantics. In Section 4, we study attack-defense equivalence. In Section 5, attack-defense semantics in Dung's argumentation is presented. In section 6, we conclude the paper. Due to space limit, we omit all proofs in this paper. Please refer to the following link ² for details.

2. Attack-defense framework and attack-defense semantics

In this section, we introduce notions of attack-defense framework and attack-defense semantics. An attack-defense is a triple (x, y, z) , in which x , y and z are arguments. To generalize the notion of attack-defense such that every argument has at least one attack-defense, we introduce two special arguments, denoted \top and \perp (slightly abusing notations), indicating that they are always accepted and rejected, respectively. So, in (x, y, z) ,

²<https://github.com/baiseliao/COMMA2024.git/ios-book-article.pdf>

z is an argument excepting \top and \perp . Given that z is not \perp , y cannot be \top . Otherwise, z is \perp . By (\top, \perp, z) , we say that z is defended by \top against \perp . By (\perp, y, z) , we say that z is defended by \perp against y . Since \perp cannot be accepted in any case, (\perp, y, z) also cannot be successful in any case. The notion of attack-defense shares some similarity with the notion of weak-defense used in [10] in the sense that the relation of one argument (partially) defending another argument is explicitly represented and in the new semantics. The difference is that in [10], an attack-defense is a tuple in the form of $\langle x, z \rangle$, where z 's attacker is not included, while in this paper, an attack-defense is a triple (x, y, z) , where z 's attacker y is included. Note that this fundamental difference has the following significant consequences. First, weak-defense is viewed as a type of argument, and no new semantics based on this notion is defined, while attack-defense is used to define a new semantics. Second, the semantics based on the notion of attack-defense has some interesting properties that will be introduced in Section 3.

Formally, we have the following definition.

Definition 1 (Attack-defense framework). *Let U be the universe of arguments, which contains \top and \perp . Attack-defense framework is a set of attack-defenses $T \subseteq U \times U \setminus \{\top\} \times U \setminus \{\top, \perp\}$. We use z_y^x to denote that x is a defender of a defendee z against an attacker y .*

Given an attack-defense z_y^x , we use the following notations to denote the defendee, defender and attacker in the attack-defense: **defendee**(z_y^x) = z , **defender**(z_y^x) = x , and **attacker**(z_y^x) = y . Given a set of attack-defenses $D \subseteq T$, we use the following notations: **defendee**(D) = $\{z \mid z_y^x \in D\}$, **defender**(D) = $\{x \mid z_y^x \in D\}$, **attacker**(D) = $\{y \mid z_y^x \in D\}$, and **argument**(D) = (**defendee**(D) \cup **defender**(D) \cup **attacker**(D)).

Then, attack-defense semantics defines sets of attack-defenses that all are successful together.

Definition 2 (Attack-defense semantics). *Let U be the universe of arguments, $U' = U \setminus \{\top\}$, and $U'' = U \setminus \{\top, \perp\}$. attack-defense semantics is defined as a partial function $\Sigma : 2^{U \times U' \times U''} \rightarrow 2^{2^{U \times U' \times U''}}$, which associates a set of attack-defenses with a set of subsets of these attack-defenses.*

Intuitively, we say that an attack-defense (x, y, z) is successful w.r.t. a set of attack-defenses D if x is \top or a defender of D , and for each $y' \neq y$ that attacks z , there is an attack-defense (x', y', z) in D .

Definition 3 (Successful attack-defense). *Let $D \subseteq T$ be a set of attack-defenses, and $z_y^x \in T$ be an attack-defense. We say that z_y^x is successful w.r.t. D if $x = \top$ or $x \in \text{defendee}(D)$, and for each $y' \neq y$ that attacks z , $\exists z_{y'}^{x'} \in D$.*

We say that a set of attack-defenses D is admissible iff every attack-defense in D is successful w.r.t. D , and no argument that is both a defendee and an attacker in D .

Definition 4 (Admissible set of attack-defenses). *$D \subseteq T$ is admissible iff every attack-defense in D is successful w.r.t. D , and **defendee**(D) \cap **attacker**(D) = \emptyset .*

Definition 5 (Complete attack-defense extension). *D is complete iff D is admissible, and for every attack-defense in T , if it is successful w.r.t. D , then it is in D .*

April 2022

Definition 6 (Preferred attack-defense extension). *D is preferred iff D is a maximal complete attack-defense extension w.r.t. set inclusion.*

Definition 7 (Stable attack-defense extension). *D is stable iff D is admissible, and for all $z_y^x \in T \setminus D$, $y \in \text{defendee}(D)$.*

Theorem 1. *If D is a stable attack-defense extension, then it is complete and preferred, but not necessarily vice versa.*

Example 1. Let $T_1 = \{c_b^a, b_a^\perp, a_\perp^\top, e_b^d, b_d^\perp, d_\perp^\top\}$. There is only one attack-defense extension under all semantics $D = \{a_\perp^\top, d_\perp^\top, c_b^a, e_b^d\}$.

Similar to Dung's theory, attack-defense semantics can also be characterized by a function.

Definition 8 (Characteristic function). *The characteristic function, denoted by F_T , of an attack-defense framework T is defined as follows:*

$$F_T : 2^T \rightarrow 2^T, \\ F_T(D) = \{z_y^x \mid z_y^x \text{ is successful w.r.t. } D\}.$$

Example 2. Let $T_2 = \{b_a^\top, c_f^e, e_d^c\}$. We have: $F_{T_2}(\emptyset) = \{b_a^\top\}$, $F_{T_2}^2(\emptyset) = F_{T_2}(\emptyset)$. $F_{T_2}(\{c_f^e, e_d^c\}) = \{b_a^\top, c_f^e, e_d^c\}$, $F_{T_2}^2(\{c_f^e, e_d^c\}) = F_{T_2}(\{c_f^e, e_d^c\})$.

Definition 9 (Grounded attack-defense extension). *The grounded attack-defense extension of T is the least fixed point of F_T .*

It is easy to see that F_T is monotonic w.r.t. set inclusion.

Theorem 2. *Let D be a set of attack-defenses. If $\text{defendee}(D) \cap \text{attacker}(D) = \emptyset$, then D is a complete attack-defense extension iff $D = F_T(D)$. The grounded attack-defense extension is the least complete attack-defense extension of F (w.r.t. set inclusion).*

In this paper we use $\Sigma(T)$ to denote the set of attack-defense extensions of T under semantics Σ , where $\Sigma \in \{CO, PR, GR, ST\}$ denotes respectively complete, preferred, grounded, and stable attack-defense semantics.

3. Properties of attack-defense semantics

Now, let us consider some properties of the attack-defense semantics.

The first property formulated in Theorem 3 is about the closure of attack-defenses: If both z_y^x and w_v^u are in an attack-defense extension, and $z_{y'}^w$ is an attack-defense, then $z_{y'}^w$ is also in the same attack-defense extension.

Theorem 3. *Let $T \subseteq \mathcal{D}^U$ be an attack-defense framework. For all $D \in \Sigma(T)$, $x, u \in \text{argument}(T)$, if $z_y^x, w_v^u \in D$, $z_{y'}^w \in T$, then $z_{y'}^w \in D$.*

The second property formulated in Theorem 4 is about the justifiability of attack-defenses: If z_y^x is in an attack-defense extension and $x \neq \top$, then there must be some $x_{y'}^u$ in the same attack-defense extension.

April 2022

Theorem 4. For all $D \in \Sigma(T)$, if $z_y^x \in D$ and $x \neq \top$ then there exists $x_y^u \in T$ s.t. $x_y^u \in D$.

The third property is about the incompleteness of an attack-defense framework.

Theorem 5. For all $z_y^x, v_y^u \in T$, it is not necessary that $v_y^x \in T$.

This property can be illustrated by Example 1. When c_b^a and e_b^d are in T , but e_b^a and c_b^d are not in T .

The fourth property is about the unsatisfiability of some types of attack-defenses.

Definition 10 (Unsatisfiability of attack-defense). We say that z_y^x is unsatisfiable under semantics Σ iff z_y^x cannot be in any attack-defense extension under semantics Σ , where $\Sigma \in \{CO, PR, GR, ST\}$.

In this paper, as typical examples, we introduce three types of unsatisfiable attack-defenses. The first type is the attack-defenses in the form z_y^\perp , which is by definition unsatisfiable.

Theorem 6. Defense $z_y^\perp \in T$ is unsatisfiable under semantics $\Sigma \in \{CO, PR, GR, ST\}$.

The second type is the attack-defenses related to self-attacking arguments.

Theorem 7. Defenses $z_y^y, z_z^x \in T$ are unsatisfiable. Furthermore, if $z_y^y \in T$, then u_v^y is unsatisfiable under semantics $\Sigma \in \{CO, PR, GR, ST\}$.

The third type is the attack-defenses related to a 3-cycle consisting of x , y and z such that x attacks y , y attacks z , and z attacks x .

Theorem 8. If there exist $x_z^y, y_x^z, z_y^x \in T$, then x_z^y, y_x^z and z_y^x are unsatisfiable under semantics $\Sigma \in \{CO, PR, GR, ST\}$.

Given an attack-defense framework T , the sets of unsatisfiable attack-defenses of Types I, II and III are denoted $u_1(T)$, $u_2(T)$ and $u_3(T)$, respectively. Let $u(T) = u_1(T) \cup u_2(T) \cup u_3(T)$.

When an attack-defense is unsatisfiable under a semantics Σ , it might be removed from an attack-defense framework without affecting the evaluation of the status of other attack-defenses in the theory.

Under stable attack-defense semantics, the removal of an attack-defense might change the emptiness of the set of extensions of an attack-defense framework. For instance, let $T = \{z_y^\perp\}$, which has no stable attack-defense extension. However, after z_y^\perp is removed, $T' = \{\}$ has a stable attack-defense extension, which is the empty set.

Under other semantics, we have the following theorem, which cannot be applied to stable attack-defense semantics.

Definition 11 (Reduct of an attack-defense framework). For any attack-defense framework T , the reduct of T is defined as $T^- = T \setminus u(T)$.

Theorem 9. For any attack-defense framework T , and for $\Sigma \in \{CO, PR, GR\}$, $\Sigma(T) = \Sigma(T^-)$.

4. Attack-defense equivalence

We differentiate two types of attack-defense equivalence: standard equivalence and root equivalence.

Definition 12 (Standard attack-defense equivalence). *Let T_1 and T_2 be two attack-defense frameworks. T_1 and T_2 are equivalent under attack-defense semantics Σ , denoted $T_1 \equiv_d^\Sigma T_2$, iff $\Sigma(T_1) = \Sigma(T_2)$.*

Root equivalence is defined in terms of the notion of transitive closure of defenses.

In this paper, if z_y^x and v_u^z are all in D , we say that x is an indirect defender of v w.r.t. D . For simplicity, when we consider the defense relation between x and z , we write (x, z) , instead of (x, y, z) or z_y^x . The defense relation is transitive, i.e., if (x, z) and (z, u) hold, then (x, u) holds. Formally, we have the following definition.

Definition 13 (Transitive closure of defenses). *For all $D \in \Sigma(T)$, let $\bar{D} = \{(x, z) \mid z_y^x \in D\}$. The transitive closure of \bar{D} is denoted \bar{D}^+ .*

Given an attack-defense extension D , for any argument $z \in \mathbf{argument}(T)$, we say an argument $x \in \mathbf{argument}(D)$ is a root reason of accepting z , if $(x, x) \in \bar{D}^+$ and $(x, z) \in \bar{D}^+$. We say that \top is a root reason of accepting z , if $(\top, z) \in \bar{D}^+$.

Example 3. Let $T_3 = \{a_f^e, b_d^c, c_a^f, d_c^a, e_b^d, f_e^b\}$. Under preferred attack-defense semantics, $PR(T_3) = \{D_1, D_2\}$, where $D_1 = \{a_f^e, d_c^a, e_b^d\}$ and $D_2 = \{b_d^c, f_e^b, c_a^f\}$. So, we have: $\bar{D}_1 = \{(a, d), (d, e), (e, a)\}$. $\bar{D}_1^+ = \bar{D}_1 \cup \{(a, a), (d, d), (e, e), (a, e), (d, a), (e, d)\}$. $\bar{D}_2 = \{(b, f), (f, c), (c, b)\}$. $\bar{D}_2^+ = \bar{D}_2 \cup \{(b, b), (c, c), (f, f), (b, c), (c, f), (f, b)\}$. So, in D_1 , a , d and e are root reasons of accepting a , d and e , resp. The results are similar in D_2 .

Definition 14 (Root reasons for accepting arguments). *Given an attack-defense extension D , for any argument $\alpha \in \mathbf{argument}(T)$, the root reasons for accepting α , denoted $r(\alpha, D)$, is defined as follows.*

$$r(z, D) = \{x \mid (x, x) \in \bar{D}^+, (x, z) \in \bar{D}^+\} \cup \{\top \mid (\top, z) \in \bar{D}^+\} \quad (1)$$

The set of root reasons for accepting arguments in T under semantics Σ is defined as follows.

$$root_\Sigma(z, T) = \{r(z, D) \mid D \in \Sigma(T)\} \quad (2)$$

Definition 15 (Root equivalence). *Let T and T' be two attack-defense frameworks. For all $B \subseteq \mathbf{argument}(T) \cap \mathbf{argument}(T')$, we say that T and T' are equivalent w.r.t. the root reasons for accepting B under semantics Σ , denoted $T \equiv_{r,B}^\Sigma T'$, iff for all $z \in B$, $root_\Sigma(z, T) = root_\Sigma(z, T')$.*

When $B = \mathbf{argument}(T) = \mathbf{argument}(T')$, we write $T \equiv_r^\Sigma T'$ for $T \equiv_{r,B}^\Sigma T'$.

Example 4. Continue Example 3. Let $T'_3 = \{a_b^a, b_a^b\}$. $PR(T'_3) = \{D'_1, D'_2\}$, where $D'_1 = \{a_b^a\}$, $D'_2 = \{b_a^b\}$. Let $B = \{a, b\}$. Under preferred attack-defense semantics, it holds that $T_3 \equiv_{r,B}^{PR} T'_3$, because $root_{PR}(a, T_3) = root_{PR}(a, T'_3) = \{\{a\}, \{\}\}$, and $root_{PR}(b, T_3) = root_{PR}(b, T'_3) = \{\{\}, \{b\}\}$.

Theorem 10. *For any attack-defense frameworks T and T' , for any $B \subseteq \mathbf{argument}(T) \cap \mathbf{argument}(T')$, if $T \equiv_d^\Sigma T'$ then $T \equiv_{r,B}^\Sigma T'$, but not vice versa.*

5. Attack-defense semantics in Dung's argumentation

In this section, we introduce the application of attack-defense semantics to Dung's argumentation. First, let us recall some notions of Dung's argumentation.

5.1. Preliminaries

According to Dung's argumentation [9], an AF is defined as $\mathcal{F} = (\mathcal{A}, \rightarrow)$, where \mathcal{A} is a set of arguments and $\rightarrow \subseteq \mathcal{A} \times \mathcal{A}$ is a set of attacks between arguments.

Let $\mathcal{F} = (\mathcal{A}, \rightarrow)$ be an AF. Given a set $B \subseteq \mathcal{A}$ and an argument $\alpha \in \mathcal{A}$, B attacks α , denoted $B \rightarrow \alpha$, iff there exists $\beta \in B$ such that $\beta \rightarrow \alpha$. We call an argument an *initial* argument if it has no attacker.

Given $\mathcal{F} = (\mathcal{A}, \rightarrow)$ and $E \subseteq \mathcal{A}$, we say: E is *conflict-free* if $\nexists \alpha, \beta \in E$ such that $\alpha \rightarrow \beta$; $\alpha \in \mathcal{A}$ is *defended* by E if $\forall \beta \rightarrow \alpha, E \rightarrow \beta$; B is *admissible* if E is conflict-free, and each argument in E is defended by E ; E is a *complete extension* of \mathcal{F} if E is admissible, and each argument in \mathcal{A} that is defended by E is in E . E is a *preferred extension* of \mathcal{F} if E is an maximal complete extension of \mathcal{F} . E is the *grounded extension* of \mathcal{F} if E is the minimal complete extension of \mathcal{F} . We use $\sigma(\mathcal{F})$ to denote the set of σ extensions of \mathcal{F} , where $\sigma \in \{\text{co}, \text{pr}, \text{gr}, \text{st}\}$ (indicating complete, preferred, grounded, stable semantics, reps.) is a function mapping each AF to a set of σ extensions, called σ semantics.

For AFs $\mathcal{F}_1 = (\mathcal{A}_1, \rightarrow_1)$ and $\mathcal{F}_2 = (\mathcal{A}_2, \rightarrow_2)$, we use $\mathcal{F}_1 \cup \mathcal{F}_2$ to denote $(\mathcal{A}_1 \cup \mathcal{A}_2, \rightarrow_1 \cup \rightarrow_2)$. The standard equivalence and strong equivalence of AFs are defined as follows.

Definition 16 (Standard and strong equiv. of AFs). [11] *Let \mathcal{F} and \mathcal{G} be two AFs.*

- \mathcal{F} and \mathcal{G} are of *standard equivalence* w.r.t. a semantics σ , in symbols $\mathcal{F} \equiv^\sigma \mathcal{G}$, iff $\sigma(\mathcal{F}) = \sigma(\mathcal{G})$.
- \mathcal{F} and \mathcal{G} are of *strong equivalence* w.r.t. a semantics σ , in symbols $\mathcal{F} \equiv_s^\sigma \mathcal{G}$, iff for all AF \mathcal{H} , it holds that $\sigma(\mathcal{F} \cup \mathcal{H}) = \sigma(\mathcal{G} \cup \mathcal{H})$.

Example 5. Consider $\mathcal{F}_3 - \mathcal{F}_6$ in Section 1. In terms of Definition 16, under complete semantics, we have: $\mathcal{F}_3 \not\equiv^{\text{co}} \mathcal{F}_4$, $\mathcal{F}_3 \not\equiv_s^{\text{co}} \mathcal{F}_4$; $\mathcal{F}_5 \equiv^{\text{co}} \mathcal{F}_6$, $\mathcal{F}_5 \not\equiv_s^{\text{co}} \mathcal{F}_6$.

Given an AF $\mathcal{F} = (\mathcal{A}, \rightarrow)$, the kernel of \mathcal{F} under complete semantics, call *c-kernel*, is defined as follows.

Definition 17 (c-kernel of an AF). [11] *For an AF $\mathcal{F} = (\mathcal{A}, \rightarrow)$, the c-kernel of \mathcal{F} is defined as $\mathcal{F}^{\text{ck}} = (\mathcal{A}, \rightarrow^{\text{ck}})$, where*

$$\rightarrow^{\text{ck}} = \rightarrow \setminus \{\alpha \rightarrow \beta \mid \alpha \neq \beta, \alpha \rightarrow \alpha, \beta \rightarrow \beta\} \quad (3)$$

According to [11], it holds that $\text{co}(\mathcal{F}) = \text{co}(\mathcal{F}^{\text{ck}})$, and for any AFs F and G : $\mathcal{F}^{\text{ck}} = \mathcal{G}^{\text{ck}}$ iff $\mathcal{F} \equiv_s^c \mathcal{G}$.

5.2. Attack-defense framework of an AF

Given $\mathcal{F} = (\mathcal{A}, \rightarrow)$, the attack-defense framework of \mathcal{F} can be defined in the following way.

For $x, y, z \in \mathcal{A}$, if $x \rightarrow y$ and $y \rightarrow z$, then (x, y, z) is an attack-defense. For each initial argument $z \in \mathcal{A}$, there is a unique defense (\top, \perp, z) . For an argument z that is attacked by an initial argument y , there is an attack-defense (\perp, y, z) .

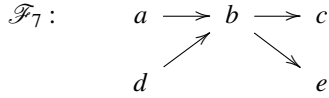
Definition 18 (Attack-defense framework of an AF). *Let $\mathcal{F} = (\mathcal{A}, \rightarrow)$ be an AF. The attack-defense framework of \mathcal{F} , denoted $d(\mathcal{F})$, is represented as follow.*

$$d(\mathcal{F}) = \{(z_y^x \mid (x, y, z \in \mathcal{A}) \wedge (x \rightarrow y) \wedge (y \rightarrow z)) \cup \{z_\perp^\top \mid (z \in \mathcal{A}) \wedge (z^- = \emptyset)\} \\ \cup \{z_y^\perp \mid (y, z \in \mathcal{A}) \wedge (y \rightarrow z) \wedge (y^- = \emptyset)\}$$

According to the definition of the attack-defense framework of an AF, we have the following theorem.

Theorem 11. *Let $d(\mathcal{F})$ be the attack-defense framework of an AF $\mathcal{F} = (\mathcal{A}, \rightarrow)$. For all $z_y^x, v_y^u \in d(\mathcal{F})$, it holds that $v_y^x \in d(\mathcal{F})$.*

Example 6. *Continue Example 1. $d(\mathcal{F}_7) = T_1 \cup \{e_b^a, c_b^d\} = \{c_b^a, b_a^\perp, a_\perp^\top, e_b^d, b_d^\perp, d_\perp^\top, e_b^a, c_b^d\}$.*



This example shows that the attack-defense framework in Example 1 cannot be represented as an attack-defense framework of an AF, in that e_b^a and c_b^d do not exist in T_1 , but have to be included in $d(\mathcal{F}_7)$. The underlying reason is that in practical dialogues, the combination of arguments and attacks to form defense are contextual and may not refer to an AF. However, in some situations, such kind of knowledge exists, as described in the first section.

5.3. Correspondence to Dung's semantics

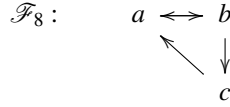
First, under a given attack-defense semantics, for each attack-defense extension of the attack-defense framework of an AF, the set of defendees of the attack-defense extension is an argument extension under a corresponding Dung's semantics.

Theorem 12. *For all $D \in \Sigma(d(\mathcal{F}))$, $\text{defendee}(D) \in \sigma(\mathcal{F})$, where $\Sigma \in \{\text{CO}, \text{PR}, \text{GR}, \text{ST}\}$ and $\sigma \in \{\text{co}, \text{pr}, \text{gr}, \text{st}\}$.*

Second, under a given Dung's semantics, for each extension E of an AF, there is a set of attack-defenses constructed from E such that this set is an attack-defense extension of the attack-defense framework generated from the AF, under a corresponding attack-defense semantics.

Theorem 13. *For all $E \in \sigma(\mathcal{F})$, let $\text{def}(E) = \{z_y^x \mid z_y^x \in d(\mathcal{F}) : x, z \in E\} \cup \{z_\perp^\top \mid z_\perp^\top \in d(\mathcal{F}) : z \in E\}$. Then, $\text{def}(E) \in \Sigma(d(\mathcal{F}))$.*

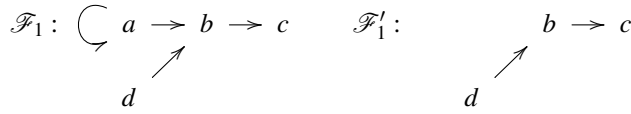
Example 7. Consider \mathcal{F}_8 below. We have: $\text{co}(\mathcal{F}_8) = \{E_1, E_2\}$, where $E_1 = \{\}$, $E_2 = \{b\}$; $\text{def}(E_1) = \{\}$, $\text{def}(E_2) = \{b_a^b\}$; $\text{CO}(\text{d}(\mathcal{F}_8)) = \{D_1, D_2\}$, where $D_1 = \{\}$, $D_2 = \{b_a^b\}$. So, it holds that $\text{def}(E_1) \in \text{CO}(\text{d}(\mathcal{F}_8))$, $\text{def}(E_2) \in \text{CO}(\text{d}(\mathcal{F}_8))$.



5.4. Reduct of the attack-defense framework of an AF

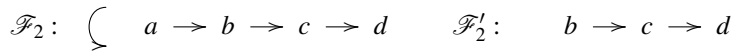
Given the attack-defense framework of an AF, some of the unsatisfiable attack-defenses can be removed, resulting a reduct of the attack-defense framework.

Example 8. Consider \mathcal{F}_1 again. In \mathcal{F}_1 , for argument c , there are two defenses c_b^d and c_b^a . Intuitively, the defense c_b^a does not contribute to the acceptance of c , because the self-attacking argument a cannot be accepted in any situation and therefore cannot provide support to the acceptance of some other arguments. In other words, when defenses are used to evaluate the status of arguments, the status of c can be determined only according to defense c_b^d , without considering unsatisfiable defenses related to the self-attacking arguments. According to Theorem 7, $\text{d}(\mathcal{F}_1) = \{a_a^a, b_a^a, c_b^a, c_b^d, d_\perp^\perp\}$. The reduct of $\text{d}(\mathcal{F}_1)$ is $\text{d}(\mathcal{F}_1)^- = \{c_b^d, d_\perp^\perp\}$. Under any semantics $\Sigma \in \{CO, PR, GR\}$, $\Sigma(\text{d}(\mathcal{F}_1)) = \Sigma(\text{d}(\mathcal{F}_1)^-) = \{D\}$, where $D = \{c_b^d, d_\perp^\perp\}$.

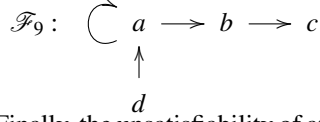


It is interesting to note that while it is natural to remove unsatisfiable attack-defenses without affecting the evaluation of some other attack-defenses, odd cycles cannot be removed from an AF without affecting the evaluation of some other arguments, except same special cases. For instance, in \mathcal{F}_1 , the self-attacking argument a can be removed without affecting the evaluation of other arguments in \mathcal{F}_1 . In other words, under Dung's semantics $\sigma \in \{co, pr, gr\}$, $\sigma(\mathcal{F}_1) = \sigma(\mathcal{F}_1')$. However, such removal of self-attacking arguments cannot be applied to all situations in general.

Example 9. Consider \mathcal{F}_2 again. Under Dung's admissibility semantics, no argument is accepted. If we remove the self-attacking argument a , resulting \mathcal{F}_2' , then arguments b and d will be accepted. This means that the removal of a self-attacking argument affects the status of other arguments. However, according to Theorem 7, in the set of attack-defenses $\text{d}(\mathcal{F}_2) = \{a_a^a, b_a^a, c_b^a, d_c^b\}$, a_a^a , b_a^a and c_b^a are unsatisfiable under any semantics. After removing all these unsatisfiable defense, we obtain a reduct of $\text{d}(\mathcal{F}_2)$, i.e., $\text{d}(\mathcal{F}_2)^- = \{d_c^b\}$. Under all semantics, both $\text{d}(\mathcal{F}_2)$ and $\text{d}(\mathcal{F}_2)^-$ have one unique extension, which is the empty set.

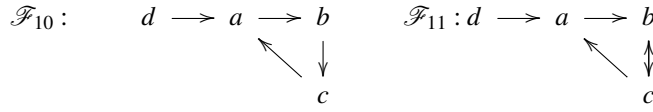


Another point worth to be noted is that the unsatisfiability of an attack-defense does not mean that its defendee is not acceptable. See the following example. In this case, b_a^a is unacceptable, but the argument b is acceptable.



Finally, the unsatisfiability of attack-defenses of an AF is not affected by the addition of some other arguments or attacks to the AF. See the following example.

Example 10. $d(\mathcal{F}_{10}) = \{d_{\perp}^{\top}, a_d^{\perp}, a_c^b, b_a^d, b_a^c, c_b^a\}$, in which a_d^{\perp} , a_c^b , b_a^c and c_b^a are unacceptable. So, $d(\mathcal{F}_{10})^- = \{d_{\perp}^{\top}, b_a^d\}$. After adding an attack from c to b , the attack-defense framework of the resulting AF is $d(\mathcal{F}_{11}) = d(\mathcal{F}_{10}) \cup \{b_c^b, c_b^c\}$, and $d(\mathcal{F}_{11})^- = d(\mathcal{F}_{10})^- \cup \{b_c^b, c_b^c\}$.



5.5. Equivalence of AFs under attack-defense semantics

The standard equivalence and strong equivalence of AFs cannot represent the equivalence of some AFs in terms of some interesting interpretations as illustrated in Section 1. The attack-defense equivalence provides more information. In this section, we introduce some relations between attack-defense equivalence and the existing two types of equivalence of AFs.

Theorem 14. Let \mathcal{F} and \mathcal{G} be two AFs. If $d(\mathcal{F}) \equiv_d^{\Sigma} d(\mathcal{G})$, then $\mathcal{F} \equiv^{\sigma} \mathcal{G}$, where $\Sigma \in \{CO, PR, GR, ST\}$, $\sigma \in \{co, pr, gr, st\}$.

Note that in many cases $\mathcal{F} \equiv^{\sigma} \mathcal{G}$, but $d(\mathcal{F}) \not\equiv_d^{\Sigma} d(\mathcal{G})$. Consider the following example.

Example 11. Since $co(\mathcal{F}_5) = co(\mathcal{F}_6) = \{\{a, c\}\}$, it holds that $\mathcal{F}_5 \equiv^{co} \mathcal{F}_6$. Since $CO(d(\mathcal{F}_5)) = \{a_{\perp}^{\top}, c_b^a\}$ and $CO(d(\mathcal{F}_6)) = \{a_{\perp}^{\top}, c_{\perp}^{\top}\}$, $CO(d(\mathcal{F}_5)) \neq CO(d(\mathcal{F}_6))$. So, it is not the case that $d(\mathcal{F}_5) \equiv_d^{CO} d(\mathcal{F}_6)$.

About the relation between attack-defense equivalence and strong equivalence of AFs, due to space limit, we only consider Dung's complete semantics. For further information about strong equivalence and kernels under Dung's semantics, the reader is referred to [11]. Formally, we have the following lemma and theorem.

Lemma 1. It holds that $CO(d(\mathcal{F})) = CO(d(\mathcal{F}^{ck}))$.

Theorem 15. Let \mathcal{F} and \mathcal{G} be two AFs. If $\mathcal{F} \equiv_s^{co} \mathcal{G}$, then $d(\mathcal{F}) \equiv_d^{CO} d(\mathcal{G})$.

Note that in many cases $d(\mathcal{F}) \equiv_d^{CO} d(\mathcal{G})$, but $\mathcal{F} \not\equiv_s^{co} \mathcal{G}$. Consider the following example.

Example 12. Since $CO(d(\mathcal{F}_{12})) = CO(d(\mathcal{F}_{13})) = \{\{a_{\perp}^{\top}, c_b^a\}\}$, $d(\mathcal{F}_{12}) \equiv_d^{CO} d(\mathcal{F}_{13})$. However, since $\mathcal{F}_{12}^{ck} \neq \mathcal{F}_{13}^{ck}$, $\mathcal{F}_{12} \not\equiv_s^{co} \mathcal{F}_{13}$.



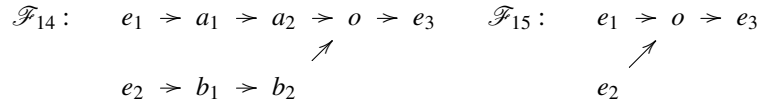
The relation between root equivalence and Dung's standard equivalence is formulated by the following property.

Theorem 16. *Let $\mathcal{F} = (\mathcal{A}_1, \rightarrow_1)$ and $\mathcal{H} = (\mathcal{A}_2, \rightarrow_2)$ be two AFs. If $d(\mathcal{F}) \equiv_r^\Sigma d(\mathcal{H})$, then $\mathcal{F} \equiv^\sigma \mathcal{H}$, where $\Sigma \in \{\text{CO}, \text{PR}, \text{GR}, \text{ST}\}$ and $\sigma \in \{\text{co}, \text{pr}, \text{gr}, \text{st}\}$.*

Note that in many cases $\mathcal{F} \equiv^\sigma \mathcal{H}$, but $d(\mathcal{F}) \not\equiv_r^\Sigma d(\mathcal{H})$. This can be easily verified by considering \mathcal{F}_5 and \mathcal{F}_6 in Example 11.

The notion of root equivalence of AFs can be used to capture a kind of summarization in the graphs. Consider the following example borrowed from [3].

Example 13. *Let $\mathcal{F}_{14} = (\mathcal{A}, \rightarrow)$ and $\mathcal{F}_{15} = (\mathcal{A}', \rightarrow')$, illustrated below. Under complete semantics, \mathcal{F}_{15} is a summarization of \mathcal{F}_{14} in the sense that $\mathcal{A}' \subseteq \mathcal{A}$, and the root reason of each argument in \mathcal{F}_{15} is the same as that of each corresponding argument in \mathcal{F}_{14} . More specifically, it holds that $\text{root}_{\text{CO}}(e_3, d(\mathcal{F}_{14})) = \text{root}_{\text{CO}}(e_3, d(\mathcal{F}_{15})) = \{\{e_1, e_2\}\}$, $\text{root}_{\text{CO}}(e_2, d(\mathcal{F}_{14})) = \text{root}_{\text{CO}}(e_2, d(\mathcal{F}_{15})) = \{\{\top\}\}$, and $\text{root}_{\text{CO}}(e_1, d(\mathcal{F}_{14})) = \text{root}_{\text{CO}}(e_1, d(\mathcal{F}_{15})) = \{\{\top\}\}$.*



Formally, we have the following definition.

Definition 19 (Summarization of AFs). *Let $\mathcal{F} = (\mathcal{A}_1, \rightarrow_1)$ and $\mathcal{H} = (\mathcal{A}_2, \rightarrow_2)$ be two AFs. \mathcal{F} is a summarization of \mathcal{H} under a semantics Σ iff $\mathcal{A}_1 \subset \mathcal{A}_2$, and $\mathcal{F} \equiv_{r, \mathcal{A}_1}^\Sigma \mathcal{H}$.*

6. Conclusions

The main contributions of this paper are three-fold. First, we have introduced novel notions of *attack-defense* and *successful attack-defenses*, and used them to formulate attack-defense framework and attack-defense semantics. It turns out that attack-defense framework is more expressive than Dung's argumentation framework, in the sense that Dung's semantics can be represented in attack-defense semantics, but not vice versa. Second, we have studied three types of unsatisfiable attack-defenses. The attack-defenses related odd cycles can be removed without affecting the evaluation of other attack-defenses, while odd cycles cannot be removed from an AF without affecting the evaluation of some other arguments in general. Third, we have formulated two new kinds of equivalence relation between AFs, i.e., attack-defense equivalence and root equivalence, and shown that attack-defense semantics can be used to capture the equivalence of AFs from the perspective of reasons for accepting arguments. In addition, we have defined a notion of summarization of AFs by exploiting root equivalence.

The idea of exploiting more usages of attacks has been used in some previous approaches, e.g., [13], [1], and [2]. However, defining a semantics in terms of the notion of attack-defense and studying the properties of this new semantics are novel.

Meanwhile, the notions of attack, defense and acceptability have very closed relations. In this paper, the notion of attack-defense is defined according to Dung's notion of admissibility. In the future work, it is worth to consider other notions of admissibility, e.g., strong admissibility [6], and weak admissibility [4].

Last but not least, attack-defense semantics and reason equivalence can be applied to dialogues and explainable AI. It would be interesting to investigate new explanation methods by combining attack-defense semantics and some existing approaches, e.g., [8], [5], and [7].

References

- [1] Ryuta Arisaka, Jeremie Dauphin, Ken Satoh, and Leendert van der Torre. Multi-agent argumentation and dialogue. *FLAP*, 9(4):921–954, 2022.
- [2] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. AFRA: argumentation framework with recursive attacks. *Int. J. Approx. Reason.*, 52(1):19–37, 2011.
- [3] Pietro Baroni, Guido Boella, Federico Cerutti, Massimiliano Giacomin, Leendert van der Torre, and Serena Villata. On the input/output behavior of argumentation frameworks. *Artificial Intelligence*, 217:144–197, 2014.
- [4] Ringo Baumann, Gerhard Brewka, and Markus Ulbricht. Shedding new light on the foundations of abstract argumentation: Modularization and weak admissibility. *Artif. Intell.*, 310:103742, 2022.
- [5] AnneMarie Borg and Floris Bex. Contrastive explanations for argumentation-based conclusions. In Piotr Faliszewski, Viviana Mascardi, Catherine Pelachaud, and Matthew E. Taylor, editors, 21st International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2022, Auckland, New Zealand, May 9-13, 2022, pages 1551–1553. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS), 2022.
- [6] Martin Caminada and Sri Harikrishnan. Strong admissibility, a tractable algorithmic approach. In Sarah Alice Gaggl, Jean-Guy Mailly, Matthias Thimm, and Johannes Peter Wallner, editors, Proceedings of the Fourth International Workshop on Systems and Algorithms for Formal Argumentation co-located with the 9th International Conference on Computational Models of Argument (COMMA 2022), Cardiff, Wales, United Kingdom, September 13, 2022, volume 3236 of CEUR Workshop Proceedings, pages 33–44. CEUR-WS.org, 2022.
- [7] Kristijonas Cyras, Antonio Rago, Emanuele Albini, Pietro Baroni, and Francesca Toni. Argumentative XAI: A survey. In Zhi-Hua Zhou, editor, Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021, pages 4392–4399. ijcai.org, 2021.
- [8] Sylvie Doutre, The o Duchatelle, and Marie-Christine Lagasque-Schiex. Visual explanations for defence in abstract argumentation. In Noa Agmon, Bo An, Alessandro Ricci, and William Yeoh, editors, Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2023, London, United Kingdom, 29 May 2023 - 2 June 2023, pages 2346–2348. ACM, 2023.
- [9] [Dung, 1995] Phan Minh Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.
- [10] Beishui Liao and Leendert van der Torre. Explanation semantics for abstract argumentation. In Henry Prakken, Stefano Bistarelli, Francesco Santini, and Carlo Taticchi, editors, Computational Models of Argument - Proceedings of COMMA 2020, Perugia, Italy, September 4-11, 2020, volume 326 of Frontiers in Artificial Intelligence and Applications, pages 271–282. IOS Press, 2020.
- [11] Emilia Oikarinen and Stefan Woltran. Characterizing strong equivalence for argumentation frameworks. In Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010, 2010.
- [12] Massimiliano Giacomin Pietro Baroni, Dov Gabbay and Leendert van der Torre, editors. Handbook of Formal Argumentation. College Publications, 2018.
- [13] Serena Villata, Guido Boella, and Leendert W. N. van der Torre. Attack semantics for abstract argumentation. In Toby Walsh, editor, IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011, pages 406–413. IJCAI/AAAI, 2011.

A. Proofs

Theorem 1. *If D is a stable attack-defense extension, then it is complete and preferred, but not necessarily vice versa.*

Proof. Since for all $z_y^x \in T \setminus D$, $y \in \mathbf{defendee}(D)$, $D \cup \{z_y^x\}$ is not admissible. So, D is complete. Assume that D is not maximal. Let $D' \subset D$ be a preferred attack-defense extension. For each $z_y^x \in D' \setminus D$, $D \cup \{z_y^x\}$ is not admissible. So, D' is not admissible. Contradiction. For the counter example, consider $T = \{a_a^a\}$. The emptyset is both a preferred attack-defense extension and a complete attack-defense extension, but not a stable attack-defense extension. \square

Theorem 2. *Let D be a set of attack-defenses. If $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$, then D is a complete attack-defense extension iff $D = F_T(D)$. The grounded attack-defense extension is the least complete attack-defense extension of F (w.r.t. set inclusion).*

Proof. Since $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$ and $D = F_T(D)$, D is admissible. Since no attack-defense in $T \setminus D$ is successful w.r.t. D , D is complete. Since F_T is monotonic, there is a unique least fixed point that is the least complete attack-defense extension of F . \square

Theorem 3. *Let $T \subseteq \mathcal{D}^U$ be an attack-defense framework. For all $D \in \Sigma(T)$, $x, u \in \mathbf{argument}(T)$, if $z_y^x, w_v^u \in D$, $z_{y'}^w \in T$, then $z_{y'}^w \in D$.*

Proof. We need to verify this property under complete attack-defense semantics. Since $w_v^u \in D$, $w \in \mathbf{defendee}(D)$. Since $z_y^x \in D$, for each $y' \neq y$ that attacks z , $\exists z_{y'}^x \in D$. Thus, $z_{y'}^w$ is successful w.r.t. D . Since D is a complete attack-defense extension, $z_{y'}^w \in D$. \square

Theorem 4. *For all $D \in \Sigma(T)$, if $z_y^x \in D$ and $x \neq \top$ then there exists $x_{y'}^u \in T$ s.t. $x_{y'}^u \in D$.*

Proof. According to Definition 3, since $z_y^x \in D$, it holds that defender $x \in \mathbf{defendee}(D)$. So, there exist $u, y' \in \mathbf{argument}(T)$ such that $x_{y'}^u \in D$. \square

The third property is about the incompleteness of an attack-defense framework.

Theorem 5. *For all $z_y^x, v_y^u \in T$, it is not necessary that $v_y^x \in T$.*

Proof. By definition, this property obviously holds. \square

Theorem 7. *Defenses $z_y^y, z_z^x \in T$ are unsatisfiable. Furthermore, if $z_y^y \in T$, then u_v^y is unsatisfiable under semantics $\Sigma \in \{CO, PR, GR, ST\}$.*

Proof. First, z_y^y means that y self-attacks and it attacks z . If z_y^y is in an admissible set of attack-defenses, then y is a defendee. According to Definition 4, $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) \neq \emptyset$. Contradiction. Second, obviously, z_z^x is not satisfiable. Third, assume that u_v^y is in some attack-defense admissible set D . Then, according to Theorem 4, there exists $y_w^x \in T$ such that $y_w^x \in D$. If $w = y$, then this contradicts $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$. Otherwise, according to the definition of complete attack-defense extension, z_y^y is also in D . This also contradicts $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$. \square

Theorem 8. *If there exist $x_z^y, y_x^z, z_y^x \in T$, then x_z^y, y_x^z and z_y^x are unsatisfiable under semantics $\Sigma \in \{CO, PR, GR, ST\}$.*

Proof. Assume that x_y^y is in some admissible set $D \subseteq T$. Then, y_x^x is also in D . As a result, $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) \supseteq \{x\}$, contradicting $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$. \square

Theorem 9. For any attack-defense framework T , and for $\Sigma \in \{CO, PR, GR\}$, $\Sigma(T) = \Sigma(T^-)$.

Proof. Under complete attack-defense semantics, let $D \in CO(T)$. Since for all $z_y^x \in u(T)$, z_y^x is unsatisfiable, $z_y^x \notin D$. According to Definitions 5 and 4, D is a complete attack-defense extension of T^- . On the other hand, let $D \in CO(T^-)$. Since any attack-defense in $u(T)$ is unsatisfiable, adding it to T^- does not affect the evaluation of other attack-defenses. Therefore, D is a complete attack-defense extension of T . Similarly, under preferred and grounded attack-defense semantics, this property also holds. \square

Theorem 10. For any attack-defense frameworks T and T' , for any $B \subseteq \mathbf{argument}(T) \cap \mathbf{argument}(T')$, if $T \equiv_d^\Sigma T'$ then $T \equiv_{r,B}^\Sigma T'$, but not vice versa.

Proof. Since $T \equiv_d^\Sigma T'$, $\Sigma(T) = \Sigma(T')$. For all $D \in \Sigma(T)$, for all $z \in B$, $\text{root}_\Sigma(z, T) = \text{root}_\Sigma(z, T')$. As a result, $T \equiv_{r,B}^\Sigma T'$.

For the converse direction, consider T_3 and T'_3 in Examples 3 and 4. Under preferred attack-defense semantics, $PR(T_3) \neq PR(T'_3)$, i.e., it is not the case that $T_3 \equiv_d^{PR} T'_3$. Meanwhile, let $B = \{a, b\}$. As described in Example 4, we have $T_3 \equiv_{r,B}^{PR} T'_3$. \square

Theorem 11. Let $d(\mathcal{F})$ be the attack-defense framework of an AF $\mathcal{F} = (\mathcal{A}, \rightarrow)$. For all $z_y^x, v_y^u \in d(\mathcal{F})$, it holds that $v_y^x \in d(\mathcal{F})$.

Proof. Since $z_y^x, v_y^u \in d(\mathcal{F})$, it holds that $x \rightarrow y$, $y \rightarrow z$, $u \rightarrow y$ and $y \rightarrow v$. Given that $x \rightarrow y$ and $y \rightarrow v$, we have $v_y^x \in d(\mathcal{F})$. \square

Theorem 12. For all $D \in \Sigma(d(\mathcal{F}))$, $\mathbf{defendee}(D) \in \sigma(\mathcal{F})$, where $\Sigma \in \{CO, PR, GR, ST\}$ and $\sigma \in \{co, pr, gr, st\}$.

Proof. Let D be a complete attack-defense extension of $d(\mathcal{F})$. According to Definition 4, $\mathbf{defendee}(D) \cap \mathbf{attacker}(D) = \emptyset$. Hence, $\mathbf{defendee}(D)$ is conflict-free. According to Definition 3, for all $z \in \mathbf{defendee}(D)$, z is an initial argument, or every attacker of z is attacked by an argument in $\mathbf{defendee}(D)$. So, $\mathbf{defendee}(D)$ is an admissible set of arguments. According to Definition 5, each successful attack-defense w.r.t. D is in D . So, each argument that is defended by $\mathbf{defendee}(D)$ is in $\mathbf{defendee}(D)$. Therefore, $\mathbf{defendee}(D)$ is a complete extension of \mathcal{F} . When D is a preferred attack-defense extension, it is easy to see that $\mathbf{defendee}(D)$ is a maximal complete extension (w.r.t. set inclusion) and therefore is a preferred extension of \mathcal{F} . Similarly, this property holds under grounded attack-defense semantics. Finally, when D is a stable attack-defense extension, for all $z_y^x \in d(\mathcal{F}) \setminus D$, $y \in \mathbf{defendee}(D)$. This means that for each argument $z \in \mathcal{A} \setminus \mathbf{defendee}(D)$, z is attacked by an argument y in $\mathbf{defendee}(D)$. Hence, $\mathbf{defendee}(D)$ is a stable extension of \mathcal{F} . \square

Theorem 13. For all $E \in \sigma(\mathcal{F})$, let $\text{def}(E) = \{z_y^x \mid z_y^x \in d(\mathcal{F}) : x, z \in E\} \cup \{z_\perp^\perp \mid z_\perp^\perp \in d(\mathcal{F}) : z \in E\}$. Then, $\text{def}(E) \in \Sigma(d(\mathcal{F}))$.

Proof. Let E be a complete extension of \mathcal{F} . For each $z_y^x \in \text{def}(E)$, since $x, z \in E$ and E is admissible, z_y^x is successful w.r.t. $\text{def}(E)$. $\mathbf{defendee}(\text{def}(E)) \cap \mathbf{attacker}(\text{def}(E)) = \emptyset$.

Otherwise, if there is an attacker in $\text{def}(E)$ that is a defendee, then $E = \mathbf{defendee}(\text{def}(E))$ is not conflict-free, contradicting the fact that E is conflict-free. As a result, $\text{def}(E)$ is admissible. Assume that $\text{def}(E)$ is not complete. Then, there exists $z_y^x \in d(\mathcal{F})$, such that z_y^x is successful w.r.t. $\text{def}(E)$, and $z_y^x \notin \text{def}(E)$. Since $z_y^x \notin \text{def}(E)$, $x \notin E$ or $z \notin E$. If $x \notin E$, then $x \notin \mathbf{defendee}(\text{def}(E))$ and therefore z_y^x is not successful w.r.t. $\text{def}(E)$. Contradiction. Alternatively, if $z \notin E$, then since E is a complete extension of \mathcal{F} , z is not defended by E . Then, there exists $y' \neq y$, such that y' attacks z and y' is not attacked by any argument in E . It turns out that z_y^x is not successful w.r.t. $\text{def}(E)$. Contradiction. Hence, $\text{def}(E)$ is a complete attack-defense extension of $d(\mathcal{F})$.

When E is a preferred extension of \mathcal{F} , E is maximal and therefore $\text{def}(E)$ is maximal (w.r.t. set inclusion). Hence, $\text{def}(E)$ is a preferred attack-defense extension of $d(\mathcal{F})$.

Similarly, this property holds under grounded semantics.

Finally, when E is a stable extension of \mathcal{F} , for all $z \in \mathcal{A} \setminus E$, z is attacked by an argument in E . This means that for each defense $z_y^x \in d(\mathcal{F}) \setminus \text{def}(E)$, $y \in \mathbf{defendee}(\text{def}(E))$. Therefore, $\text{def}(E)$ is a stable attack-defense extension of $d(\mathcal{F})$. \square

Theorem 14. Let \mathcal{F} and \mathcal{G} be two AFs. If $d(\mathcal{F}) \equiv_d^\Sigma d(\mathcal{G})$, then $\mathcal{F} \equiv^\sigma \mathcal{G}$, where $\Sigma \in \{CO, PR, GR, ST\}$, $\sigma \in \{co, pr, gr, st\}$.

Proof. If $d(\mathcal{F}) \equiv_d^\Sigma d(\mathcal{G})$, then $\Sigma(d(\mathcal{F})) = \Sigma(d(\mathcal{G}))$. According to Theorem 12, $\sigma(\mathcal{F}) = \mathbf{defendee}(\Sigma(d(\mathcal{F}))) = \mathbf{defendee}(\Sigma(d(\mathcal{G}))) = \sigma(\mathcal{G})$. Since $\sigma(\mathcal{F}) = \sigma(\mathcal{G})$, $\mathcal{F} \equiv^\sigma \mathcal{G}$. \square

Lemma 2. It holds that $CO(d(\mathcal{F})) = CO(d(\mathcal{F}^{\text{ck}}))$.

Proof. Since for every attack-defense that is related to a self-attacking argument is unsatisfiable, it is clear that $CO(d(\mathcal{F})) = CO(d(\mathcal{F}^{\text{ck}}))$. \square

Theorem 15. Let \mathcal{F} and \mathcal{G} be two AFs. If $\mathcal{F} \equiv_s^{\text{co}} \mathcal{G}$, then $d(\mathcal{F}) \equiv_d^{\text{CO}} d(\mathcal{G})$.

Proof. Obvious. \square

Theorem 16. Let $\mathcal{F} = (\mathcal{A}_1, \rightarrow_1)$ and $\mathcal{H} = (\mathcal{A}_2, \rightarrow_2)$ be two AFs. If $d(\mathcal{F}) \equiv_r^\Sigma d(\mathcal{H})$, then $\mathcal{F} \equiv^\sigma \mathcal{H}$, where $\Sigma \in \{CO, PR, GR, ST\}$ and $\sigma \in \{co, pr, gr, st\}$.

Proof. According to Definition 14, under complete semantics, the number of extensions of $\text{co}(\mathcal{F})$ is equal to the number of $\text{root}_{CO}(z, d(\mathcal{F}))$, where $z \in \mathcal{A}_1$. Since $\text{root}_{CO}(z, d(\mathcal{F})) = \text{root}_{CO}(z, d(\mathcal{H}))$, $\mathcal{A}_1 = \mathcal{A}_2$.

Let $\text{root}_{CO}(z, d(\mathcal{F})) = \text{root}_{CO}(z, d(\mathcal{H})) = \{R_1, \dots, R_n\}$. Let $\text{co}(\mathcal{F}) = \{E_1, \dots, E_n\}$ be the set of extensions of \mathcal{F} , where $n \geq 1$.

For all $\alpha \in \mathcal{A}_1$, for all R_i , $i = 1, \dots, n$, we have $\alpha \in E_i$ iff $R_i \neq \{\}$, in that in terms of Definition 14, when $R_i \neq \{\}$, there is a reason to accept α .

On the other hand, let $\text{co}(\mathcal{H}) = \{S_1, \dots, S_n\}$ be the set of extensions of \mathcal{H} . For all $\alpha \in \mathcal{A}_2 = \mathcal{A}_1$, for all R_i , $i = 1, \dots, n$, for the same reason, we have $\alpha \in S_i$ iff $R_i \neq \{\}$. So, it holds that $E_i = S_i$ for $i = 1, \dots, n$, and hence $\text{co}(\mathcal{F}) = \text{co}(\mathcal{H})$, i.e., $\mathcal{F} \equiv^{\text{co}} \mathcal{H}$.

Similarly, this property holds under preferred semantics and grounded semantics, respectively.

Finally, under stable semantics, if $ST(d(\mathcal{F})) = ST(d(\mathcal{H})) = \emptyset$, then $st(\mathcal{F}) = st(\mathcal{H}) = \emptyset$. Hence, $\mathcal{F} \equiv^{st} \mathcal{H}$. Otherwise, $st(\mathcal{F}) = st(\mathcal{H}) \neq \emptyset$. In this case, as verified under complete semantics, it holds that $\mathcal{F} \equiv^{st} \mathcal{H}$. \square