Characterising Serialisation Equivalence for Abstract Argumentation

Lars Bengel, Julian Sander and Matthias Thimm

Artificial Intelligence Group, University of Hagen, Germany

Abstract. We introduce the notion of *serialisation equivalence*, which provides a notion of equivalence that takes the underlying dialectical structure of extensions in an argumentation framework into account. Under this notion, two argumentation frameworks are considered equivalent if they possess not only the same extensions wrt. some semantics but also the same serialisation sequences. A serialisation sequence is a decomposition of an extension into a series of minimal acceptable sets and essentially offers insight into the order in which arguments need to brought forward to resolve the conflicts and to justify a particular position in the argumentation framework. We analyse serialisation equivalence in detail and show that it is generally more strict than standard equivalence and less strict than strong equivalence. Furthermore, we provide a full analysis of the computational complexity of deciding serialisation equivalence.

1 Introduction

The *(abstract) argumentation frameworks* as introduced by Dung [11] have emerged as an important formalism for knowledge representation and reasoning. Due to their argumentative nature, they are well suited to provide human-understandable explanations [1, 15]. Many recent works on the topic of explainable AI are concerned with argumentation-based approaches, see e. g. [10, 20] for an overview. An argumentation framework (AF) consists of a set of arguments and an attack relation between these arguments and we can simply represent it as a directed graph. Reasoning in abstract argumentation is done via argumentation semantics [2], which determine sets of arguments (called extensions) that are considered jointly acceptable wrt. different criteria. For instance, the basic property of *admissibility* requires a set to be conflict-free and to defend all of its arguments. Maximal admissible sets are then called *preferred* extensions. As a form of defeasible reasoning, formal argumentation is inherently linked to dialectics [13, 9]. A major aspect of dialectics is its procedural nature, where arguments are followed by counterarguments (and so on) [13, 18]. Although an AF models this well syntactically, the procedural aspect is somewhat lost in the semantical extensions [21]. In order to adhere to the dialectics that are modelled in formal argumentation, a notion of equivalence between AFs should take into account the order in which the arguments of an extension need to be presented for it to be admissible.

Recently, the notion of *serialisability* for admissibility-based semantics has been introduced [19], which provides a decomposition scheme for extensions into a series of non-empty minimal admissible sets, called serialisation sequences. These serialisation sequences provide a deeper insight into the acceptability than the extension itself. In particular, they provide a view of the order in which the arguments of the extension need to be accepted, highlighting which conflicts in the argumentation framework need to be resolved before others can be addressed. From a dialectical perspective, it seems only natural to take this additional information into account when deciding the equivalence of two AFs. For instance, the AFs in Figure 1 are considered equivalent wrt. preferred semantics under the standard equivalence notion since they both only have the preferred extension $\{a, c\}$. However, from a dialectical point of view, it would be sensible to consider them not equivalent, since the reason for the acceptance of this extension clearly differs for both AFs. In F_1 the argument a defends c, while it is the other way around in G_1.

Figure 1. The AFs F_1 and G_1 which are equivalent wrt. preferred semantics.

A stricter equivalence notion is needed to distinguish the AFs F_1 and G_1 from above. One prominent approach to distinguish these kinds of AFs is strong equivalence [16], where two AFs F and G are only considered equivalent if they possess the same extensions after being conjoined with some arbitrary AF H (see Section 3). For that reason, the AFs in Figure 1 are not strongly equivalent wrt. preferred semantics. However, the strictness of this equivalence notion means that, for example, the AFs F_2 and G_2 from Figure 2 are not strongly equivalent wrt. preferred semantics, even though they can reasonably be considered equivalent. Both frameworks have a defending d against b and c, with the only difference being the two attacks between the always rejected arguments b and c. From a dialectical point of view, these attacks are insignificant as they do not influence the acceptable sets of argument, and so we might want to consider these AFs to be equivalent. In particular, one might argue that the underlying reasoning behind the only preferred conclusion $\{a, d\}$ is the same in both F_2 and G_2: a refutes both b and c and thus makes d acceptable.

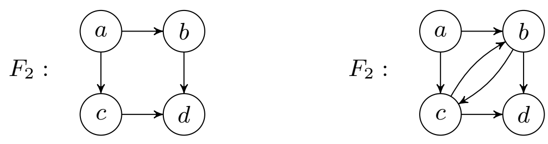


Figure 2. The AFs F_2 and G_2 which are not strongly equivalent wrt. preferred semantics.

In this work, we introduce the notion of *serialisation equivalence*, which provides a stronger notion of equivalence than the standard equivalence notion by incorporating the additional implicit information about the order of argument acceptance and the order of conflict resolution within an extension. Specifically, serialisation equivalence provides a notion to compare AFs based on the actual underlying reasoning that they represent and not only whether they come to the same semantical conclusions. We investigate the relationship between serialisation equivalence wrt. different semantics and to strong and standard equivalence. In particular, our results show that serialisation equivalence is generally at least as strict as standard equivalence and less strict than strong equivalence. Interestingly, we have that standard and serialisation equivalence coincide for admissibility, along with serialisation equivalence wrt. preferred semantics. Furthermore, we analyse serialisation equivalence in terms of complexity, which turns out to be as hard as deciding the complement problem of credulous acceptance.

To summarise, the main contributions of this work are as follows:

1. We characterise serialisation equivalence and investigate the relations between the equivalence wrt. different semantics (Section 4.1).
2. We examine its relation to standard and strong equivalence for a wide series of semantics (Section 4.2).
3. We provide a full analysis of the computational complexity of deciding serialisation equivalence (Section 5).

In Section 2 we recall the necessary background on argumentation frameworks and serialisability and in Section 3 we discuss relevant equivalence notions from the literature. Section 6 concludes the paper.

The proofs of all technical results can be found in the supplementary material[1].

2 Preliminaries

An *(abstract) argumentation framework* (AF) is a tuple $F = (\mathcal{A}, \mathcal{R})$ where $\mathcal{A}$ is a finite set of arguments and $\mathcal{R}$ is a relation $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$ [11]. For two arguments $a, b \in \mathcal{A}$, the relation $a\mathcal{R}b$ means that a attacks b. For a set $S \subseteq \mathcal{A}$, we denote by $F|_S = (S, \mathcal{R} \cap (S \times S))$ the projection of F on S. For a set $S \subseteq \mathcal{A}$ we define $S_F^+ = \{a \in \mathcal{A} \mid \exists b \in S : b\mathcal{R}a\}$ and $S_F^- = \{a \in \mathcal{A} \mid \exists b \in S : a\mathcal{R}b\}$. For a singleton set S, we omit brackets for readability, i. e., we write a_F^- (a_F^+). For two sets S and S' we write $S\mathcal{R}S'$ iff $S' \cap S_F^+ \neq \emptyset$. We say that a set $S \subseteq \mathcal{A}$ is *conflict-free* iff for all $a, b \in S$ it is not the case that $a\mathcal{R}b$. A set S *defends* an argument $b \in \mathcal{A}$ iff for all a with $a\mathcal{R}b$ there is $c \in S$ with $c\mathcal{R}a$. Furthermore, a set S is called *admissible* (ad) iff it is conflict-free and S defends all $a \in S$.

We define different semantics by imposing constraints on admissible sets [2]. In particular, an admissible set E is

- *complete* (co) iff for all $a \in \mathcal{A}$, if E defends a then $a \in E$,
- *grounded* (gr) iff E is complete and minimal,
- *preferred* (pr) iff E is maximal,
- *stable* (st) iff $E \cup E_F^+ = \mathcal{A}$,
- a *strongly admissible* (sa) iff $E = \emptyset$ or each $a \in E$ is defended by some strongly admissible $E' \subseteq E \setminus \{a\}$.

All statements on minimality/maximality are meant to be wrt. set inclusion. For the context of this work we consider ad to be a semantics itself. For any semantics σ let $\sigma(F)$ denote the set of σ-extensions of F.

Non-empty minimal admissible sets have been coined *initial sets* by Xu and Cayrol [22].

[1] https://mthimm.de/misc/lbjsmt_ecai24_appendix.pdf

Definition 1. *For $F = (\mathcal{A}, \mathcal{R})$, a set $S \subseteq \mathcal{A}$ with $S \neq \emptyset$ is called an* initial set *if S is admissible and there is no admissible $S' \subsetneq S$ with $S' \neq \emptyset$.*

With $\mathsf{IS}(F)$ we denote the set of initial sets of F. We differentiate between three types of initial sets [19].

Definition 2. *For $F = (\mathcal{A}, \mathcal{R})$ and $S \in \mathsf{IS}(F)$, we say that*

1. *S is* unattacked *iff $S^- = \emptyset$,*
2. *S is* unchallenged *iff $S^- \neq \emptyset$ and $\nexists S' \in \mathsf{IS}(F)$ with $S'\mathcal{R}S$,*
3. *S is* challenged *iff $\exists S' \in \mathsf{IS}(F)$ with $S'\mathcal{R}S$.*

In the following, we denote with $\mathsf{IS}^{\not\leftarrow}(F)$, $\mathsf{IS}^{\not\leftrightarrow}(F)$, and $\mathsf{IS}^{\leftrightarrow}(F)$ the set of unattacked, unchallenged, and challenged initial sets, respectively. Furthermore, we recall the definition of the reduct [6].

Definition 3. *For $F = (\mathcal{A}, \mathcal{R})$ and $S \subseteq \mathcal{A}$, the S-reduct F^S is defined via $F^S = F|_{A \setminus (S \cup S^+)}$.*

Example 1. *Consider the AF F_3 in Figure 3. The minimal admissible sets of F_3 are $\{a\}$, $\{d\}$ and $\{f\}$. We have that $\{a\}$ and {d} attack each other, meaning both are challenged initial sets. On the other hand, $\{f\}$ is attacked by c, but since c is not in any admissible set itself it follows that $\{f\}$ is an unchallenged initial set of F_3. Consider, for instance, the $\{d\}$-reduct $F_3^{\{d\}} = (\{b, c, f\}, \{(b, c), (c, c), (c, f), (f, c)\})$. In this AF, $\{f\}$ is still an unchallenged initial set, but in addition to that $\{b\}$ is now an unattacked initial set of $F_3^{\{d\}}$, since its only attacker e is no longer part of the AF.*

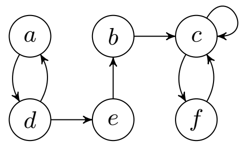
F_3 :


Figure 3. The AF F_3 with the initial sets $\{a\}$, $\{d\}$ and $\{f\}$.

We recall *serialisability* [19, 8], which is a property of a semantics that allows us to characterise extensions in a constructive manner. For that we define the *serialisation sequence* $\mathcal{S}$ which is a decomposition of an extension into a series of initial sets.

Definition 4. *A* serialisation sequence *for $F = (\mathcal{A}, \mathcal{R})$ is a sequence $\mathcal{S} = (S_1, \ldots S_n)$ with $S_1 \in \mathsf{IS}(F)$ and for each $2 \leq i \leq n$ we have that $S_i \in \mathsf{IS}(F^{S_1 \cup \cdots \cup S_{i-1}})$.*

As shown in [8], a serialisation sequence $(S_1, \ldots S_n)$ induces an admissible set $E = S_1 \cup \cdots \cup S_n$ and for every admissible set there is at least one such sequence. Recalling the distinction between initial sets from Definition 2, one can further restrict the serialisation sequences to characterise a wide variety of admissibility-based semantics [19].

Theorem 1. *Let $F = (\mathcal{A}, \mathcal{R})$ be an argumentation framework and $E \subseteq \mathcal{A}$. We have that:*

- *$E \in \mathsf{ad}(F)$ if and only if there is a serialisation sequence $(S_1, \ldots, S_n)$ with $E = S_1 \cup \cdots \cup S_n$.*
- *$E \in \mathsf{co}(F)$ if and only if there is a serialisation sequence $(S_1, \ldots, S_n)$ with $E = S_1 \cup \cdots \cup S_n$ and it holds that $\mathsf{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_n}) = \emptyset$.*

- $E \in \mathit{gr}(F)$ *if and only if there is a serialisation sequence* $(S_1, \ldots, S_n)$ *with* $E = S_1 \cup \cdots \cup S_n$ *and for all* S_i, $i = 1, \ldots, n$, *it holds that* $S_i \in \mathit{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_{i-1}})$ *and it holds that* $\mathit{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_n}) = \emptyset$.
- $E \in \mathit{pr}(F)$ *if and only if there is a serialisation sequence* $(S_1, \ldots, S_n)$ *with* $E = S_1 \cup \cdots \cup S_n$ *and it holds that* $\mathit{IS}(F^{S_1 \cup \cdots \cup S_n}) = \emptyset$.
- $E \in \mathit{st}(F)$ *if and only if there is a serialisation sequence* $(S_1, \ldots, S_n)$ *with* $E = S_1 \cup \cdots \cup S_n$ *and it holds that* $F^{S_1 \cup \cdots \cup S_n} = (\emptyset, \emptyset)$.
- $E \in \mathit{sa}(F)$ *if and only if there is a serialisation sequence* $(S_1, \ldots, S_n)$ *with* $E = S_1 \cup \cdots \cup S_n$ *and for all* S_i, $i = 1, \ldots, n$, *it holds that* $S_i \in \mathit{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_{i-1}})$.

Furthermore, we also consider the unchallenged semantics [19, 7], which is directly defined in terms of serialisation sequences.

Definition 5. *Let* $F = (\mathcal{A}, \mathcal{R})$ *be an AF. Then* $\mathit{uc}(F)$ *is the set that contains exactly those sets* $E \subseteq \mathcal{A}$ *for which there is a serialisation sequence* $(S_1, \ldots, S_n)$ *with* $E = S_1 \cup \cdots \cup S_n$ *and for all* S_i *it holds that* $S_i \in \mathit{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_{i-1}}) \cup \mathit{IS}^{\not\leftrightarrow}(F^{S_1 \cup \cdots \cup S_{i-1}})$ *and it holds that* $\mathit{IS}^{\not\leftarrow}(F^{S_1 \cup \cdots \cup S_n}) \cup \mathit{IS}^{\not\leftrightarrow}(F^{S_1 \cup \cdots \cup S_n}) = \emptyset$.

We denote with $\Sigma = \{\mathsf{ad}, \mathsf{co}, \mathsf{gr}, \mathsf{st}, \mathsf{pr}, \mathsf{sa}, \mathsf{uc}\}$ the set of *serialisable* semantics. For an argumentation framework F, we denote with $\mathfrak{S}_\sigma(F)$ the set of all serialisation sequences of F wrt. the semantics σ.

Example 2. *Consider the AF* F_3 *in Figure 3. We have that* $\{b, d, f\}$ *is a preferred extension of* F_3. *Corresponding to that, we have the preferred serialisation sequence* $\mathcal{S}_1 = (\{d\}, \{b\}, \{f\})$. *As explained in Example 1,* $\{d\}$ *is an initial set of* F_3 *and* $\{b\}$ *is an initial set of* $F_3^{\{d\}}$. *Clearly,* $\{f\}$ *is an initial set of* $F_3^{\{b,d\}} = (\{f\}, \emptyset)$ *and* $\mathit{IS}(F_3^{\{b,d,f\}}) = \emptyset$. *Note that* $\mathcal{S}_2 = (\{f\}, \{d\}, \{b\})$ *is the only other preferred serialisation sequence for the extension* $\{b, d, f\}$.

3 Equivalence in Abstract Argumentation

We now introduce some equivalence notions for argumentation frameworks from the literature. First, we consider *standard equivalence* wrt. some semantics σ, which is simply defined via the equivalence of the set of σ-extensions.

Definition 6. *Let* F *and* G *be argumentation frameworks. We say that* F *and* G *are* equivalent *to each other wrt. a semantics* σ, *written as* $F \equiv_\sigma G$, *if and only if we have that* $\sigma(F) = \sigma(G)$.

Example 3. *Consider the AFs* F_1 *and* G_1 *in Figure 1. We have that* $F_1 \equiv_{\mathit{pr}} G_1$, *since* $\mathit{pr}(F_1) = \mathit{pr}(G_1) = \{\{a, c\}\}$. *On the other hand, we have* $F_1 \not\equiv_{\mathit{ad}} G_1$ *since* $\{a\} \in \mathit{ad}(F_1)$ *and* $\{a\} \notin \mathit{ad}(G_1)$.

A more restrictive approach to equivalence is provided by the notion of *strong equivalence* wrt. some semantics σ. It was originally introduced with the intention of providing a notion of equivalence in a dynamic argumentation scenario [16]. Intuitively, strong equivalence requires that both frameworks are still equivalent after being conjoined with another framework H. For two argumentation frameworks $F = (\mathcal{A}, \mathcal{R})$ and $F' = (\mathcal{A}', \mathcal{R}')$ we define $F \cup F' = (\mathcal{A} \cup \mathcal{A}', \mathcal{R} \cup \mathcal{R}')$.

Definition 7. *Let* F *and* G *be argumentation frameworks. We say that* F *and* G *are* strongly equivalent *to each other wrt. a semantics* σ, *written as* $F \equiv^s_\sigma G$, *if and only if for each argumentation framework* H, *we have that* $\sigma(F \cup H) = \sigma(G \cup H)$.

To facilitate deciding strong equivalence wrt. a semantics σ, so called *kernels* have been introduced that remove semantically redundant attacks from an AF [16]. The different kernels $k \in \{sk, ak, gk, ck\}$ of an AF $F = (\mathcal{A}, \mathcal{R})$ are defined as $F^k = (\mathcal{A}, \mathcal{R}^k)$, with

$$\mathcal{R}^{sk} = \mathcal{R} \setminus \{(a, b) \mid a \neq b, (a, a) \in \mathcal{R}\} \quad (1)$$

$$\mathcal{R}^{ak} = \mathcal{R} \setminus \{(a, b) \mid a \neq b, (a, a) \in \mathcal{R}, \{(b, a), (b, b)\} \cap \mathcal{R} \neq \emptyset\} \quad (2)$$

$$\mathcal{R}^{gk} = \mathcal{R} \setminus \{(a, b) \mid a \neq b, (b, b) \in \mathcal{R}, \{(a, a), (b, a)\} \cap \mathcal{R} \neq \emptyset\} \quad (3)$$

$$\mathcal{R}^{ck} = \mathcal{R} \setminus \{(a, b) \mid a \neq b, (a, a) \in \mathcal{R}, (b, b) \in \mathcal{R}\} \quad (4)$$

As the following theorem from [16] shows, strong equivalence can be characterised by kernel identity.

Theorem 2. *Let* F *and* G *be argumentation frameworks. It holds that*

1. $F^{sk} = G^{sk}$ *if and only if* $F \equiv^s_{\mathit{st}} G$,
2. $F^{ak} = G^{ak}$ *if and only if* $F \equiv^s_{\mathit{ad}} G$,
3. $F^{ak} = G^{ak}$ *if and only if* $F \equiv^s_{\mathit{pr}} G$,
4. $F^{gk} = G^{gk}$ *if and only if* $F \equiv^s_{\mathit{gr}} G$,
5. $F^{ck} = G^{ck}$ *if and only if* $F \equiv^s_{\mathit{co}} G$.

While the strongly admissible semantics has not been considered in [16], the following result follows easily.

Proposition 1. *Let* F *and* G *be argumentation frameworks. It holds that* $F^{gk} = G^{gk}$ *if and only if* $F \equiv^s_{\mathit{sa}} G$.

As a side product of the above, we have that the σ-extensions of an AF and its respective kernel framework are the same.

Corollary 1. *Let* F *be an AF. Then it holds that*

1. $\mathit{st}(F) = \mathit{st}(F^{sk})$,
2. $\sigma(F) = \sigma(F^{ak})$, *for* $\sigma \in \{\mathit{ad}, \mathit{pr}\}$,
3. $\sigma(F) = \sigma(F^{gk})$, *for* $\sigma \in \{\mathit{gr}, \mathit{sa}\}$,
4. $\mathit{co}(F) = \mathit{co}(F^{ck})$.

Example 4. *Consider the AFs* F_4 *and* G_4 *in Figure 4. For the admissible semantics, the kernel* F_4^{ak} *of* F_4 *removes only the attack* (b, c), *as depicted in Figure 4. For the* ad*-kernel* G_4^{ak} *of* G_4, *we remove the attack* (b, a) *from* G_4, *thus we have that* $F_4^{ak} = G_4^{ak}$ *and it follows that* $F_4 \equiv^s_{\mathit{ad}} G_4$. *The same holds true for the stable semantics, i. e.,* $F_4 \equiv^s_{\mathit{st}} G_4$. *Conversely, we have that* $F_4 \not\equiv^s_{\mathit{gr}} G_4$, *since* $F_4^{gk} = (\{a, b, c\}, \{(a, b), (b, b), (b, c)\})$ *and* $G_4^{gk} = (\{a, b, c\}, \{(b, a), (b, b), (c, b)\})$.

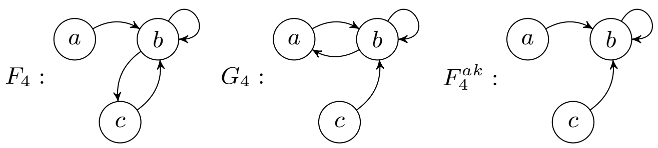


Figure 4. The AFs F_4 and G_4 and their ad-kernel framework.

4 Serialisation Equivalence

We now introduce a novel notion of equivalence that compares argumentation frameworks based on their serialisation sequences (wrt. some $\sigma \in \Sigma$).

Definition 8. *Let F and G be argumentation frameworks. We say that F and G are* serialisation equivalent *to each other wrt. a semantics σ, written as $F \equiv_{\sigma}^{se} G$, if and only if we have that $\mathfrak{S}_\sigma(F) = \mathfrak{S}_\sigma(G)$.*

Intuitively, under serialisation equivalence we take into account every possible sequence in which the arguments of each extension E can be accepted in order to construct the extension. That is, we are considering more dialectical information, in the form of the order in which arguments of an extension may be accepted, when comparing argumentation frameworks.

With this new notion of serialisation equivalence, we can now appropriately handle the argumentation frameworks from Section 1, as shown by the following example.

Example 5. *With this equivalence notion, we can now distinguish between the AFs F_1 and G_1 in Figure 1. We have that $F_1 \not\equiv_{\sigma}^{se} G_1$ for any $\sigma \in \Sigma$. For instance, the only preferred serialisation sequence of F_1 is $(\{a\}, \{c\})$, while the only preferred sequence of G_1 is $(\{c\}, \{a\})$, highlighting that the extension $\{a, c\}$ is constructed differently in both frameworks.*

Consider now the AFs F_2 and G_2 in Figure 2. We have that $F_2 \equiv_{\sigma}^{se} G_2$ for any semantics $\sigma \in \Sigma$, because, for example, the only preferred serialisation sequence of both F_2 and G_2 is $(\{a\}, \{d\})$.

Example 6. *Consider the AFs F_5, G_5 and H_5 in Figure 5. For the complete semantics, we have that $F_5 \equiv_{co}^{se} G_5$. For both AFs there exist two serialisation sequences: the empty sequence $\mathcal{S}_0$ and $\mathcal{S}_1 = (\{d\}, \{a\})$. On the other hand we have $F_5 \not\equiv_{co}^{se} H_5$, because the AF H_5 has, besides $\mathcal{S}_0$ and $\mathcal{S}_1$, the complete serialisation sequences $\mathcal{S}_2 = (\{a\}, \{d\})$ and $\mathcal{S}_3 = (\{a\})$. Note that we also have $F_5 \not\equiv_{co} H_5$, since $\{a\} \notin co(F_5)$. Clearly, the same holds for G_5 and H_5.*

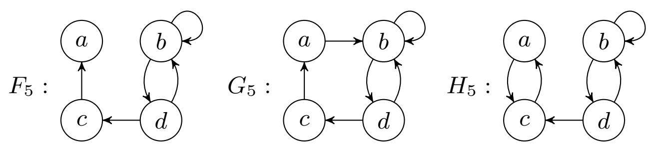


Figure 5. The AFs F_5, G_5 and H_5 from Example 6.

In the following, we analyse the relationships between serialisation equivalence wrt. all semantics in Σ. Afterwards, we investigate the relation of serialisation equivalence to existing equivalence notions, namely standard equivalence and strong equivalence (Section 4.2). We then provide a comprehensive overview over the relationships between the three types of equivalences wrt. all considered semantics.

4.1 Relation between Serialisation Equivalences wrt. different Semantics

Table 1 shows the relations between all semantics wrt. serialisation equivalence. Interestingly, we have that serialisation equivalence wrt. admissible and preferred semantics coincide. The same holds true for the strongly admissible and grounded semantics. Analogously to strong equivalence, serialisation equivalence wrt. complete semantics implies serialisation equivalence wrt. admissible (preferred) semantics, a relation that does not hold for standard equivalence. Serialisation equivalence wrt. the unchallenged semantics is implied by admissible (preferred) and thus, by transitivity, also by the complete semantics. These results are summarised in Theorem 3 (its

Table 1. Relations between serialisation equivalence wrt. different semantics. For line σ_1 and column σ_2, the table reads as 'serialisation equivalence wrt. σ_1 implies serialisation equivalence wrt. σ_2'. A ✓ indicates that the relation holds for all AFs, while ✗ indicates that it does not hold and the corresponding superscript refers to the counterexample below.

⇒	ad	co	gr	pr	st	uc	sa
ad	✓	✗^{7}	✗^{7}	✓	✗^{7}	✓	✗^{7}
co	✓	✓	✗^{8}	✓	✗^{8}	✓	✗^{8}
gr	✗^{9}	✗^{9}	✓	✗^{9}	✗^{9}	✗^{10}	✓
pr	✓	✗^{7}	✗^{7}	✓	✗^{7}	✓	✗^{7}
st	✗^{11}	✗^{11}	✗^{11}	✗^{11}	✓	✗^{11}	✗^{11}
uc	✗^{9}	✗^{7}	✗^{7}	✗^{9}	✗^{7}	✓	✗^{7}
sa	✗^{9}	✗^{9}	✓	✗^{9}	✗^{9}	✗^{10}	✓

proof can be found in the supplementary material). All other relations do not hold in general, as shown by the following series of examples. Most notably, while standard (strong) equivalence wrt. grounded semantics is implied by equivalence wrt. complete semantics, this relation does not hold for serialisation equivalence (see Example 8).

Theorem 3. *Let F, G be AFs. Then it holds that*

1. *$F \equiv_{ad}^{se} G$ if and only if $F \equiv_{pr}^{se} G$,*
2. *If $F \equiv_{ad}^{se} G$, then $F \equiv_{uc}^{se} G$,*
3. *If $F \equiv_{pr}^{se} G$, then $F \equiv_{uc}^{se} G$,*
4. *If $F \equiv_{co}^{se} G$, then $F \equiv_{pr}^{se} G$,*
5. *If $F \equiv_{co}^{se} G$, then $F \equiv_{ad}^{se} G$,*
6. *If $F \equiv_{co}^{se} G$, then $F \equiv_{uc}^{se} G$,*
7. *$F \equiv_{sa}^{se} G$ if and only if $F \equiv_{gr}^{se} G$.*

Note that, in all of the following examples self-attacks are only used for the sake of making the examples easier to read. They can be substituted by any structure that contains only non-acceptable arguments, e. g., odd cycles. This is one of the advantages of serialisation equivalence when compared to strong equivalence, which relies entirely on the existence of self-attacks.

In general, the non-implications between semantics come down to how the different types of initial sets are treated by the semantics (see Theorem 1). In particular, in the Examples 7 and 8 we have an unattacked initial set in the first AF which is only unchallenged initial in the second AF. That then leads to different serialisation sequences for the semantics where initial sets of a different type are actually treated differently. The grounded and strongly admissible semantics allow only unattacked initial sets to be selected and a serialisation sequence is only in $\mathfrak{S}_{co}$ iff there are no unattacked initial sets left.

Example 7. *Consider the AFs F_6 and G_6 in Figure 6. We have that F_6 and G_6 are serialisation equivalent wrt. $\sigma \in \{ad, pr, uc\}$, but not serialisation equivalent wrt. $\sigma \in \{co, gr, sa, st\}$. For instance, the ad, pr and uc serialisation sequences for F_6 and G_6 are the empty sequence and $(\{a\})$, while for gr and sa F_6 has only the serialisation sequence $(\{a\})$ and G_6 has only the empty sequence.*

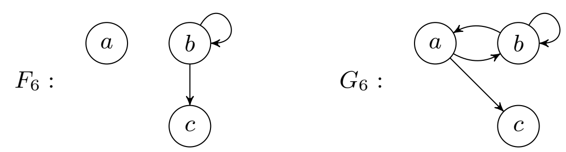


Figure 6. The AFs F_6 and G_6 from Example 7.

Example 8. *Consider the AFs F_7 and G_7 in Figure 7. We have that F_7 and G_7 are serialisation equivalent wrt. $\sigma \in \{ad, co, pr, uc\}$,* *but not serialisation equivalent wrt. $\sigma \in \{\mathsf{gr}, \mathsf{st}, \mathsf{sa}\}$. Both AFs have two complete sequences: $(\{c\}, \{b\})$ and $(\{b\}, \{c\})$. However, we have that $(\{b\}) \in \mathsf{IS}^{\not\leftarrow}(G_7)$ and thus $(\{b\}, \{c\})$ is not a grounded sequence of G_7. Note that $(\{c\}, \{b\})$ is still a grounded serialisation sequence of G_7 since in the reduct $G_7^{\{c\}}$ the set $\{b\}$ is then unattacked initial. Highlighting nicely the difference that in a complete extension an argument may defend itself, while for the grounded (and every strongly admissible) extension arguments must always be defended by other arguments in the extension. This information is intrinsically contained in the grounded and strongly admissible serialisation sequences.*

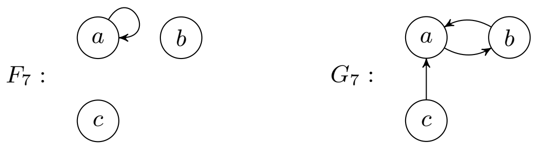


Figure 7. The AFs F_7 and G_7 from Example 8.

As the following example shows, for the semantics gr, sa and uc the fact that they disregard challenged initial sets comes into play and rules out an implication to equivalence wrt. the other semantics.

Example 9. *Consider the AFs F_8 and G_8 in Figure 8. We have that F_8 and G_8 are serialisation equivalent wrt. $\sigma \in \{\mathsf{gr}, \mathsf{uc}, \mathsf{sa}\}$, but not serialisation equivalent wrt. $\sigma \in \{\mathsf{ad}, \mathsf{co}, \mathsf{pr}, \mathsf{st}\}$. F_8 has only two* ad *serialisation sequences: the empty sequence and $(\{a\})$. Clearly, for G_8 we additionally have multiple sequences containing the challenged initial sets $\{b\}$ or $\{c\}$, e. g., $(\{a\}, \{b\})$ or $(\{c\})$. The same holds for all semantics that allow challenged initial sets to be included.*

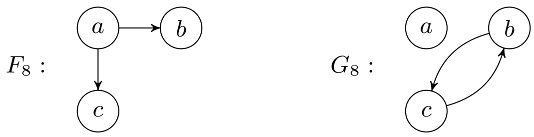


Figure 8. The AFs F_8 and G_8 from Example 9.

Since the grounded and strongly admissible semantics also disregard unchallenged initial sets, that rules out an implication onto the unchallenged semantics as shown in the following example.

Example 10. *Consider the AFs F_9 and G_9 in Figure 9. We have that F_9 and G_9 are serialisation equivalent wrt. $\sigma \in \{\mathsf{gr}, \mathsf{sa}\}$, but not serialisation equivalent wrt. $\sigma \in \{\mathsf{ad}, \mathsf{co}, \mathsf{pr}, \mathsf{st}, \mathsf{uc}\}$. Note that $\{b\} \in \mathsf{IS}^{\not\leftarrow}(G_9)$ but $\{b\} \notin \mathsf{IS}(F_9)$. That allows all semantics besides* gr *and* sa *to select it as the first set in a serialisation sequence. So $(\{c\}, \{b\})$ is a sequence for all semantics, while $(\{b\}, \{c\})$ is a serialisation sequence of G_9 but not F_9 for* ad, co, pr, st *and* uc.

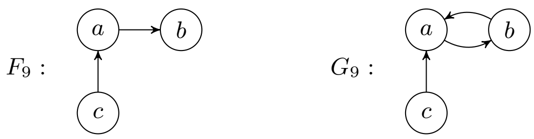


Figure 9. The AFs F_9 and G_9 from Example 10.

Finally, for the stable semantics the fact that no serialisation sequence may exist comes into play which rules out any implication in both directions, cf. Examples 7 and 11.

Example 11. *Consider the AFs F_{10} and G_{10} in Figure 10. We have that F_{10} and G_{10} are serialisation equivalent wrt. $\sigma \in \{\mathsf{st}\}$, but not serialisation equivalent wrt. $\sigma \in \{\mathsf{ad}, \mathsf{co}, \mathsf{gr}, \mathsf{pr}, \mathsf{uc}, \mathsf{sa}\}$. Both AFs have no stable sequence, while F_{10} has the sequence $(\{b\})$ for all of the other semantics and G_{10} only the empty sequence.*

Figure 10. The AFs F_{10} and G_{10} from Example 11.

4.2 *Relation to Standard and Strong Equivalence*

In the following, we show that our notion of serialisation equivalence constitutes a middle ground between the very loose standard equivalence and the very strict strong equivalence. As shown by Theorem 4, serialisation equivalence is at least as strict as standard equivalence for all semantics, i. e., serialisation sequence wrt. some semantics σ implies standard equivalence wrt. the same semantics.

Theorem 4. *Let $\sigma \in \Sigma$ be a semantics. For any two argumentation frameworks F and G, if $F \equiv_\sigma^{se} G$, then it follows that $F \equiv_\sigma G$.*

Conversely, the other direction does not hold in general for most of the serialisable semantics, as shown by the following example.

Example 12. *Consider the argumentation frameworks F_1 and G_1 from Figure 1. We have that $\mathsf{pr}(F_1) = \mathsf{gr}(F_1) = \mathsf{co}(F_1) = \mathsf{st}(F_1) = \mathsf{uc}(F_1) = \{\{a, c\}\} = \mathsf{pr}(G_1) = \mathsf{gr}(G_1) = \mathsf{co}(G_1) = \mathsf{st}(G_1) = \mathsf{uc}(G_1)$. Thus F_1 and G_1 are equivalent wrt.* pr, gr, co, st *and* uc *semantics. However, they are not serialisation equivalent wrt. any of those semantics. For all of the above semantics we only have one serialisation sequence for the extension $\{a, c\}$ in F_1, namely $(\{a\}, \{c\})$. On the other hand, for G_1 the only serialisation sequence is $(\{c\}, \{a\})$ for all of the above semantics. Thus, $F \equiv_\sigma G$ does not imply $F \equiv_\sigma^{se} G$ in general for $\sigma \in \{\mathsf{pr}, \mathsf{gr}, \mathsf{co}, \mathsf{st}, \mathsf{uc}\}$.*

Notably, only for the (strong) admissible semantics does the reverse direction also hold, as stated in Propositions 2 and 3. This means, standard equivalence and serialisation equivalence wrt. admissible semantics coincide, along with serialisation equivalence wrt. preferred semantics. The same is true for standard and serialisation equivalence wrt. the strongly admissible semantics and serialisation equivalence wrt. grounded semantics.

Proposition 2. *Let F, G be argumentation frameworks. It holds that $\mathsf{ad}(F) = \mathsf{ad}(G)$ if and only if $\mathfrak{S}_{\mathsf{ad}}(F) = \mathfrak{S}_{\mathsf{ad}}(G)$.*

Proposition 3. *Let F, G be argumentation frameworks. It holds that $\mathsf{sa}(F) = \mathsf{sa}(G)$ if and only if $\mathfrak{S}_{\mathsf{sa}}(F) = \mathfrak{S}_{\mathsf{sa}}(G)$.*

So we have that serialisation equivalence is generally stricter than standard equivalence for all semantics with the exception of admissibility and strong admissibility, where it coincides with standard equivalence. This ties in nicely with the dialectical aspects mentioned in the introduction. Essentially, standard equivalence only takes into account the possible conclusions (i. e., the extensions) of the discussion modelled in the AF. On the other hand, under serialisation equivalence the order in which the conflicts have to be resolved is taken into consideration. This means serialisation equivalence provides a notion to determine if two AFs represent not only the same conclusions but also if they are justified in the same way.

Before we turn to the relation of serialisation equivalence and strong equivalence, we consider first the unchallenged semantics for which no kernel characterisation has been established yet. As the following result in Proposition 4 shows, strong equivalence wrt. unchallenged semantics coincides with strong equivalence wrt. admissible (and preferred) semantics. That means strong equivalence wrt. unchallenged semantics is characterised by the admissible kernel.

Proposition 4. *Let F, G be argumentation frameworks. It holds that*

$$F \equiv^s_{ad} G \text{ iff } F \equiv^s_{pr} G \text{ iff } F \equiv^s_{uc} G.$$

We examine now the relationship between serialisation and strong equivalence (wrt. some semantics σ). As it turns out, we have that strong equivalence implies serialisation equivalence wrt. the same semantics. Only the stable semantics presents a special case. We show the implication of strong equivalence onto serialisation equivalence via the syntactic kernel characterisation of strong equivalence for the different semantics (see Equations (1)-(4)). In particular, we show that the set of serialisation sequences $\mathfrak{S}_\sigma(F)$ of an AF is always the same as the serialisation sequences $\mathfrak{S}_\sigma(F^k)$ of its kernel framework F^k wrt. the corresponding strong equivalence kernel.

Lemma 1. *Let $F = (\mathcal{A}, \mathcal{R})$ be an argumentation framework. Then the following statements hold*

1. $\mathfrak{S}_{ad}(F) = \mathfrak{S}_{ad}(F^{ak})$,
2. $\mathfrak{S}_{pr}(F) = \mathfrak{S}_{pr}(F^{ak})$,
3. $\mathfrak{S}_{uc}(F) = \mathfrak{S}_{uc}(F^{ak})$,
4. $\mathfrak{S}_{sa}(F) = \mathfrak{S}_{sa}(F^{gk})$,
5. $\mathfrak{S}_{gr}(F) = \mathfrak{S}_{gr}(F^{gk})$,
6. $\mathfrak{S}_{co}(F) = \mathfrak{S}_{co}(F^{co})$.

The following theorem summarises the implication between strong equivalence and serialisation equivalence.

Theorem 5. *Let $\sigma \in \{\mathsf{ad}, \mathsf{pr}, \mathsf{uc}, \mathsf{sa}, \mathsf{gr}, \mathsf{co}\}$ be a semantics. For any two argumentation frameworks F and G, if $F \equiv^s_\sigma G$ then it follows that $F \equiv^{se}_\sigma G$.*

The reverse direction does not hold for any serialisable semantics in general, i. e., serialisation equivalence does not imply strong equivalence for any semantics $\sigma \in \Sigma$.

Example 13. *Consider the AFs F_{11} and G_{11} in Figure 11. For any semantics $\sigma \in \Sigma$ we have that $F_{11} \equiv^{se}_\sigma G_{11}$ and $F_{11} \not\equiv^s_\sigma G_{11}$. For all semantics, there exists only one serialisation sequence $(\{a\})$ for the extension $\{a\}$ for both AFs. While the empty set is also admissible, trivially it also has the same empty serialisation sequence in both AFs. On the other hand F_{11} and G_{11} are not strongly equivalent wrt. any of the above semantics. For any semantics σ, consider the AF $H = (\{a, c\}, \{(c, a)\})$. Then $\sigma(F_{11} \cup H) = \{\{c\}\} \neq \{\{b, c\}\} = \sigma(G_{11} \cup H)$ and thus $F_{11} \not\equiv^s_\sigma G_{11}$.*

Figure 11. The AFs F_{11} and G_{11} which are serialisation equivalent wrt. $\sigma \in \{\mathsf{ad}, \mathsf{pr}, \mathsf{sa}, \mathsf{gr}, \mathsf{co}, \mathsf{st}, \mathsf{uc}\}$, but not strongly equivalent wrt. those semantics.

As the following example shows, for the case of stable semantics strong equivalence of two AFs does not necessarily imply serialisation equivalence wrt. stable semantics.

Example 14. *Consider the AFs F_{12} and G_{12} in Figure 12. G_{12} is the sk-kernel of F_{12}, thus both AFs are strongly equivalent wrt. stable semantics. However, they are not serialisation equivalent wrt. stable semantics. The only stable serialisation sequence for F_{12} is $\mathcal{S}_1 = (\{c\}, \{a\})$, while G_{12} has two stable serialisation sequences $\mathcal{S}_1$ and $\mathcal{S}_2 = (\{a\}, \{c\})$. In other words, under strong equivalence, the information that c defends a gets lost in the kernel framework while serialisation equivalence takes this into account.*

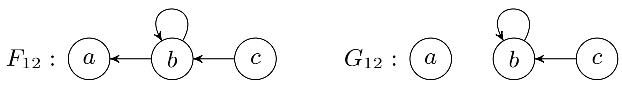


Figure 12. The AFs F_{12} and G_{12} are strongly equivalent wrt. stable semantics but not serialisation equivalent.

As follows from Examples 13 and 14, for the stable semantics we can establish that serialisation equivalence and strong equivalence are independent of each other.

Proposition 5. *Let F an G be argumentation frameworks. Then $F \equiv^{se}_{st} G$ does not generally imply $F \equiv^s_{st} G$ and vice versa.*

The above results and the examples highlight an important advantage of serialisation equivalence compared to strong equivalence. Recall that strong equivalence can be decided on a syntactical level by simply comparing the kernels. For every kernel, this comes down to checking whether some attacks from or to self-attacking arguments are redundant. That means, in the absence of self-attacks strong equivalence collapses to exact syntactic equivalence of the graphs. This is not the case for serialisation equivalence, which yields the same distinctive equivalence notions for odd-cycle-free argumentation frameworks (except equivalence wrt. the stable semantics, which coincides with preferred semantics in such frameworks).

4.3 Summary of Results

Together with the results from Section 4.1, we can now establish a comprehensive picture of the relationships between standard, serialisation and strong equivalence wrt. to the different semantics, depicted in Figure 13. The equivalence notions are ordered by strictness from left to right. Strong equivalence being the strictest and standard equivalence being the most relaxed. The semantics are arranged from top to bottom.

Generally, all three equivalence notions behave quite similarly, but there are some interesting differences to be observed. For instance, while equivalence wrt. complete semantics implies equivalence wrt. grounded semantics for both strong and standard, this relation does not hold for serialisation equivalence. Similarly, while equivalence wrt. complete semantics implies equivalence wrt. admissible semantics for strong and serialisation equivalence, this does not hold anymore for standard equivalence. Notably, while equivalence wrt. admissible and preferred semantics coincide for strong and serialisation equivalence, this relation gets weakened for standard equivalence to just an implication. We can observe the same behaviour for the strongly admissible and grounded semantics. Perhaps most interesting the fact that standard and serialisation equivalence coincide for the admissible semantics, along with serialisation equivalence wrt. preferred semantics. Again, this is mirrored by the strongly admissible and grounded semantics.

For equivalence wrt. the unchallenged semantics, we have that it is generally implied by the respective equivalence wrt. admissible semantics. Notably, for strong equivalence we have an even stronger relation, i. e., strong equivalence wrt. unchallenged semantics coincides with strong equivalence wrt. admissible (and preferred) semantics, as shown in Proposition 4. For strong and serialisation equivalence, it is also implied by complete and preferred semantics, while this relation disappears for standard equivalence. As established in Proposition 5, there is no relationship between strong and serialisation equivalence wrt. stable semantics, but both imply standard equivalence wrt. stable semantics.

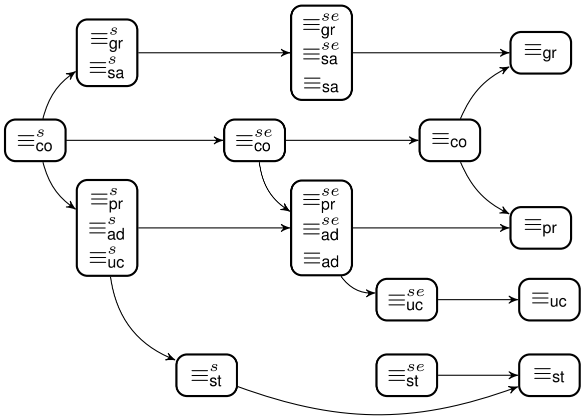


Figure 13. Comprehensive depiction of the relations between all considered equivalence notions and semantics. The non-existence of an arrow implies that no relation exists.

5 Computational Complexity

We turn now to the complexity of deciding whether two argumentation frameworks F_1, F_2 are serialisation equivalent wrt. some semantics $\sigma \in \Sigma$. For that, we assume familiarity with the basic concepts of computational complexity, in particular with the basic complexity classes P, NP and coNP [17]. Furthermore, we also consider the classes Σ_2^P and Π_2^P. The Σ_2^P class denotes decision problems that are solvable in polynomial time by a non-deterministic algorithm that has access to an NP-oracle, i. e., at each step of the algorithm it can immediately obtain the answer to a NP-complete problem. The Π_2^P class is the complementary class of Σ_2^P, i. e., $\Pi_2^P = \mathsf{co}\Sigma_2^P = \mathsf{coNP}^{\mathsf{NP}}$.

We consider the following decision problems:

Eq^{se}_σ	Given $F_1 = (\mathcal{A}_1, \mathcal{R}_1)$ and $F_2 = (\mathcal{A}_2, \mathcal{R}_2)$, decide whether $F_1 \equiv^{se}_\sigma F_2$,
coCRED$_\sigma$	Given $F = (\mathcal{A}, \mathcal{R})$ and $a \in \mathcal{A}$, decide whether there is no set $S \in \sigma(F)$ with $a \in S$.

The problem *coCRED*$_\sigma$, i. e., deciding whether a given argument a is not part of any σ-extension of F, can be reduced to the problem of deciding serialisation equivalence wrt. some semantics σ. This puts the problem Eq^{se}_σ for the grounded and strongly admissible semantics in P, while it is coNP-complete for admissible, preferred, complete and stable semantics, cf. [12]. Additionally, in accordance to [7], we have that Eq^{se}_{uc} is Π_2^P-complete.

Theorem 6.

1. *Eq^{se}_σ is coNP-complete, for $\sigma \in \{\mathsf{ad}, \mathsf{pr}, \mathsf{co}, \mathsf{st}\}$.*
2. *Eq^{se}_σ is in P, for $\sigma \in \{\mathsf{sa}, \mathsf{gr}\}$.*
3. *Eq^{se}_{uc} is Π_2^P-complete.*

For comparison, we consider the complexity of deciding strong and standard equivalence as established in [5]. Deciding strong equivalence can be done in L for any semantics. Determining standard equivalence wrt. grounded and strongly admissible semantics is P-complete, while it is coNP-complete for admissible, complete and stable semantics. Finally, standard equivalence wrt. preferred semantics is Π_2^P-complete.

That means, deciding serialisation equivalence is generally as hard as deciding standard equivalence. Only for the preferred semantics, we have that deciding serialisation equivalence is actually easier than deciding standard equivalence. So, while serialisation equivalence is on the same (or better) level of complexity as standard equivalence is still provides a more fine-grained equivalence notion that takes the order of argument acceptance into account.

6 Conclusion

In this work, we introduced the notion of *serialisation equivalence* under which two AFs are equivalent if they possess the same set of serialisation sequences wrt. some semantics σ. That enables it to better take into account the dialectical aspect of argumentation by considering the order in which the conflicts in the AF must be resolved for an extension to be accepted, thus representing an equivalence notion based on the actual reasoning inside the AF and not just the conclusions of this reasoning. We examined serialisation equivalence in detail. In particular, we analysed the relationship between serialisation equivalence wrt. different semantics and its relation to other equivalence notions from the literature, namely standard and strong equivalence [16]. As it turns out, serialisation equivalence is generally more strict than standard equivalence and less strict than strong equivalence. Only for the stable semantics we obtain an equivalence notion that is independent of the respective strong equivalence notion. Notably, for the admissible semantics semantics we have an accordance between serialisation and standard equivalence along with serialisation equivalence wrt. preferred semantics. We can observe exactly the same behaviour between strong admissibility and the grounded semantics. Furthermore, even in AFs without self-attacks serialisation equivalence yields distinct equivalence notions wrt. the different semantics, which is not the case for strong equivalence. In terms of complexity, we showed that deciding serialisation equivalence wrt. some semantics σ is as hard as deciding whether some argument is not credulously accepted wrt. σ.

While we focus our analysis in this work on standard and strong equivalence, there exist other equivalence notions in the literature [4]. These include the *expansion equivalence* of [3], which is related to strong equivalence and deals with equivalence under a restricted conjoined AF H. For example, only allowing new attacks to be added or only permitting outgoing attacks from H. Baumann et al. [2019] introduce a general parameterised notion of equivalence that subsumes standard and strong equivalence, also focusing on a dynamic argumentation scenario. Finally, there is also the notion of *defense equivalence* based on so-called defense semantics [14]. These semantics make explicit the (partial) defense of arguments through a meta AF. Two AFs are then considered equivalent if they possess the same defense structure.

In future work, we intend to investigate the relation of serialisation equivalence to the above mentioned equivalence notions from the literature. An especially interesting candidate for that are the notions of normal and strong expansion equivalence [3], which have also been shown to be between strong and standard equivalence.

Acknowledgements

The research reported here was partially supported by the Deutsche Forschungsgemeinschaft (grant 375588274).

References

[1] C. Antaki and I. Leudar. Explaining in conversation: Towards an argument model. *European Journal of Social Psychology*, 22(2):181–194, 1992.

[2] P. Baroni, M. Caminada, and M. Giacomin. Abstract argumentation frameworks and their semantics. In P. Baroni, D. Gabbay, M. Giacomin, and L. van der Torre, editors, *Handbook of Formal Argumentation*, pages 159–236. College Publications, 2018.

[3] R. Baumann. Normal and strong expansion equivalence for argumentation frameworks. *Artificial Intelligence*, 193:18–44, 2012.

[4] R. Baumann and G. Brewka. The equivalence zoo for dung-style semantics. *Journal of Logic and Computation*, 28(3):477–498, 2018.

[5] R. Baumann, W. Dvořák, T. Linsbichler, and S. Woltran. A general notion of equivalence for abstract argumentation. *Artificial Intelligence*, 275:379–410, 2019.

[6] R. Baumann, G. Brewka, and M. Ulbricht. Revisiting the foundations of abstract argumentation–semantics based on weak admissibility and weak defense. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 2742–2749, 2020.

[7] L. Bengel and M. Thimm. Serialisable semantics for abstract argumentation. In *Computational Models of Argument: Proceedings of COMMA 2022*, pages 80–91. IOS Press, 2022.

[8] L. Blümel and M. Thimm. A ranking semantics for abstract argumentation based on serialisability. *Computational Models of Argument: Proceedings of COMMA 2022*, 353:104, 2022.

[9] M. Caminada. Argumentation semantics as formal discussion. In P. Baroni, D. Gabbay, M. Giacomin, and L. van der Torre, editors, *Handbook of Formal Argumentation*, pages 487–518. College Publications, 2018.

[10] K. Čyras, A. Rago, E. Albini, P. Baroni, and F. Toni. Argumentative xai: a survey. *arXiv preprint arXiv:2105.11266*, 2021.

[11] P. M. Dung. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-Person Games. *Artificial Intelligence*, 77(2):321–358, 1995.

[12] P. E. Dunne and T. J. M. Bench-Capon. Coherence in finite argument systems. *Artificial Intelligence*, 141(1/2):187–203, 2002. doi: 10.1016/S0004-3702(02)00261-8. URL https://doi.org/10.1016/S0004-3702(02)00261-8.

[13] J. Hage. Dialectical models in artificial intelligence and law. *Artificial Intelligence and Law*, 8(2-3):137–172, 2000.

[14] B. Liao and L. van der Torre. Defense semantics of argumentation: encoding reasons for accepting arguments. *arXiv preprint arXiv:1705.00303*, 2017.

[15] B. Moulin, H. Irandoust, M. Bélanger, and G. Desbordes. Explanation and argumentation capabilities: Towards the creation of more persuasive agents. *Artificial Intelligence Review*, 17(3):169–222, 2002.

[16] E. Oikarinen and S. Woltran. Characterizing strong equivalence for argumentation frameworks. *Artificial intelligence*, 175(14-15):1985–2009, 2011.

[17] C. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.

[18] N. Rescher. *Dialectics: A controversy-oriented approach to the theory of knowledge*. Suny Press, 1977.

[19] M. Thimm. Revisiting initial sets in abstract argumentation. *Argument & Computation*, 13(3):325–360, 2022.

[20] A. Vassiliades, N. Bassiliades, and T. Patkos. Argumentation and explainable artificial intelligence: a survey. *The Knowledge Engineering Review*, 36:e5, 2021.

[21] B. Verheij. Two approaches to dialectical argumentation: admissible sets and argumentation stages. *Proc. NAIC*, 96:357–368, 1996.

[22] Y. Xu and C. Cayrol. Initial sets in abstract argumentation frameworks. In *Proceedings of the 1st Chinese Conference on Logic and Argumentation (CLAR'16)*, volume 1811, pages 72–85, 2016.