Consequence Operators of Characterization Logics – The Case of Abstract Argumentation

Ringo Baumann[1,2][0000−0003−2534−3299] and Hannes Strass[3][0000−0001−6180−6452]

[1] Leipzig University, Germany
[2] Center for Scalable Data Analytics and Artificial Intelligence (ScaDS.AI) Dresden/Leipzig, Germany
baumann@informatik.uni-leipzig.de
[3] TU Dresden, Germany
hannes.strass@tu-dresden.de

Abstract The analysis of properties of consequence operators has been a very active field in the formative years of non-monotonic reasoning. One possible approach to do this is to start with a model-theoretic semantics and then to study the logical consequence relation induced by that semantics. In this paper we follow that approach and analyse resulting consequence operators of so-called characterization logics. Roughly speaking, a characterization logic characterizes, via its own notion of ordinary equivalence, another logic's notion of *strong equivalence*. For example, the logic of here and there is a characterization logic for answer set programs, because strong equivalence of the latter is characterized by ordinary equivalence of the former.
In previous work, we showed that the consideration of finite knowledge bases only – a common assumption in the field of knowledge representation – guarantees the existence (and uniqueness) of characterization logics. In this paper, we apply this existence result to the field of abstract argumentation. We show that the associated consequence operator outputs a so-called reverse kernel, a useful construct that received comparably little attention in the literature so far. As an aside, we clarify that for several well-known logics, their canonical characterization consequence operators are well-behaved.

Keywords: Characterization Logic · Strong Equivalence · Abstract Argumentation

1 Introduction

After many decades, the field of knowledge representation finds itself in a comfortable situation, as it is equipped with a variety of logical formalisms. Selecting the most adequate one for a specific purpose is one of the decisive questions before starting to apply a formalism. In order to make an informed choice, it is important to be aware of intrinsic properties of the available formalisms. For

instance, one far-reaching decision is whether it is possible to withdraw former conclusions (cf. [12] for an excellent overview). This distinctive feature divides the formalisms into so-called monotonic and non-monotonic ones. Another important feature is the availability of semantically neutral replaceability, via a notion of *strong equivalence*, which when present allows one to simplify parts of a theory without looking at the rest. In a series of interesting developments, researchers have succeeded in precisely characterizing strong equivalence for several formalisms, among them logic programs [14,18], causal theories [19], default logic [18] and some classes of non-monotonic logics in general [16,17].

In several cases, to characterize strong equivalence in formalism $\mathcal{L}$, we can use ordinary equivalence in formalism $\mathcal{L}'$: for example, strong equivalence in normal logic programs under stable models can be characterized by the standard semantics of the logic of here and there [14]. We recently studied this phenomenon in a general setting and coined the term *characterization logics* [4,5]. One main result was that characterization logics exist in general if we restrict ourselves to finite knowledge bases only – a common assumption in applications of knowledge representation. However, although this represents a quite remarkable theoretical result, the canonical semantics of these logics are somewhat unhandy and less telling as they return infinite unions of equivalence classes. In this paper we instead focus on the resulting consequence operators and show that they are both enlightening and surprising. Firstly, we show that for a whole family of logics (including well-known logics like classical propositional logic and first-order logic) that their characterization logics return the same consequences as the characterized logic. Secondly, and as our main contribution, we study a specific non-monotonic formalism, namely Dung's argumentation frameworks [9]. It turns out that the associated consequence operator outputs the *reverse kernel*, an object that has received comparably little attention in the literature so far.

2 Preliminaries

2.1 Semantics, Consequences and Characterization Logics

A model-theoretic semantics for a language $\mathcal{L}$ uses a set $\mathcal{I}$ of interpretations and a model function $\sigma\colon 2^{\mathcal{L}} \to 2^{\mathcal{I}}$. The intuition is that σ assigns to each $T \subseteq \mathcal{L}$, a so-called *$\mathcal{L}$-theory*, a set $\sigma(T)$ of *models* of T. (Note that a theory is merely a set of formulas, there are no further assumptions on them.) A triple $(\mathcal{L}, \mathcal{I}, \sigma)$ is called a *logic*. As the set $\mathcal{I}$ of interpretations is implicit in σ, in the remainder of the paper we will typically disregard $\mathcal{I}$ and denote logics by pairs $(\mathcal{L}, \sigma)$. We now formally introduce two well-known notions of equvialence.

Definition 1. *Let $(\mathcal{L}, \sigma)$ be a logic and $T_1, T_2 \subseteq \mathcal{L}$ be theories. T_1 and T_2 are* ordinarily equivalent *if $\sigma(T_1) = \sigma(T_2)$. If even $\sigma(T_1 \cup U) = \sigma(T_2 \cup U)$ for all theories $U \subseteq \mathcal{L}$, we call T_1 and T_2* strongly equivalent*.*

We use $[T]^{\sigma}_{s} = \{S \subseteq \mathcal{L} \mid S \text{ is strongly equivalent to } T\}$ to denote the associated equivalence class. Obviously, strong equivalence implies ordinary equivalence but not necessarily vice versa [14,18]. Note that strong equivalence enables

to replace a part of a theory without affecting the semantics, in arbitrary contexts. If in a given logic $(\mathcal{L}, \sigma)$ both notions coincide we say that $(\mathcal{L}, \sigma)$ possesses the *replacement property*. Thus the replacement property guarantees that already ordinary equivalence allows for semantically neutral replacements.

Definition 2. *Let $(\mathcal{L}, \sigma)$ be a logic. The model function (the logic) possesses the* intersection property *if for each $\mathcal{L}$-theory T, we have $\sigma(T) = \bigcap_{\varphi \in T} \sigma(\{\varphi\})$.*

In previous work, we have shown that the intersection property implies the replacement property [5, Proposition 3]. For a given logic we also defined an associated consequence operator Cn^σ (intuitively assigning to a given $\mathcal{L}$-theory T the theory $Cn^\sigma(T)$ of its semantical consequences) in a canonical way.

Definition 3. *Let $(\mathcal{L}, \sigma)$ be a logic. We call Cn^σ the* canonical consequence operator of σ *where $Cn^\sigma \colon 2^\mathcal{L} \to 2^\mathcal{L}$ with $T \mapsto \bigcup_{S \subseteq \mathcal{L}, \sigma(T) \subseteq \sigma(S)} S$.*

If the logic $(\mathcal{L}, \sigma)$ possesses the intersection property, then the canonical consequence operator is a closure operator [5, Proposition 6]. Thus in addition to being increasing ($T \subseteq Cn^\sigma(T)$) and idempotent ($Cn^\sigma(Cn^\sigma(T)) \subseteq Cn^\sigma(T)$), Cn^σ is in particular monotone ($T_1 \subseteq T_2$ implies $Cn^\sigma(T_1) \subseteq Cn^\sigma(T_2)$), that is, the intersection property is sufficient for the logic being monotone.

Finally, we introduce the central definition of a characterization logic $(\mathcal{L}, \sigma')$ of $(\mathcal{L}, \sigma)$. Such a logic possesses two properties: Firstly, it characterizes strong equivalence in $(\mathcal{L}, \sigma)$ via its own ordinary equivalence and secondly, $(\mathcal{L}, \sigma')$ has the intersection property (thus is a monotone logic).

Definition 4. *Logic $(\mathcal{L}, \sigma')$ is a* characterization logic *for logic $(\mathcal{L}, \sigma)$ if:*

1. $\forall T_1, T_2 \in 2^\mathcal{L} \colon \sigma'(T_1) = \sigma'(T_2) \iff [T_1]^\sigma_s = [T_2]^\sigma_s$, *and* *(characterization)*
2. $\forall \mathcal{T} \subseteq 2^\mathcal{L} \colon \sigma'\big(\bigcup_{T \in \mathcal{T}} T\big) = \bigcap_{T \in \mathcal{T}} \sigma'(T)$.[4] *(intersection)*

Note that the second property enforces that characterization logics are monotonic. However, the logic being characterized need not be monotonic. For example, it is already known that strong equivalence in normal logic programs under stable models can be characterized by the standard semantics of the logic of here and there [14]. This means, a highly non-monotonic formalism is characterized by a monotonic one. Moreover, a logic can be its own characterization logic. For instance, classical propositional logic is its own characterization logic, as ordinary and strong equivalence coincide and intersection holds by definition.

2.2 Argumentation Frameworks and Semantics

An *argumentation framework* (AF) is a pair $F = (A, R)$ where A (the set of arguments) is a subset of a fixed infinite background set $\mathcal{U}$. Moreover, R (the set

[4] We mention that the second item is equivalent to the more common version presented in Definition 2 [5, Proposition 9].

of attacks) is a subset of $A \times A$ [9]. The set of all (finite) AFs is denoted by $\mathcal{F}$ ($\mathcal{F}_{\text{fin}}$). An *extension-based semantics* $\rho\colon \mathcal{F} \to 2^{2^{\mathcal{U}}}$ assigns to each AF $F = (A, R)$ a set $\rho(F) \subseteq 2^A$ of sets of arguments. Each $E \in \rho(F)$, a so-called ρ-extension, is considered to be acceptable (according to ρ) with respect to F.

Some prominent semantics already introduced by Dung [9] are *stable*, *admissible*, *complete*, *preferred* and *grounded* semantics (abbreviated *stb*, *ad*, *co*, *gr*, *pr*). We refrain from presenting the concrete definitions as we are concentrating on consequence operators of characterization logics. However, for readers far from the field we recommend an overview [1].

Similar to Definition 1 we introduce *strong equivalence* for two AFs F and G (abbreviated $F \equiv_s^\rho G$) as: for each AF H, $\rho(F \sqcup H) = \rho(G \sqcup H)$. The union $(A, R) \sqcup (B, S)$ is defined as $(A \cup B, R \cup S)$. Please note that we have to use "$\sqcup$" instead of "$\cup$" as we are dealing with directed graphs. The embedding of AFs in the general setup of $\mathcal{L}$-theories will be considered in detail in Section 4.2.

3 Propositional Logic, FOL, and Friends – Logics with Intersection Property

We start with formalisms that possess the intersection property. Representatives are well-known logics like propositional logic or first-order logic.[5] Firstly, note that for these logics, the intersection property holds by definition: Indeed, their semantics σ (usually denoted as *Mod*) are firstly defined for single formulas φ and then generalized to theories T by *setting* $\sigma(T) = \bigcap_{\varphi \in T} \sigma(\{\varphi\})$. Furthermore, the standard approach to define logical consequences $Cn(\cdot)$ based on model theory, namely setting $Cn(T) = \{\varphi \in \mathcal{L} \mid \sigma(T) \subseteq \sigma(\{\varphi\})\}$, coincides with Definition 3's notion of canonical consequence: If $\varphi \in Cn(T)$, then clearly $\varphi \in Cn^\sigma(T)$ via $S = \{\varphi\}$; conversely, if $\varphi \in S \subseteq \mathcal{L}$ with $\sigma(T) \subseteq \sigma(S)$, then $\sigma(S) = \sigma(S \cup \{\varphi\}) = \sigma(S) \cap \sigma(\{\varphi\})$ whence $\sigma(S) \subseteq \sigma(\{\varphi\})$ and by transitivity $\sigma(T) \subseteq \sigma(\{\varphi\})$, that is, $\varphi \in Cn(T)$. Thus for classical logic, the canonical consequence operator of Definition 3 exactly embodies the standard notion of logical consequence.

How does a consequence operator of a characterization logic look like in general? We already know that any characterization logic is a sublogic of the initial one [5], that is, consequences in the characterizing logic are also consequences in the characterized logic. In case of logics with intersection property we can show even more. Since such formalisms have the replacement property, we may consider the formalism itself as its own characterization logic. The following result shows that the associated consequence operator of *any other* characterization logic coincides with the already known consequence operator of the initial logic. Thus, there is no space for surprising consequences, a reassuring result.

Proposition 1. *Let $(\mathcal{L}, \sigma)$ be a logic that has the intersection property. For any characterization logic $(\mathcal{L}, \sigma')$ of $(\mathcal{L}, \sigma)$ we have: $Cn^\sigma = Cn^{\sigma'}$.*

[5] Although these logics are monotonic, it would be a misconception to claim that *all* monotonic logics have the replacement property [5, Exm. 5], in other words, to say that being non-monotonic is the *reasong* for not having the replacement property.

Proof. Let T be an arbitrary $\mathcal{L}$-theory. We show $Cn^{\sigma}(T) = Cn^{\sigma'}(T)$.

($\subseteq$) Let $\varphi \in Cn^{\sigma}(T)$. By definition of the canonical consequence there is a set S, s.t. $\varphi \in S$ and $\sigma(T) \subseteq \sigma(S)$. Consequently, $\varphi \in S \cup T$ and by intersection property we obtain: $\sigma(S \cup T) = \sigma(S) \cap \sigma(T) = \sigma(T)$. Since intersection guarantees replacement we have $[S \cup T]^{\sigma}_s = [T]^{\sigma}_s$. As $(\mathcal{L}, \sigma')$ is assumed to be a characterization logic of $(\mathcal{L}, \sigma)$ we derive $\sigma'(S \cup T) = \sigma'(T)$. This means, $Cn^{\sigma'}(S \cup T) = Cn^{\sigma'}(T)$. Since characterization logics possesses the intersection property we derive that $Cn^{\sigma'}$ is a closure operator [5, Proposition 6]. Thus, by inclusion we get $S \cup T \subseteq Cn^{\sigma'}(S \cup T) = Cn^{\sigma'}(T)$. Finally, using $\varphi \in S \cup T$ yields $\varphi \in Cn^{\sigma'}(T)$

($\supseteq$) The sublogic property, i.e. $Cn^{\sigma'}(T) \subseteq Cn^{\sigma}(T)$ for any T, holds without any restriction [5, Proposition 11, Item 1]. □

4 Logics without Intersection Property

In the section before we have clarified the case of logics possessing the intersection property. In fact, there are no deviations from the initial consequences. Now let us turn to logics without intersection property.

4.1 What is already known?

At the very beginning, we must clarify whether such logics possess characterization logics at all. From our previous results [4], we know that, firstly, not every formalism (that can be cast in our abstract, model-theoretic framework [5]) has one, but secondly, there are restrictions that guarantee the existence of characterization logics. One of those restrictions is the consideration of finite knowledge bases only. Note that this restriction, especially in the field of knowledge representation, is indeed not overly limiting, as finite knowledge bases are the most relevant for practical purposes.

The next definition translates this assumption into our setting. For a given logic we call the restriction to finite knowledge bases the *finite-theory version*.

Definition 5. *Let $(\mathcal{L}, \sigma)$ be a logic. The finite-theory version $(\mathcal{L}, \sigma_{\text{fin}})$ of $(\mathcal{L}, \sigma)$ is defined by the semantics $\sigma_{\text{fin}} \colon \left(2^{\mathcal{L}}\right)_{\text{fin}} \to \sigma\left(2^{\mathcal{L}}\right)$ with $\sigma_{\text{fin}}(T) = \sigma(T)$ where $\left(2^{\mathcal{L}}\right)_{\text{fin}} = \{T \in 2^{\mathcal{L}} \mid T \text{ is finite}\}$.*

For finite-theory restrictions of logics, we adequately relax our requirements on characterization logics (refer to Definition 4). Indeed, only finite theories are considered and arbitrary unions are disallowed as they may lead to infinite sets.

Definition 6. *Let $(\mathcal{L}, \sigma_{\text{fin}})$ be the finite-theory version of $(\mathcal{L}, \sigma)$. We say that $(\mathcal{L}, \sigma'_{\text{fin}})$ is a finite-theory characterization logic for $(\mathcal{L}, \sigma)$ if and only if:*

1. $\forall T_1, T_2 \in \left(2^{\mathcal{L}}\right)_{\text{fin}} : \sigma'_{\text{fin}}(T_1) = \sigma'_{\text{fin}}(T_2)$ *iff* $[T_1]^{\sigma_{\text{fin}}}_s = [T_2]^{\sigma_{\text{fin}}}_s$, *(characterization)*

2. $\forall T_1, T_2 \in \left(2^{\mathcal{L}}\right)_{\text{fin}} : \sigma'_{\text{fin}}(T_1 \cup T_2) = \sigma'_{\text{fin}}(T_1) \cap \sigma'_{\text{fin}}(T_2)$. *(intersection)*

Now, we recall the central theorem stating that any logic possesses a finite-theory characterization logic [4, Theorem 9]. We also put the associated consequence operator (cf. Definition 3) in the theorem as the analysis of this operator is the main aim of this paper.

Theorem 1. *Given a logic $(\mathcal{L}, \sigma)$, a finite-theory characterization logic for it is given by $(\mathcal{L}, \sigma'_{\text{fin}})$ with model function $\sigma'_{\text{fin}} \colon \left(2^{\mathcal{L}}\right)_{\text{fin}} \to 2^{2^{\mathcal{L}}}$ and consequence operator $Cn^{\sigma'_{\text{fin}}} \colon \left(2^{\mathcal{L}}\right)_{\text{fin}} \to \left(2^{\mathcal{L}}\right)_{\text{fin}}$ are given by, respectively,*

$$T \mapsto \bigcup_{\substack{S \in (2^{\mathcal{L}})_{\text{fin}}, \\ T \subseteq S}} [S]_s^{\sigma_{\text{fin}}} \qquad \text{and} \qquad T \mapsto \bigcup_{\substack{S \in (2^{\mathcal{L}})_{\text{fin}}, \\ \sigma'_{\text{fin}}(T) \subseteq \sigma'_{\text{fin}}(S)}} S$$

The specifics of these consequence operators heavily rely on the underlying logic $(\mathcal{L}, \sigma_{\text{fin}})$. In this very first paper regarding consequences we consider a well-known non-monotonic formalism and leave other logics for future work. We will see that the analysis requires a lot of technical details. However, in the end the great effort is worth it and rewards us with an unexpected outcome.

4.2 The Non-monotonic Theory of Abstract Argumentation

As discussed in Section 2.2, an AF F is a directed graph. However, in order to apply the definitions and results of the preceding section to argumentation theory we have to have $\mathcal{L}$-theories which correspond to AFs, s.t. the standard set union $\cup$ of such theories corresponds to $\sqcup$ on the AF level. Moreover, it has to be ensured that the semantics of theories correspond to the argumentation semantics of the associated AFs. This can be done in the following way.

Definition 7. *Given an argumentation semantics $\rho \colon \mathcal{F} \to 2^{2^{\mathcal{U}}}$, we define the logic $(\mathcal{L}_{\mathcal{F}}, \mathcal{I}_{\mathcal{F}}, \sigma_\rho)$ where $\mathcal{L}_{\mathcal{F}} = \{(\{a\}, \emptyset), (\{a, b\}, \{(a, b)\}) \mid a, b \in \mathcal{U}\}$, $\mathcal{I}_{\mathcal{F}} = 2^{\mathcal{U}}$ and $\sigma_\rho \colon 2^{\mathcal{L}_{\mathcal{F}}} \to 2^{\mathcal{I}_{\mathcal{F}}}$ with $\sigma_\rho(T) = \rho(\bigsqcup T)$.*

For a given $\mathcal{L}_{\mathcal{F}}$-theory T we call $AF(T) = \bigsqcup T$ the *associated AF*. We extend this definition to a set $\mathcal{T}$ of $\mathcal{L}_{\mathcal{F}}$-theories via $AF(\mathcal{T}) = \{AF(T) \mid T \in \mathcal{T}\}$. Moreover, the *canonical theory* T of a given AF $F = (A, R)$ is defined as $C(F) = \{(\{a\}, \emptyset) \mid a \in A\} \cup \{(\{a, b\}, \{(a, b)\}) \mid (a, b) \in R\}$. It follows that for single AFs F we have $AF(C(F)) = F$.

In the following we state three important properties showing that the concrete representation does not matter and that the argumentation semantics ρ and the induced one σ_ρ are compatible as desired (cf. [5, Proposition 23,Theorem 24]).

Proposition 2. *Given a logic $(\mathcal{L}_{\mathcal{F}}, \mathcal{I}_{\mathcal{F}}, \sigma_\rho)$ and $\mathcal{L}_{\mathcal{F}}$-theories S and T, we have*

1. $AF(S \cup T) = AF(S) \sqcup AF(T)$,

2. $\sigma_\rho(S \cup T) = \rho(AF(S) \sqcup AF(T))$, *and*
3. $AF(S) \equiv_s^\rho AF(T)$ *iff* $[S]_s^{\sigma_\rho} = [T]_s^{\sigma_\rho}$.

Now, we are ready to consider the associated finite-theory charaterization logic (Theorem 1) as well as the resulting consequence operator.

Corollary 1. *Given an argumentation semantics ρ with induced logic $(\mathcal{L}_\mathcal{F}, \mathcal{I}_\mathcal{F}, \sigma_\rho)$, $\left(\mathcal{L}_\mathcal{F}, (\sigma_\rho)'_{\text{fin}}\right)$ is a finite-theory characterization logic of $(\mathcal{L}_\mathcal{F}, \mathcal{I}_\mathcal{F}, \sigma_\rho)$, where the characterizing model function $(\sigma_\rho)'_{\text{fin}} : \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}} \to 2^{2^{\mathcal{L}_\mathcal{F}}}$ and consequence operator $Cn^{(\sigma_\rho)'_{\text{fin}}} : \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}} \to \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}}$ are given by, respectively,*

$$T \mapsto \bigcup_{\substack{S \in \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}}, \\ T \subseteq S}} [S]_s^{(\sigma_\rho)_{\text{fin}}} \qquad \textit{and} \qquad T \mapsto \bigcup_{\substack{S \in \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}}, \\ (\sigma_\rho)'_{\text{fin}}(T) \subseteq (\sigma_\rho)'_{\text{fin}}(S)}} S$$

How can we interpret these finite-theory characterization logics in terms of argumentation theory? In other words, what is the corresponding consequence operator on the level of pure AFs (instead of theories associated with AFs)? This question will be tackled in the remainder of this paper.

4.3 Interpreting the Consequence Operator on AF-level

Now we translate the consequence operator $Cn^{(\sigma_\rho)'_{\text{fin}}}$ to realm of abstract argumentation frameworks. This means, instead of mapping an $\mathcal{L}_\mathcal{F}$-theory T to another $\mathcal{L}_\mathcal{F}$-theory S we do the following: (1) We start with an AF F, (2) Translate it to the level of $\mathcal{L}_\mathcal{F}$-theories via the canonical theory $C(F)$, (3) Apply then the consequence operator $Cn^{(\sigma_\rho)'_{\text{fin}}}$ from Corollary 1, and (4) Retranslate the obtained $\mathcal{L}_\mathcal{F}$-theory $Cn^{(\sigma_\rho)'_{\text{fin}}}(C(F))$ via the associated AF.

Definition 8. *Given an argumentation semantics $\rho : \mathcal{F} \to 2^{2^\mathcal{U}}$. We define the translated consequence operator $Cn^{\rho'_{\text{fin}}}$ as*

$$Cn^{\rho'_{\text{fin}}} : \mathcal{F}_{\text{fin}} \to \mathcal{F}_{\text{fin}}, \qquad F \mapsto AF\left(Cn^{(\sigma_\rho)'_{\text{fin}}}(C(F))\right)$$

In the following we will prove several technical results needed to show that the output $AF\left(Cn^{(\sigma_\rho)'_{\text{fin}}}(C(F))\right)$ can be greatly simplified.

Proposition 3. *Given two $\mathcal{L}_\mathcal{F}$-theories S and T. We have:*

$$(\sigma_\rho)'_{\text{fin}}(T) \subseteq (\sigma_\rho)'_{\text{fin}}(S) \quad \textit{iff} \quad AF\left((\sigma_\rho)'_{\text{fin}}(T)\right) \subseteq AF\left((\sigma_\rho)'_{\text{fin}}(S)\right)$$

Proof. Let S and T be $\mathcal{L}_\mathcal{F}$-theories.

($\Rightarrow$) Remember that $(\sigma_\rho)'_{\text{fin}} : \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}} \to 2^{2^{\mathcal{L}_\mathcal{F}}}$. This means, a $\mathcal{L}_\mathcal{F}$-theory is mapped to a set of $\mathcal{L}_\mathcal{F}$-theories. Assume $U \in AF\left((\sigma_\rho)'_{\text{fin}}(T)\right)$. As AF is applied pointwise there exists a $T' \in (\sigma_\rho)'_{\text{fin}}(T)$ s.t. $U = AF(T')$. Since $(\sigma_\rho)'_{\text{fin}}(T) \subseteq (\sigma_\rho)'_{\text{fin}}(S)$ is assumed we deduce $U \in AF\left((\sigma_\rho)'_{\text{fin}}(S)\right)$.

($\Leftarrow$) We show the contrapositive. So let $(\sigma_\rho)'_{\text{fin}}(T) \not\subseteq (\sigma_\rho)'_{\text{fin}}(S)$. Hence, there is a $\mathcal{L}_\mathcal{F}$-theory $T' \in (\sigma_\rho)'_{\text{fin}}(T) \setminus (\sigma_\rho)'_{\text{fin}}(S)$. Consequently, there has to be a $\mathcal{L}_\mathcal{F}$-theory T'' with $T \subseteq T''$ and $T' \in [T'']_s^{(\sigma_\rho)_{\text{fin}}}$. Since we are considering equivalence classes we even have $[T'']_s^{(\sigma_\rho)_{\text{fin}}} \cap (\sigma_\rho)'_{\text{fin}}(S) = \emptyset$. Note that $\mathcal{L}_\mathcal{F}$-theories resulting in identical frameworks are in the same strong equivalence class (Proposition 2, Item 3). This means, for each $\mathcal{L}_\mathcal{F}$-theory S' with $AF(T') = AF(S')$ we have $S' \in [T'']_s^{(\sigma_\rho)_{\text{fin}}}$. Consequently, $AF(T') \notin AF\big((\sigma_\rho)'_{\text{fin}}(S)\big)$. Hence, $AF\big((\sigma_\rho)'_{\text{fin}}(T)\big) \not\subseteq AF\big((\sigma_\rho)'_{\text{fin}}(S)\big)$. □

In order to proceed we retranslate the semantics $(\sigma_\rho)'_{\text{fin}}$ to AF-level. This can be done in similar way to Definition 8 and was already considered in previous work [5, Definition 15]. We name the semantics ρ'_{fin} as it perfectly fits with the translated consequence operator $Cn^{\rho'_{\text{fin}}}$ (Definition 8).

Definition 9. *Given an argumentation semantics $\rho\colon \mathcal{F} \to 2^{2^\mathcal{U}}$, its* translated characterization semantics *is $\rho'_{\text{fin}}\colon \mathcal{F}_{\text{fin}} \to 2^\mathcal{F}$ with $F \mapsto AF\big((\sigma_\rho)'_{\text{fin}}(C(F))\big)$.*

It was shown that crucial properties of the finite-theory characterization logic $\big(\mathcal{L}_\mathcal{F}, (\sigma_\rho)'_{\text{fin}}\big)$ transfer to $(\mathcal{F}, \rho'_{\text{fin}})$ [5, Proposition 26]. In particular, the characterization property is fulfilled, i.e. for two AFs F and G: $\rho'_{\text{fin}}(F) = \rho'_{\text{fin}}(G)$ if and only if $[F]_s^\rho = [G]_s^\rho$. We now show that $Cn^{\rho'_{\text{fin}}}$ can be alternatively expressed with the help of the translated characterization semantics ρ'_{fin}, thereby justifying the naming convention (confer Definition 3).

Proposition 4. *Consider an argumentation semantics $\rho\colon \mathcal{F} \to 2^{2^\mathcal{U}}$. For any AF $F \in \mathcal{F}_{\text{fin}}$ we have:*

$$Cn^{\rho'_{\text{fin}}}(F) = \bigsqcup_{\substack{G \in \mathcal{F}_{\text{fin}},\\ \rho'_{\text{fin}}(F) \subseteq \rho'_{\text{fin}}(G)}} G$$

Proof. Given an AF $F \in \mathcal{F}_{\text{fin}}$, we have the following equalities:

$$\begin{aligned}
Cn^{\rho'_{\text{fin}}}(F) &= AF\left(Cn^{(\sigma_\rho)'_{\text{fin}}}(C(F))\right) && \text{(Definition 8)}\\
&= AF\left(\bigcup_{\substack{S \in (2^{\mathcal{L}_\mathcal{F}})_{\text{fin}},\\ (\sigma_\rho)'_{\text{fin}}(C(F)) \subseteq (\sigma_\rho)'_{\text{fin}}(S)}} S\right) && \text{(Definition } Cn^{(\sigma_\rho)'_{\text{fin}}}\text{, Corollary 1)}\\
&= \bigsqcup_{\substack{S \in (2^{\mathcal{L}_\mathcal{F}})_{\text{fin}},\\ (\sigma_\rho)'_{\text{fin}}(C(F)) \subseteq (\sigma_\rho)'_{\text{fin}}(S)}} AF(S) && \text{(Proposition 2, Item 1)}\\
&= \bigsqcup_{\substack{S \in (2^{\mathcal{L}_\mathcal{F}})_{\text{fin}},\\ AF((\sigma_\rho)'_{\text{fin}}(C(F))) \subseteq AF((\sigma_\rho)'_{\text{fin}}(S))}} AF(S) && \text{(Proposition 3)}
\end{aligned}$$

$$= \bigsqcup_{\substack{S \in \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}}, \\ \rho'_{\text{fin}} \subseteq AF\left((\sigma_\rho)'_{\text{fin}}(S)\right)}} AF(S) \qquad \textit{(Definition 9)}$$

$$= \bigsqcup_{\substack{S \in \left(2^{\mathcal{L}_\mathcal{F}}\right)_{\text{fin}}, \\ \rho'_{\text{fin}} \subseteq AF\left((\sigma_\rho)'_{\text{fin}}(C(AF(S)))\right)}} AF(S) \qquad \textit{(}(\sigma_\rho)'_{\text{fin}}(C(AF(S))) = (\sigma_\rho)'_{\text{fin}}(S)\textit{)}$$

$$= \bigsqcup_{\substack{G \in \mathcal{F}_{\text{fin}}, \\ \rho'_{\text{fin}} \subseteq AF\left((\sigma_\rho)'_{\text{fin}}(C(G))\right)}} G \qquad \textit{(set } AF(S) = G \textit{)}$$

$$= \bigsqcup_{\substack{G \in \mathcal{F}_{\text{fin}}, \\ \rho'_{\text{fin}}(F) \subseteq \rho'_{\text{fin}}(G)}} G \qquad \textit{(Definition 9)} \quad \square$$

4.4 Characterizing Strong Equivalence: (Reverse) Kernels

The notion of strong equivalence for AFs was firstly tackled by Oikarinen and Woltran [15]. They provided a series of characterization theorems for deciding strong equivalence. The surprising result was that being strongly equivalent can be decided syntactically, which is a distinctive characteristic in the realm of non-monontonic logics [14,18,19]. More precisely, they introduced the notion of a *kernel* of an AF F which is simply a subgraph of F where certain *redundant* attacks are deleted. It was shown that syntactical identity of these subgraphs characterizes strong equivalence w.r.t. the considered semantics.

Later on, it was recognized that the presented kernel definitions are not uniquely determined [2]. In more detail, the classical kernels [15] represents the $\sqsubseteq$-least element in the associated strong equivalence class. Remember that $(A, R) \sqsubseteq (B, S)$ if both, $A \subseteq B$ and $R \subseteq S$. Alternatively, one may consider $\sqsubseteq$-greatest elements (in case of existence). This means, such a so-called *reverse kernel* adds all redundant attacks to an AF instead of removing them. As an aside, such alternative kernels have already shown to be useful in the context of orderings and boundaries [3] and will be key for the aim of this paper.

In the following we restrict ourselves to the traditional Dung semantics. However, further characterization results exist and can be used to define reverse kernels [13,6].

Definition 10. *Let $\rho \in \{stb, ad, co, gr, pr\}$. For any AF $F = (A, R)$ we define the ρ-reverse kernel $F^{k^+(\sigma)} = \left(A, R^{k^+(\sigma)}\right)$ as:*

1. $R^{k^+(stb)} = R \cup \{(a, b) \mid a \neq b, (a, a) \in R\}$,
2. $R^{k^+(ad)} = R \cup \{(a, b) \mid a \neq b, (a, a) \in R, \{(b, a), (b, b)\} \cap R \neq \emptyset\}$,
3. $R^{k^+(co)} = R \cup \{(a, b) \mid a \neq b, (a, a), (b, b) \in R\}$,
4. $R^{k^+(gr)} = R \cup \{(a, b) \mid a \neq b, (b, b) \in R, \{(a, a), (b, a)\} \cap R \neq \emptyset\}$,
5. $R^{k^+(pr)} = R \cup \{(a, b) \mid a \neq b, (a, a) \in R, \{(b, a), (b, b)\} \cap R \neq \emptyset\}$.

At first we state that these alternative kernels are characterizing too.

Proposition 5. *Given two AFs F and G and $\rho \in \{stb, ad, co, gr, pr\}$. We have:*

$$[F]_s^\rho = [G]_s^\rho \text{ iff } F^{k^+(\rho)} = G^{k^+(\rho)}$$

Proof (Sketch). The classical kernels $k(\rho)$ are characterizing, i.e. $[F]_s^\rho = [G]_s^\rho$ iff $F^{k(\rho)} = G^{k(\rho)}$ [15]. These kernels simply delete the redundant attacks, e.g. for $\rho = stb$ we have $R^{k(stb)} = R \setminus \{(a,b) \mid a \neq b, (a,a) \in R\}$. Now, it suffices to see that for each AF F: 1. $\left(F^{k^+(\rho)}\right)^{k(\rho)} = F^{k(\rho)}$ and 2. $\left(F^{k(\rho)}\right)^{k^+(\rho)} = F^{k^+(\rho)}$.

Now we recall an already known alternative formulation for the translated characterization semantics ρ'_{fin} [5, Proposition 27]. This formulation will lift our consequence operator to the level of equivalence classes and finally to reverse kernels via Proposition 5.

Proposition 6. *Let $\rho\colon \mathcal{F} \to 2^{2^{\mathcal{U}}}$ be a semantics and ρ'_{fin} as defined in Definition 9. We have:*

$$\rho'_{\text{fin}}(F) = \bigcup_{H \in \mathcal{F},\ F \sqsubseteq H} [H]_s^\rho$$

Finally, we present the main theorem of the paper stating that the induced consequence operator maps an AF F to its associated reverse kernel. Note that this result stems from a completely abstract view on logics and model theory, respectively [5]. We just applied the conditional existence result to abstract argumentation theory. It will be interesting to see what can be achieved for other well-known non-monotonic logics.

Theorem 2. *Given a semantics $\rho\colon \mathcal{F} \to 2^{2^{\mathcal{U}}}$, for any AF $F \in \mathcal{F}_{\text{fin}}$ we have:*

$$Cn^{\rho'_{\text{fin}}}(F) = F^{k^+(\rho)}.$$

Proof. At first we reformulate $Cn^{\rho'_{\text{fin}}}(F)$ as given in Proposition 4. More precisely, we show:

$$Cn^{\rho'_{\text{fin}}}(F) = \bigsqcup \left\{ G \in \mathcal{F}_{\text{fin}} \mid \bigcup_{\substack{H \in \mathcal{F}, \\ F^{k^+(\rho)} \sqsubseteq H}} [H]_s^\rho \subseteq \bigcup_{\substack{H \in \mathcal{F}, \\ G \sqsubseteq H}} [H]_s^\rho \right\} \qquad (1)$$

This can be seen as follows: First, remember that ρ'_{fin} is characterizing, i.e. $\rho'_{\text{fin}}(F) = \rho'_{\text{fin}}(G)$ if and only if $[F]_s^\rho = [G]_s^\rho$. Moreover, since obviously $[F]_s^\rho = \left[F^{k^+(\rho)}\right]_s^\rho$ we obtain $\rho'_{\text{fin}}(F) = \rho'_{\text{fin}}\left(F^{k^+(\rho)}\right)$. Hence, we get

$$Cn^{\rho'_{\text{fin}}}(F) = \bigsqcup_{\substack{G \in \mathcal{F}_{\text{fin}}, \\ \rho'_{\text{fin}}(F) \subseteq \rho'_{\text{fin}}(G)}} G = \bigsqcup_{\substack{G \in \mathcal{F}_{\text{fin}}, \\ \rho'_{\text{fin}}\left(F^{k^+(\rho)}\right) \subseteq \rho'_{\text{fin}}(G)}} G.$$

Now, applying Propositions 4 to the latter term results in (1).

Next, we prove the following equality

$$\left\{G \in \mathcal{F}_{\text{fin}} \mid G \sqsubseteq F^{k^+(\rho)}\right\} = \left\{G \in \mathcal{F}_{\text{fin}} \mid \bigcup_{\substack{H \in \mathcal{F}, \\ F^{k^+(\rho)} \sqsubseteq H}} [H]^\rho_s \subseteq \bigcup_{\substack{H \in \mathcal{F}, \\ G \sqsubseteq H}} [H]^\rho_s\right\} \tag{2}$$

(⊆) Given an AF G, s.t. $G \sqsubseteq F^{k^+(\rho)}$. Let $H' \in \bigcup_{H \in \mathcal{F}, F^{k^+(\rho)} \sqsubseteq H} [H]^\rho_s$. Hence, there is an H with $F^{k^+(\rho)} \sqsubseteq H$ and $H' \in [H]^\rho_s$. Applying the assumption yields $G \sqsubseteq H$ and $H' \in \bigcup_{H \in \mathcal{F}, G \sqsubseteq H} [H]^\rho_s$ is shown.

(⊇) Now, consider an AF G, s.t. $\bigcup_{H \in \mathcal{F}, F^{k^+(\rho)} \sqsubseteq H} [H]^\rho_s \subseteq \bigcup_{H \in \mathcal{F}, G \sqsubseteq H} [H]^\rho_s$. Since $F^{k^+(\rho)} \sqsubseteq F^{k^+(\rho)}$ we deduce $\left[F^{k^+(\rho)}\right]^\rho_s \subseteq \bigcup_{H \in \mathcal{F}, G \sqsubseteq H} [H]^\rho_s$. Thus, there is an F' with $G \sqsubseteq F'$ and $F' \in \left[F^{k^+(\rho)}\right]^\rho_s$. As $F^{k^+(\rho)}$ is the $\sqsubseteq$-greatest element in $\left[F^{k^+(\rho)}\right]^\rho_s$ we get $F' \sqsubseteq F^{k^+(\rho)}$ and thus, $G \sqsubseteq F^{k^+(\rho)}$.

Finally, combining (2) and (1) yields $Cn^{\rho'}(F) = \bigsqcup \left\{G \in \mathcal{F}_{\text{fin}} \mid G \sqsubseteq F^{k^+(\rho)}\right\} = F^{k^+(\rho)}$ concluding the proof. □

Note that $Cn^{\rho'_{\text{fin}}}$ indeed satisfies the decisive properties of a consequence operator. We put them in the form of a proposition.

Proposition 7. *Consider two AFs F and G and any $\rho \in \{stb, ad, co, gr, pr\}$.*

1. $F \sqsubseteq F^{k^+(\rho)}$ *(inclusion)*
2. $\left(F^{k^+(\rho)}\right)^{k^+(\rho)} = F^{k^+(\rho)}$ *(idempotency)*
3. *If $F \sqsubseteq G$, then $F^{k^+(\rho)} \sqsubseteq G^{k^+(\rho)}$* *(monotonicity)*

5 Conclusion and Related Work

We have shown that for a whole family of logics, namely logics satisfying the intersection property, we have that consequence operators of any characterization logic coincide with the consequence operator of the underlying logic. We want to highlight two points: First, this result covers (but is not limited to) well-known logics like propositional logic, first-order logic or modal logic and secondly, the result is achieved from a very abstract view on logics, in particular, there is no recourse on syntactical specifics of a certain logic.

The main part of the paper was the consideration of abstract argumentation theory. We applied a former existence result to the specifics of argumentation semantics. We showed that characterization logics return the reverse kernel of a given AF. Such a kernel contains the initial AF and additionally includes any

redundant (w.r.t. strong equivalence) attack. Note that, at least after certain period of reflection, this result is not as strange as it first seems: In propositional logic, by comparison, the consequence operator returns the initial theory and additionally includes any already implicit – that is, "redundant" (w.r.t. ordinary equivalence) – formula.

For future work, it would be interesting to consider other formalisms (apart from argumentation frameworks) whose strong equivalence has been studied in the literature. Most prominently, we envision analysing (recent extensions of) answer set programming languages [11,8,7]. In another direction, we also plan to analyze uniform [10] and other intermediate notions of equivalence [20] in our general setting. The main challenge there is to translate the typically syntactical restrictions (on theories U that are allowed for extending) into our general framework, where there is no access to syntax.

Acknowledgments. This work was supported by the German Research Foundation (DFG, BA 6170/3-1) and by the German Federal Ministry of Education and Research (BMBF, 01/S18026A-F) by funding the competence center for Big Data and AI "ScaDS.AI" Dresden/Leipzig. We also acknowledge funding from BMBF within projects KIMEDS (grant no. GW0552B), MEDGE (grant no. 16ME0529), and SEMECO (grant no. 03ZU1210B).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Baroni, P., Caminada, M., Giacomin, M.: Abstract argumentation frameworks and their semantics. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, chap. 4. College Publications (February 2018)
2. Baumann, R.: On the nature of argumentation semantics: Existence and uniqueness, expressibility, and replaceability. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, chap. 14. College Publications (February 2018)
3. Baumann, R., Penndorf, C.H.: Static and dynamic orderings on Dungean argumentation frameworks – An overview. Int. J. Approx. Reason. **163**, 109036 (2023), https://doi.org/10.1016/j.ijar.2023.109036
4. Baumann, R., Strass, H.: An abstract logical approach to characterizing strong equivalence in logic-based knowledge representation formalisms. In: Baral, C., Delgrande, J.P., Wolter, F. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Fifteenth International Conference, KR 2016, Cape Town, South Africa, April 25–29, 2016. pp. 525–528. AAAI Press (2016), http://www.aaai.org/ocs/index.php/KR/KR16/paper/view/12834
5. Baumann, R., Strass, H.: An abstract, logical approach to characterizing strong equivalence in non-monotonic knowledge representation formalisms. Artificial Intelligence **305**, 103680 (2022). https://doi.org/10.1016/j.artint.2022.103680
6. Baumann, R., Woltran, S.: The role of self-attacking arguments in characterizations of equivalence notions. Journal of Logic and Computation **26**(4), 1293–1313 (2016), http://dx.doi.org/10.1093/logcom/exu010

7. Cabalar, P., Fandinno, J., Schaub, T., Wanko, P.: On the semantics of hybrid ASP systems based on clingo. Algorithms **16**(4), 185 (2023), https://doi.org/10.3390/a16040185
8. Cabalar, P., Kaminski, R., Ostrowski, M., Schaub, T.: An ASP semantics for default reasoning with constraints. In: Kambhampati, S. (ed.) Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016. pp. 1015–1021. IJCAI/AAAI Press (2016), http://www.ijcai.org/Abstract/16/148
9. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence **77**(2), 321–357 (1995)
10. Eiter, T., Fink, M.: Uniform equivalence of logic programs under the stable model semantics. In: Palamidessi, C. (ed.) Logic Programming, 19th International Conference, ICLP 2003, Mumbai, India, December 9-13, 2003, Proceedings. Lecture Notes in Computer Science, vol. 2916, pp. 224–238. Springer (2003), https://doi.org/10.1007/978-3-540-24599-5_16
11. Fandinno, J., Lifschitz, V.: Omega-completeness of the logic of here-and-there and strong equivalence of logic programs. In: Marquis, P., Son, T.C., Kern-Isberner, G. (eds.) Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning, KR 2023, Rhodes, Greece, September 2-8, 2023. pp. 240–251 (2023), https://doi.org/10.24963/kr.2023/24
12. Gabbay, D.M., Hogger, C.J., Robinson, J.A. (eds.): Handbook of Logic in Artificial Intelligence and Logic Programming: Nonmonotonic Reasoning and Uncertain Reasoning, vol. 3. Oxford University Press, Inc., New York, NY, USA (1994)
13. Gaggl, S.A., Woltran, S.: The cf2 argumentation semantics revisited. Journal of Logic and Computation **23**, 925–949 (2013)
14. Lifschitz, V., Pearce, D., Valverde, A.: Strongly equivalent logic programs. ACM Transactions on Computational Logic **2**(4), 526–541 (2001). https://doi.org/10.1145/502166.502170, http://doi.acm.org/10.1145/502166.502170
15. Oikarinen, E., Woltran, S.: Characterizing strong equivalence for argumentation frameworks. Artificial Intelligence **175**, 1985–2009 (2011)
16. Truszczyński, M.: Strong and uniform equivalence of nonmonotonic theories – an algebraic approach. Annals of Mathematics and Artificial Intelligence **48**(3–4), 245–265 (2006). https://doi.org/10.1007/s10472-007-9049-2, http://dx.doi.org/10.1007/s10472-007-9049-2
17. Truszczyński, M.: The modal logic s4f, the default logic, and the logic here-and-there. In: Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22–26, 2007, Vancouver, British Columbia, Canada. pp. 508–514. AAAI Press (2007), http://www.aaai.org/Library/AAAI/2007/aaai07-080.php
18. Turner, H.: Strong equivalence for logic programs and default theories (made easy). In: LPNMR. pp. 81–92 (2001)
19. Turner, H.: Strong equivalence for causal theories. In: Lifschitz, V., Niemelä, I. (eds.) Logic Programming and Nonmonotonic Reasoning, 7th International Conference, LPNMR 2004, Fort Lauderdale, FL, USA, January 6–8, 2004, Proceedings. Lecture Notes in Computer Science, vol. 2923, pp. 289–301. Springer (2004), https://doi.org/10.1007/978-3-540-24609-1_25
20. Woltran, S.: A common view on strong, uniform, and other notions of equivalence in answer-set programming. Theory Pract. Log. Program. **8**(2), 217–234 (2008), https://doi.org/10.1017/S1471068407003250