Are Ranking Semantics Sensitive to the Notion of Core?

Bruno Yun*
University Of Montpellier
yun@lirmm.fr

Madalina Croitoru
University Of Montpellier
croitoru@lirmm.fr

Pierre Bisquert
IATE, INRA Montpellier
pierre.bisquert@inra.fr

ABSTRACT

In this paper, we study the impact of two notions of core on the output of ranking semantics in logical argumentation frameworks. We consider the existential rules fragment, a language widely used in Semantic Web and Ontology Based Data Access applications. Using burden semantics as example we show how some ranking semantics yield different outputs on the argumentation graph and its cores. We extend existing results in the literature regarding core equivalences on logical argumentation frameworks and propose the first formal characterisation of core-induced modification for a class of ranking semantics satisfying given postulates.

1. INTRODUCTION

Logical based argumentation instantiates abstract argumentation frameworks [18] by constructing arguments from inconsistent knowledge bases (KB), computing attacks between them and using so-called argumentation semantics in order to select acceptable arguments and their conclusions. Several approaches for logic based argumentation exist in the literature: assumption-based argumentation frameworks (ABA) [11], DeLP [19], deductive argumentation (where an argument is perceived as a tuple (H, C) of a set of premises H and a conclusion C) [9] or ASPIC/ASPIC+ [23]. In this paper we do not follow the argumentation semantics "a la Dung" introduced by [18] but study ranking semantics [8, 22, 21, 15, 2] that return a total order over the set of arguments in the logical argumentation framework. For instantiation, we focus on a particular subset of first order logic: existential rules. The choice of the language is motivated by the practical relevance demonstrated by many applications [25, 24, 27] but also certain features that make it challenging for instantiating argumentation frameworks such as n-ary (as opposed to binary only) negative constraints or existential variables in the head of rules.

In logical based argumentation, arguments are sometimes based upon equivalent data. Cores are notions introduced in [4] that delete such arguments. We investigate two different notions of core in such a logically instantiated argumentation framework that will remove redundant arguments and attacks in a different manner. We then ask the following research question: "*Will the manner of defining the core of a logically instantiated argumentation framework affect the ranking output of ranking semantics?*". Our initial intuition was that the answer is "*no*" since the core of an argumentation framework is supposed to return an equivalent, but smaller, argumentation framework. Surprisingly, the answer is "*yes*". We give an example of such a change using one particular ranking semantics and show how such a behaviour is not unique. Our contribution is thus not only to uncover this unexpected behaviour but also to investigate some of its reasons. The salient points of the paper are :

- The first investigation of ranking semantics in the first order logic fragment of existential rules.
- The study of several notions of core in logical argumentation framework and the proof of their equivalences and properties.
- The first characterisation of core-induced ranking modifications of semantics satisfying postulates from [12].

The paper is organized as follows. Section 2 gives the relevant background regarding existential rules and presents the logical instantiation of an argumentation framework using this language (i.e. the structure of the arguments and the attacks). In Section 3, we recall the existing notions of core and focus on two types of core and their different properties. In Section 4, we explain the different changes that can occur to a ranking of arguments when considering the cores and conclude the paper in Section 5.

2. LOGIC BASED ARGUMENTATION

This section will put the basis for logical argumentation and its instantiation using existential rules. After describing the logical language used in this paper, existential rules, we give an instantiation of a logical argumentation framework (AF) that uses this language. We extend the state of the art by considering a definition of an argument that imposes minimality.

In this paper we are interested in argumentation frameworks instantiated using existential rules [16, 17, 5]. The existential rules language [13] extends plain Datalog with *existential variables* in the rule head and is composed of formulae built with the usual quantifiers ($\exists, \forall$) and *only* two connectors: implication ($\rightarrow$) and conjunction ($\wedge$). A subset of this language, also known as $Datalog^{\pm}$, refers to identified decidable existential rule fragments [20, 7]. The language has attracted much interest recently in the Semantic Web and Knowledge Representation community for its suitabil-

*Corresponding Author.

Appears in: *Proc. of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017), S. Das, E. Durfee, K. Larson, M. Winikoff (eds.), May 8–12, 2017, São Paulo, Brazil.*

ity for representing knowledge in a distributed context (such as Ontology Based Data Access applications). The language is composed of the following elements:

- A *fact* is a ground atom (of the form $p(t_1, \ldots, t_k)$ where p is a predicate of arity k and $t_i, i \in [1, \ldots, k]$ are constants.
- An existential *rule* is of the form $\forall \overrightarrow{X}, \overrightarrow{Y}\ H[\overrightarrow{X}, \overrightarrow{Y}] \rightarrow \exists \overrightarrow{Z} C[\overrightarrow{Z}, \overrightarrow{X}]$ where H and C are conjunctions of closed atoms and $\overrightarrow{X}, \overrightarrow{Y}, \overrightarrow{Z}$ their respective sets of variables.
- A *negative constraint* is a particular kind of rule where H is a conjunction of atoms and C is $\perp$ (*absurdum*). It implements weak negation.
- A KB $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ is composed of a set of facts $\mathcal{F}$, a set of rules $\mathcal{R}$ and a set of negative constraints $\mathcal{N}$. We denote by $\mathcal{C}\ell^*_{\mathcal{R}}(\mathcal{F})$ the *closure* of $\mathcal{F}$ by $\mathcal{R}$ (computed by all possible applications of the rules in $\mathcal{R}$ over $\mathcal{F}$ until a fixed point is reached). $\mathcal{C}\ell^*_{\mathcal{R}}(\mathcal{F})$ is said to be $\mathcal{R}$-*consistent* if no negative constraint hypothesis can be deduced from it. Otherwise $\mathcal{C}\ell^*_{\mathcal{R}}(\mathcal{F})$ is $\mathcal{R}$-*inconsistent*. When considering consistent facts, entailment implicitly considers rules application (i.e. $\mathcal{F} \models q$ is equivalent to $\mathcal{C}\ell^*_{\mathcal{R}}(\mathcal{F}) \models q$).

Let us define the notion of an argument in this logical language. For decidability reasons in the following we only consider the $Datalog^{\pm}$ fragment. An argument is composed of a set of facts called hypothesis (or the support of the argument) and a conclusion entailed from the hypothesis. The following definition of an argument improves upon the state of the art [16, 17] by considering hypothesis *minimality* and removing sequences of derivations. We chose to represent the conclusion as a conjunction of facts instead of a set of facts because the entailment of a conjunction is not equal to the entailment of every fact of a set.

Definition 1. Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a KB. An argument a is a tuple (H, C) with H a $\mathcal{R}$-consistent subset of $\mathcal{F}$ and C a conjunction of facts such that:

- $H \subseteq \mathcal{F}$ and $\mathcal{C}\ell^*_{\mathcal{R}}(H) \not\models \perp$.
- $\mathcal{C}\ell^*_{\mathcal{R}}(H) \models C$.
- $\nexists H' \subset H$ s.t. $\mathcal{C}\ell^*_{\mathcal{R}}(H') \models C$.

The support H of an argument a is denoted by $Supp(a)$ and the conclusion C by $Conc(a)$. If X is a set of arguments, we denote by $Base(X) = \bigcup_{a \in X} Supp(a)$.

EXAMPLE 1. *Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a $Datalog^{\pm}$ KB with:*

- $\mathcal{F} = \{Pussycat(Tom), Cat(Tom), Mouse(Jerry), Turtle(John)\}$.
- $\mathcal{R} = \{\forall x(Cat(x) \rightarrow Pussycat(x)), \forall x(Pussycat(x) \rightarrow Cat(x)), \forall x(Cat(x) \rightarrow Mammal(x)), \forall x(Mouse(x) \rightarrow Mammal(x))\}$.
- $\mathcal{N} = \{\forall x, y, z(Cat(x) \wedge Mouse(y) \wedge Turtle(z) \rightarrow \perp)\}$.

From Definition 1 we can obtain the set of all arguments $\mathcal{A}$ of $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ (it contains 75 arguments). An example of an argument $a_1 \in \mathcal{A}$ is $a_1 = (\{Pussycat(Tom)\}, Pussycat(Tom))$. Another example is $a_{18} = (\{Cat(Tom), Mouse(Jerry)\}, Mammal(Tom) \wedge Mammal(Jerry))$.

Similar to [17] we consider here the undermine attack: an argument a attacks an argument b iff the conclusion of a and an element of the support of b are $\mathcal{R}$-inconsistent. Please note that, due to the possibility of having more than two atoms in the hypothesis of negative constraints in the language, this attack is not symmetric [16, 17].

Definition 2. Argument a attacks b denoted by $(a, b) \in \mathcal{C}$ iff $\exists \phi \in Supp(b)$ s.t. $\{Conc(a), \phi\}$ is $\mathcal{R}$-inconsistent. The set of attackers of $a \in \mathcal{A}$ is denoted by $Att(a)$.

EXAMPLE 1 (CONT.). *Let us consider $a_{12} \in \mathcal{A}$ such that $a_{12} = (\{Turtle(John), Mouse(Jerry)\}, Turtle(John) \wedge Mouse(Jerry))$. We have $(a_{12}, a_1) \in \mathcal{C}$ but $(a_1, a_{12}) \notin \mathcal{C}$ because conclusion of a_1 ($Pussycat(Tom)$) is not $\mathcal{R}$-inconsistent if it is matched with only one fact of $Supp(a_{12})$.*

Following the proof of [17] we can easily see that the obtained argumentation framework $\mathcal{AS}_{\mathcal{K}} = (\mathcal{A}, \mathcal{C})$ with $\mathcal{A}$ being the set of arguments and $\mathcal{C}$ the set of attacks defined above satisfies the rationality postulates of [1, 14]. However, its main drawback lies in the large number of the generated arguments (and corresponding attacks). As shown in Example 1, for a simple and basic KB of four facts, four rules and a negative constraint we can generate 75 arguments (even when taking into account minimality). The large number of (potentially equivalent) arguments might impact on the structure of the ranking output of logical argumentation frameworks.

This is why in the next section we investigate the notion of core. Our hypothesis is that considering the core will allow us to reduce the number of arguments and attacks while preserving the same logically equivalent output.

3. CORE EQUIVALENCES

The *core* is a notion introduced in [4] that enables to simplify logically instantiated argumentation frameworks without losing data. In this section we will use two notions of core initially defined in [4] and we will adapt them to the logical instantiation using existential rules of this paper. We give an example of how the two core notions yield argumentation frameworks with significantly less arguments for the same logical output and prove two new key results that extend the state of the art. First, we give the relation between the base of the two cores for existential rule instantiated argumentation frameworks. Second, we show how the two cores can be obtained from each other.

3.1 Equivalence & Core Definitions

The notion of core relies on the notion of equivalence of formulae, arguments and, subsequently, of induced argumentation frameworks. To define the notion of core we first need to define the notion of equivalence of formulae. Adapting [4] for existential rules, two facts are equivalent if the sets given by the closure[1] of each fact are equal. Similarly, we say that two sets of facts are equal if, for each fact in every set, we can find an equivalent fact in the other set.

Definition 3. Let f_1, f_2 be two facts or conjunction of facts and $\mathcal{F}_1$, $\mathcal{F}_2$ be two sets of facts. We say that:

- f_1 and f_2 are equivalent ($f_1 \equiv f_2$) iff $\mathcal{C}\ell^*_{\mathcal{R}}(f_1) = \mathcal{C}\ell^*_{\mathcal{R}}(f_2)$.

[1]In the following we consider that the rule application is using the restricted chase that does not consider redundant new facts generated by each step of the rule application (see more details in [6]).

- $\mathcal{F}_1$ and $\mathcal{F}_2$ are equivalent ($\mathcal{F}_1 \cong \mathcal{F}_2$) iff $\forall f \in \mathcal{F}_1, \exists f' \in \mathcal{F}_2$ s.t. $f \equiv f'$ and $\forall f \in \mathcal{F}_2, \exists f' \in \mathcal{F}_1$ s.t. $f \equiv f'$. $\mathcal{F}_1 \not\cong \mathcal{F}_2$ otherwise.

EXAMPLE 1 (CONT.). *We have that* $Pussycat(Tom) \equiv Cat(Tom)$ *since* $\mathcal{C}\ell^*_{\mathcal{R}}(Pussycat(Tom)) = \mathcal{C}\ell^*_{\mathcal{R}}(Cat(Tom)) = \{Cat(Tom), Pussycat(Tom), Mammal(Tom)\}$.

Using the equivalence of formulae in Definition 3 and following [4] we can now define the notion of equivalence between arguments. We will consider *two equivalence relations*. The first one ($\approx_1$) considers two arguments as being equivalent if they have *equal supports and equivalent conclusions*. The second one ($\approx_2$) considers two arguments as being equivalent if they have *equivalent supports and equivalent conclusions*. Note that if there are two arguments a and a' such that $a \approx_1 a'$ then obviously $a \approx_2 a'$.

Definition 4. [4] Let a and a' be two arguments. We have:

- $a \approx_1 a'$ iff $Supp(a) = Supp(a')$ and $Conc(a) \equiv Conc(a')$.
- $a \approx_2 a'$ iff $Supp(a) \cong Supp(a')$ and $Conc(a) \equiv Conc(a')$.

EXAMPLE 1 (CONT.). *Let us consider* $a_{13} \in \mathcal{A}$ *s.t.* $a_{13} = (\{Cat(Tom)\}, Cat(Tom))$. *Thus,* $a_{13} \not\approx_1 a_1$ *and* $a_{13} \approx_2 a_1$.

PROPOSITION 1. *Let* $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ *be an AF and* $a, a' \in \mathcal{A}$. *If* $(a, a') \in \mathcal{C}$ *then* $a \not\approx_1 a'$ *and* $a \not\approx_2 a'$.

Before we can define the notion of core, we first need to give the notions of equivalence relation, equivalence class and the set of all possible equivalence classes .

Definition 5. If X is a set of elements, $\sim$ an equivalence relation on X and $x \in X$, then $\bar{x}_\sim = \{x' \in X | x' \sim x\}$ (we say that $\bar{x}_\sim$ is the equivalence class of an element x for equivalence relation $\sim$). Finally, the set of all possible equivalence classes will be denoted by $X/\sim = \{\bar{x}_\sim | x \in X\}$. Note that for readability purposes, we will sometimes denote $\bar{x}_\sim$ by $\bar{x}$ whenever the equivalence relation is obvious.

We are now ready to define the notion of core of a logical argumentation framework. A core of an argumentation system $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ is an argumentation system that can be seen as a particular subgraph $\mathcal{AS}' = (\mathcal{A}', \mathcal{C}')$ of $\mathcal{AS}$. There are three restrictions. First, $\mathcal{A}'$ must obviously be a subset of the set of arguments $\mathcal{A}$. Second, for a given equivalence relation $\approx$ on the arguments, there must be a *unique* argument in $\mathcal{A}'$ for each equivalence class. Last but not least, $\mathcal{C}'$ must be a restriction of $\mathcal{C}$ to the arguments of $\mathcal{AS}'$.

Definition 6. [4] Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ and $\mathcal{AS}' = (\mathcal{A}', \mathcal{C}')$ be two AFs and $\approx$ an equivalence relation on arguments. $\mathcal{AS}'$ is a core of $\mathcal{AS}$ iff:

- $\mathcal{A}' \subseteq \mathcal{A}$
- $\forall G \in \mathcal{A}/\approx, \exists! a \in \mathcal{A}$ s.t. $a \in G \cap \mathcal{A}'$ for the given equivalence relation $\approx$.
- $\mathcal{C}' = \mathcal{C}_{|\mathcal{A}'}$.

We denote by $Core_\approx(\mathcal{AS})$ the set of all cores of an argumentation framework $\mathcal{AS}$ for equivalence relation $\approx$.

Note that since we consider two equivalence relations for arguments we can naturally construct two sets of cores from an AF $\mathcal{AS}$: $Core_{\approx_1}(\mathcal{AS})$ that follows the first equivalence relation and $Core_{\approx_2}(\mathcal{AS})$ that follows the second.

EXAMPLE 1 (CONT.). *We are interested in what arguments are contained in two distinct notions of core using the previous definitions. Table 1 has five columns. The first three columns represents an example of 20 arguments (out of 75) that can be constructed over the KB of Example 1 along with their respective supports and conclusions. The last two columns show whether the 20 arguments belong or not to two examples of cores* c_1 *and* c_2. *The two examples of cores have been constructed using respectively the first and the second equivalence relations: core* $c_1 \in Core_{\approx_1}(\mathcal{AS}_\mathcal{K})$ *and core* $c_2 \in Core_{\approx_2}(\mathcal{AS}_\mathcal{K})$ *(such that it is included in* $Core_{\approx_2}(c_1)$*, as it can be verified in Table 1). Please note that for space reasons we did not write the full predicate names.*

Name	Support	Conclusion	c_1	c_2
a_1	$\{P\}$	P	✓	✓
a_2	$\{P\}$	M_T	✓	✓
a_3	$\{M\}$	M	✓	✓
a_4	$\{M\}$	M_J	✓	✓
a_5	$\{T\}$	T	✓	✓
a_6	$\{P, M\}$	$P \wedge M$	✓	✓
a_7	$\{P, M\}$	$M_T \wedge M$	✓	✓
a_8	$\{P, M\}$	$P \wedge M_J$	✓	✓
a_9	$\{P, M\}$	$M_T \wedge M_J$	✓	✓
a_{10}	$\{P, T\}$	$P \wedge T$	✓	✓
a_{11}	$\{P, T\}$	$M_T \wedge T$	✓	✓
a_{12}	$\{T, M\}$	$T \wedge M$	✓	✓
a_{13}	$\{C\}$	C	✓	
a_{14}	$\{C\}$	M_T	✓	
a_{15}	$\{C, M\}$	$C \wedge M$	✓	
a_{16}	$\{C, M\}$	$M_T \wedge M$	✓	
a_{17}	$\{C, M\}$	$C \wedge M_J$	✓	
a_{18}	$\{C, M\}$	$M_T \wedge M_J$	✓	
a_{19}	$\{C, T\}$	$C \wedge T$	✓	
a_{20}	$\{C, T\}$	$M_T \wedge T$	✓	

Acronym	Meaning
C	$Cat(Tom)$
M_J	$Mammal(Jerry)$
M_T	$Mammal(Tom)$
M	$Mouse(Jerry)$
P	$Pussycat(Tom)$
T	$Turtle(John)$

Table 1: Some arguments constructed from the KB of Example 1 and two particular cores obtained using respectively $\approx_1$ and $\approx_2$.

The next section shows properties on the two types of cores obtained from the equivalence relation $\approx_1$ and $\approx_2$.

3.2 Core equivalence properties

Let us first summarize the theoretical results of this section. In Proposition 2, we show that the attack relation of Definition 2 satisfies properties of [4] which implies equivalences between the argumentation framework and any of its cores. In Proposition 3, we show that it is not useful to employ a more restrictive equivalence relation (and therefore a more general AF) once a core has already been obtained using a less restrictive one (outputting a more constrained AF). In Proposition 4, we show that all cores constructed using $\approx_2$ can be constructed using specific cores of $\approx_1$ on which we compute a core using $\approx_2$. This basically means that we can bypass the core constructed with $\approx_1$ when we are interested by a less restrictive relation such as $\approx_2$. Proposition 3 and Proposition 4 combined provide an important result as it will allow us not to be concerned about the order of applying cores on the argumentation framework.

According to [4] there are two properties that are satisfied by the attack relation. First, if two arguments have equivalent conclusions, then they attack the same arguments. Second, if two arguments have equivalent supports, then they are attacked by the same arguments. We show next that we respect both conditions in the following proposition.

PROPOSITION 2. *Given a logical AF $\mathcal{AS}_\mathcal{K} = (\mathcal{A}, \mathcal{C})$ with $\mathcal{A}$ being the set of arguments defined by Definition 1 and $\mathcal{C}$ the set of attacks defined according to Definition 2, the set $\mathcal{C}$ enjoys the following properties:*

1. $C1b : \forall a, b, c \in \mathcal{A}$, *if* $Conc(a) \equiv Conc(b)$ *then* $((a, c) \in \mathcal{C} \Leftrightarrow (b, c) \in \mathcal{C})$.
2. $C2b : \forall a, b, c \in \mathcal{A}$, *if* $Supp(a) \cong Supp(b)$ *then* $((c, a) \in \mathcal{C} \Leftrightarrow (c, b) \in \mathcal{C})$.

PROOF. Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ a KB expressed using existential rules and $\mathcal{AS}_\mathcal{K} = (\mathcal{A}, \mathcal{C})$ the corresponding argumentation framework. Now, we consider $a, b, c \in \mathcal{A}$.

1. Suppose that $Conc(a) \equiv Conc(b)$. If $(a, c) \in \mathcal{C}$, it means that $\exists \phi \in Supp(c)$ such that there is a negative constraint $N \in \mathcal{N}$ with $\mathcal{C}\ell^*_\mathcal{R}(\{Conc(a), \phi\}) \models H_N$, where H_N denotes the existential closure of the hypothesis of N. However, since $Conc(a) \equiv Conc(b)$, we can infer that $\mathcal{C}\ell^*_\mathcal{R}(Conc(a)) = \mathcal{C}\ell^*_\mathcal{R}(Conc(b))$, thus $\mathcal{C}\ell^*_\mathcal{R}(\{Conc(b), \phi\}) \models H_N$ and $(b, c) \in \mathcal{C}$. Likewise, $((b, c) \in \mathcal{C} \Rightarrow (a, c) \in \mathcal{C})$ which ends the proof.
2. Suppose now that $Supp(a) \cong Supp(b)$. If $(c, a) \in \mathcal{C}$, it means that $\exists \phi \in Supp(a)$ such that there is a negative constraint $N \in \mathcal{N}$ with $\mathcal{C}\ell^*_\mathcal{R}(\{Conc(c), \phi\}) \models H_N$. However, since $Supp(a) \cong Supp(b)$, by definition, we have that $\exists \phi' \in Supp(b)$ s.t. $\phi' \equiv \phi$, i.e. $\mathcal{C}\ell^*_\mathcal{R}(\phi') = \mathcal{C}\ell^*_\mathcal{R}(\phi)$. Therefore, we can infer that $\mathcal{C}\ell^*_\mathcal{R}(\{Conc(c), \phi'\}) \models H_N$ and $(c, b) \in \mathcal{C}$. Likewise, $((c, b) \in \mathcal{C} \Rightarrow (c, a) \in \mathcal{C})$ which ends the proof.

□

A natural question one can ask at this point is whether the order of applying the cores matters. To answer this question, we provide two main results. The first proposition shows that using a more restrictive equivalence relation than the one used to compute a core does not change this core. We begin by defining the notion of less restrictive equivalence relation and follow with the proposition.

Definition 7. Let $\approx$ and $\approx'$ be two equivalence relation on a set of elements X, we say that $\approx$ is *more restrictive* than $\approx'$ (and thus, $\approx'$ is *less restrictive* than $\approx$) iff $\forall x, x' \in X$ s.t. $x \approx x' \Rightarrow x \approx' x'$.

PROPOSITION 3. *Let $\mathcal{AS}$ be an AF and $\approx, \approx'$ two equivalence relation such that $\approx$ is more restrictive than $\approx'$. It holds that: $\forall c' \in Core_{\approx'}(\mathcal{AS}), Core_\approx(c') = \{c'\}$.*

PROOF. Suppose that we have $\forall a, a' \in \mathcal{A}, a \approx a' \Rightarrow a \approx' a'$. Now, let us consider $c' = (\mathcal{A}', \mathcal{C}') \in Core_{\approx'}(\mathcal{AS})$ and $c = (\mathcal{A}, \mathcal{C}) \in Core_\approx(c')$. We denote by X the set such that $\mathcal{A}' = X \cup \mathcal{A}$ and $X \cap \mathcal{A} = \emptyset$. If $X \neq \emptyset$, it means that $\exists b \in X$ and $\exists b' \in \mathcal{A}$ s.t. $b \not\approx' b'$ and $b \approx b'$, contradiction. It follows that: $X = \emptyset$ and that c' is the only core of $Core_\approx(c')$. □

Now, we can prove the most important property of this section and namely that the set of cores of an argumentation framework $\mathcal{AS}$ using $\approx_2$ is equal to the set of cores using $\approx_2$ that are built on cores of $\mathcal{AS}$ using $\approx_1$.

PROPOSITION 4. *Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF and $\approx_1, \approx_2$ be the equivalence relations defined in Definition 4. We have that: $Core_{\approx_2}(\mathcal{AS}) = \bigcup_{c_1 \in Core_{\approx_1}(\mathcal{AS})} Core_{\approx_2}(c_1)$.*

PROOF. This proof will be split in two parts:

- ($\subseteq$) We prove this inclusion by construction. Let $c' = (\mathcal{A}', \mathcal{C}') \in Core_{\approx_2}(\mathcal{AS})$, by definition, $\forall G \in \mathcal{A}/\approx_1$, we chose an unique x in G for c_1. Here, if $\exists a \in G \cap A'$ then we choose $x = a$ otherwise we choose a random element of G. Now that we have a specific core c_1 of $\mathcal{A}$ for $\approx_1$, we repeat the process and construct c_2 from c_1. In the end, $c_2 = c'$.
- ($\supseteq$) Let $c_1 = (\mathcal{A}_1, \mathcal{C}_1) \in Core_{\approx_1}(\mathcal{AS})$ and $c_2 = (\mathcal{A}_2, \mathcal{C}_2) \in Core_{\approx_2}(c_1)$. We prove that $c_2 \in Core_{\approx_2}(\mathcal{AS})$. We will proceed by proving each parts of Definition 6.
  - Since c_2 is a core of c_1 for equivalence relation $\approx_2$, then $\mathcal{A}_2 \subseteq \mathcal{A}_1$. Likewise, since c_1 is a core of $\mathcal{AS}$, we have $\mathcal{A}_1 \subseteq \mathcal{A}$. Finally, we have that $\mathcal{A}_2 \subseteq \mathcal{A}$.
  - Let $f : \mathcal{A}_1/\approx_2 \rightarrow \mathcal{A}/\approx_2$ a function that takes as input an element $\bar{x}$ of $\mathcal{A}_1/\approx_2$ and returns an element $\bar{\bar{x}}$ of $\mathcal{A}/\approx_2$ s.t. $\bar{x} \cap \bar{\bar{x}} \neq \emptyset$. We will show that this function is a bijection.
    * *Injective:* Suppose that $\exists \bar{x}, \bar{y} \in \mathcal{A}_1/\approx_2$ s.t. $f(\bar{x}) = f(\bar{y}) = \bar{\bar{z}}$ and $\bar{x} \neq \bar{y}$. By definition, it means that $\bar{\bar{z}} \cap \bar{x} \neq \emptyset$ and $\bar{\bar{z}} \cap \bar{y} \neq \emptyset$. Let $z_1 \in \bar{\bar{z}} \cap \bar{x}$ and $z_2 \in \bar{\bar{z}} \cap \bar{y}$. Since $z_1, z_2 \in \bar{\bar{z}}$, we have that $z_1 \approx_2 z_2$, contradiction with $\bar{x} \neq \bar{y}$.
    * *Surjective:* We have to prove that $\forall \bar{\bar{x}} \in \mathcal{A}/\approx_2, \exists \bar{x} \in \mathcal{A}_1/\approx_2$ s.t. $f(\bar{x}) = \bar{\bar{x}}$. Suppose that $\nexists \bar{x} \in \mathcal{A}_1/\approx_2$ s.t. $f(\bar{x}) = \bar{\bar{x}}$. Let us consider an argument $x \in \bar{\bar{x}}$. Then, $\exists G_1 \in \mathcal{A}/\approx_1$ s.t. $x \in G_1$. Furthermore, $\exists x_1 \in G_1$ s.t. $x_1 \in \mathcal{A}_1$. Keep in mind that $x_1 \in \bar{\bar{x}}$ since $x_1 \approx_1 x$ and thus $x_1 \approx_2 x$. Now, let $G \in A_1/\approx_2$ s.t. $x_1 \in G$. By definition of the core, $\exists! x_2 \in G \cap A_2$ and $x_2 \approx_2 x_1$ thus, $x_2 \in \bar{\bar{x}}$, contradiction.

    Since c_2 is a core of c_1 for $\approx_2$, we have that $\forall \bar{x} \in \mathcal{A}_1/\approx_2, \exists! x \in \bar{x} \cap \mathcal{A}_2$. But now, since f is a bijection, we can easily conclude that $\forall \bar{\bar{x}} \in \mathcal{A}/\approx_2, \exists! x \in \bar{\bar{x}} \cap \mathcal{A}_2$.
  - The final point is obvious since it is only a restriction of attacks.

This ends the proof. □

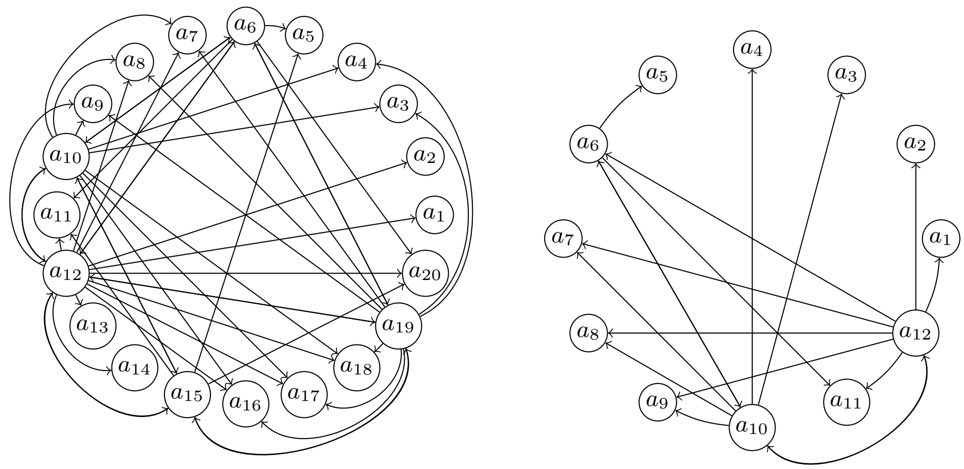
From Proposition 3 and Proposition 4, the following proposition holds:

PROPOSITION 5. *Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF and $\approx_1, \approx_2$ be the equivalence relations defined in Definition 4. We have that: $\bigcup_{c_2 \in Core_{\approx_2}(\mathcal{AS})} Core_{\approx_1}(c_2) = \bigcup_{c_1 \in Core_{\approx_1}(\mathcal{AS})} Core_{\approx_2}(c_1)$.*

Proposition 5 is very important for the next section that characterises ranking changes induced by cores. Proposition 5 tells us that if we are only concerned by $\approx_2$-induced ranking changes, we can bypass the core obtained via $\approx_1$.

4. RANKINGS ON DIFFERENT CORES

Now that we have investigated the notions of core for an argumentation framework, we can study how ranking semantics behave on them. In [3], the authors define the



(a) Representation of the core c_1 of $\mathcal{AS}_\mathcal{K}$ using $\approx_1$ and displayed in Table 1.

(b) Representation of the core c_2 of $\mathcal{AS}_\mathcal{K}$ using $\approx_2$ and displayed in Table 1.

Figure 1: Representation of cores of $\mathcal{AS}_\mathcal{K}$ using different equivalence relations.

notion of a argumentation based ranking logic (ARL) that takes a KB as input and, using ranking semantics on the instantiated argumentation graph, provides a ranking of the formulae of the KB. In the following we adapt their results and provide the corresponding existential rule argumentation based ranking logic. The extension we provide is two fold. First we consider the existential rules framework for instantiation. Second, we take into account the notion of core in the reasoning mechanism. The new process is composed of four steps:

1. First, an argumentation framework is instantiated from a KB $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ (See Example 1 for the KB considered as example throughout the paper).
2. Second, a core c constructed using an equivalence relation is considered (See Table 1 for two examples of cores on the KB from Example 1 considering the two equivalence relations defined in the previous section, and Figure 1 for the visual depiction of the cores as graphs).
3. Third, the arguments of c are ranked using a ranking semantics S (See Table 2 for the ranking of the arguments of the two cores from Figure 1 and Table 1 outputted by burden ranking semantics [2]).
4. Finally, their conclusions are ranked following a simple principle: a formula is ranked higher than another formula if it is supported by an argument which is ranked higher than any argument supporting the second formula (See Table 3).

Before commenting on the results of ranking on the KB, let us first define the ARL for existential rules. The definition follows the definition of [3] adapted for existential rules and the notion of core. Please note that the ranking on arguments resulting from a ranking semantics S on an argumentation framework $\mathcal{AS}$ will be denoted by $\preceq^S_{\mathcal{AS}}$ or simply by $\preceq$ if there is no ambiguity. For two arguments $a, b \in \mathcal{A}$, the notation $a \preceq b$ means that b is at least as acceptable as a.

Definition 8. An existential rule ARL is a tuple

$$L = (\mathcal{K}, \mathcal{AS}, c, S, K, \mathfrak{C}')$$

with $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ a KB expressed using existential rules, $\mathcal{AS}$ the instantiated argumentation framework (this may be omitted if the instantiation used is clear), $c = (\mathcal{A}', \mathcal{C}')$ the chosen core of $\mathcal{AS}$ for a given equivalence relation, S a ranking semantics and $K, \mathfrak{C}'$ defined as follows:

- $\forall X \subseteq \mathcal{F}, \mathfrak{C}'(X) = \{\phi | \exists a \in \mathcal{A}' \cap Arg(X) \text{ s.t. } Conc(a) \equiv \phi\}$, i.e. $\mathfrak{C}'(X)$ is the set of equivalent facts that can be concluded by arguments of the core c constructed on subsets of X.
- $\forall X \subseteq \mathcal{F}, \forall \phi, \psi \in \mathfrak{C}'(X), (\phi, \psi) \in K(X)$ iff $\exists a \in \mathcal{A}' \cap Arg(X)$ s.t. $Conc(a) \equiv \phi$ and $\exists b \in \mathcal{A}' \cap Arg(X)$ s.t. $Conc(b) \equiv \psi$, $(a, b) \in S((\mathcal{A}', \mathcal{C}'))$. $K(X)$ corresponds to a ranking on elements of $\mathfrak{C}'(X)$ obtained via the ranking of arguments $S((\mathcal{A}', \mathcal{C}'))$.

Note that the equivalence relation is absent from L because the core is already given. Let us now show by the means of an example that the ranking semantics considered (namely burden semantics) is sensitive to the notion of core and thus outputs different rankings for logically equivalent argumentation graphs.

Example 1 (cont.). *Let c_1 (resp. c_2) be the core of $\mathcal{AS}_\mathcal{K}$ using equivalence relation $\approx_1$ (resp. $\approx_2$). The argumentation graph of c_1 (resp. c_2) is represented in Figure 1a (resp. Figure 1b). The ranking on arguments of c_1 (resp. c_2) computed with the burden-based semantics is given in Table 2a (resp. Table 2b). Finally, the ranking of conclusions is computed and displayed in Table 3a (resp. Table 3b). Note that in this example, the discussion-based semantics [2] gives the same ranking.*

This example shows that, surprisingly, a core does not always have the same ranking as the original argumentation framework (since c_1 and c_2 have different rankings). For instance, a_1 is ranked higher than a_3 for c_1 (Table 2a) but a_1

a_1, a_2, a_{13}, a_{14}
a_3, a_4, a_5
$a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{15}, a_{16}, a_{17}, a_{18}, a_{19}, a_{20}$
a_{12}

(a) Ranking on arguments of c_1 using the burden-based (and Discussion-based) semantics.

a_1, a_2, a_3, a_4, a_5
$a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}$

(b) Ranking on arguments of c_2 using the burden-based (and Discussion-based) semantics.

Table 2: Ranking on arguments induced by different cores of $\mathcal{AS}_\mathcal{K}$.

P, M_T, C
M, M_J, T
$P \wedge M, M_T \wedge M, P \wedge M_J, M_T \wedge M_J, P \wedge T, M_T \wedge T,$ $C \wedge M, C \wedge M_J, C \wedge T$
$T \wedge M$

(a) Ranking on conclusions using the ranking on arguments of Table 2a.

P, M_T, M, M_J, T
$P \wedge M, M_T \wedge M, P \wedge M_J, M_T \wedge M_J, P \wedge T,$ $M_T \wedge T, T \wedge M$

(b) Ranking on conclusions using the ranking on arguments of Table 2b.

Table 3: Ranking on conclusions induced by different cores of $\mathcal{AS}_\mathcal{K}$.

is ranked equal to a_3 for c_2 (Table 2b). This change in the ranking of arguments is significant as it impacts the ranking of their conclusions. Hence, the atom $Cat(Tom)$ which was ranked higher than $Mouse(Jerry)$ and $Turtle(John)$ (Table 3a) is now ranked equal to them (Table 3b). This is caused by the existence of equivalences ($Cat(Tom)$ and $Pussycat(Tom)$) in the KB. In fact, these equivalences generate redundant attacks between arguments that decrease the ranking of other arguments. That is why, by deleting redundancy in cores, we can observe that the ranking of some arguments is modified.

Hence, the chosen equivalence relation also plays a role in the ranking (as we have different rankings for the two cores). The next subsection investigates the reasons for such a behaviour.

4.1 Characterising ranking changes

In the rest of this section, we consider an argumentation framework and one of its cores constructed either using $\approx_1$ or $\approx_2$. We give a necessary and sufficient condition for obtaining a equal (w.r.t. the set of arguments) $\approx_1$ and $\approx_2$-induced core from its original argumentation framework. Then, for those argumentation framework where the **induced core is different**, we provide sufficient conditions for characterising the difference between the ranking of the core and the one of its original argumentation framework. More precisely, we show that:

1. We provide a sufficient condition for **arguments that have their rank increased** in the induced core. The new ranking of these arguments is further characterised by a sufficient condition on their respective positions. This is done via the CP postulate characterisation.
2. We provide a sufficient condition for arguments that **do not change their rank** in the induced core. This is done via the NaE postulate characterisation.
3. Last, we provide a sufficient condition for arguments that **have their rank decreased**. This is done via the CP and SCT postulates characterisation.

Identity of induced core. We begin by introducing the notation needed for the rest of this section.

Definition 9. Let us consider an AF $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ and one of its core $c' = (\mathcal{A}', \mathcal{C}')$ for an equivalence relation, we will denote by $X_{c'}$ (or simply X if the core is obvious) the set of arguments that have been deleted, namely $\mathcal{A} = \mathcal{A}' \cup X, X \cap \mathcal{A}' = \emptyset$ and $\mathcal{C}' = \mathcal{C}_{|\mathcal{A}'}$. If $X \neq \emptyset$ then the core is said to be *different* than the argumentation framework, otherwise it is *no different*.

The next proposition gives a necessary and sufficient condition for all core using $\approx_1$ of an AF $\mathcal{AS}$ to be *no different* than the $\mathcal{AS}$.

Proposition 6. *Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a KB and $\mathcal{AS}_\mathcal{K}$ the corresponding AF. We have that $Core_{\approx_1}(\mathcal{AS}_\mathcal{K}) = \{\mathcal{AS}_\mathcal{K}\}$ iff for all $\mathcal{R}$-consistent subset Y of $\mathcal{F}, \nexists y_1, y_2, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_1, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_2$, $y_1 \neq y_2$ and $y_1 \equiv y_2$.*

Proof. We divide this proof in two parts:

- ($\Rightarrow$) We show that contrapositive of this implication is true by *reductio ad absurdum.* Suppose that there is a $\mathcal{R}$-consistent subset Y of $\mathcal{F}, \exists y_1, y_2, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_1, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_2, y_1 \neq y_2, y_1 \equiv y_2$ and $Core_{\approx_1}(\mathcal{AS}_\mathcal{K}) = \{\mathcal{AS}_\mathcal{K}\}$. Let us consider $Y'' \subseteq Y$ s.t. $\nexists Y' \subset Y''$ and $\mathcal{C}\ell^*_\mathcal{R} \models y_1$. We have that $a = (Y'', y_1)$ and $b = (Y'', y_2)$ are two arguments of $\mathcal{AS}_\mathcal{K}$. Furthermore, we have that $a \approx_1 b$ meaning that $\mathcal{AS}_\mathcal{K} \notin Core_{\approx_1}(\mathcal{AS}_\mathcal{K})$, contradiction.
- ($\Leftarrow$) We show that this implication is true by *reductio ad absurdum.* Suppose that $Core_{\approx_1}(\mathcal{AS}_\mathcal{K}) \neq \{\mathcal{AS}_\mathcal{K}\}$. It means that $\exists c_1 = (\mathcal{A}_1, \mathcal{C}_1) \in Core_{\approx_1}(\mathcal{AS}_\mathcal{K})$ with $X \neq \emptyset$. Therefore, it exists an argument $x \in X$ s.t. $x \in \mathcal{A}$ and $x \notin \mathcal{A}_1$. We deduce that $\exists x' \in \mathcal{A}_1$ s.t. $Conc(x) \equiv Conc(x'), Supp(x) = Supp(x')$. By definition of an argument, we have that $\mathcal{C}\ell^*_\mathcal{R}(Supp(x)) \models Conc(x)$ and $\mathcal{C}\ell^*_\mathcal{R}(Supp(x')) \models Conc(x')$, contradiction.

This ends the proof. □

Similarly, we show a necessary and sufficient condition for all core using $\approx_2$ of an AF $\mathcal{AS}$ to be *no different* than the $\mathcal{AS}$.

Proposition 7. *Let $\mathcal{K} = (\mathcal{F}, \mathcal{R}, \mathcal{N})$ be a KB and $\mathcal{AS}_\mathcal{K}$ the corresponding AF. We have $Core_{\approx_2}(\mathcal{AS}_\mathcal{K}) = \{\mathcal{AS}_\mathcal{K}\}$ iff $\nexists f_1, f_2 \in \mathcal{F}$ s.t. $f_1 \equiv f_2, f_1 \neq f_2$ and for all $\mathcal{R}$-consistent subset $Y \subseteq \mathcal{F}, \nexists y_1, y_2, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_1, \mathcal{C}\ell^*_\mathcal{R}(Y) \models y_2, y_1 \neq y_2$ and $y_1 \equiv y_2$.*

Proof. We divide this proof in two parts:

- ($\Rightarrow$) We show that contrapositive of this implication is true by *reductio ad absurdum* in the same fashion as the proof of Proposition 6. Indeed, following the same reasoning, we can deduce that there exists two arguments $a, b \in \mathcal{AS}_\mathcal{K}$ s.t. $a \approx_1 b$ and thus $a \approx_2 b$. It means that $\mathcal{AS}_\mathcal{K} \notin Core_{\approx_2}(\mathcal{AS}_\mathcal{K})$, contradiction.
- ($\Leftarrow$) We show that this implication is true by *reductio ad absurdum*. Suppose that $Core_{\approx_2}(\mathcal{AS}_\mathcal{K}) \neq \{\mathcal{AS}_\mathcal{K}\}$. It means that $\exists c_2 = (\mathcal{A}_2, \mathcal{C}_2) \in Core_{\approx_2}(\mathcal{AS}_\mathcal{K})$ with $X \neq \emptyset$. Therefore, it exists an argument $x \in X$ s.t. $x \in \mathcal{A}$ and $x \notin \mathcal{A}_2$. It means that $\exists x' \in \mathcal{A}_2$ s.t. $Conc(x) \equiv Conc(x'), Supp(x) \cong Supp(x')$. We can consider two cases which both lead to contradictions:
  - If $Supp(x) \neq Supp(x')$, there exists $a \in Supp(x)$ and $a \notin Supp(x')$ (resp. $a \notin Supp(x)$ and $a \in Supp(x')$). Since we have $Supp(x) \cong Supp(x')$, there exists $a' \in Supp(x')$ (resp. $a' \in Supp(x)$) s.t. $a \equiv a'$, contradiction.
  - If $Supp(x) = Supp(x')$, then since $\mathcal{C}\ell^*_\mathcal{R}(Supp(x)) \models Conc(x)$ and $\mathcal{C}\ell^*_\mathcal{R}(Supp(x')) \models Conc(x')$, contradiction

This ends the proof. $\square$

Rank increase. From now on, we consider an argumentation framework $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ and $c' = (\mathcal{A}', \mathcal{C}')$ one of its core for equivalence relation $\approx_1$ or $\approx_2$. An interesting property is that for each attack that comes from a argument removed by the core and reaches an argument of the core, we can find an attack that comes from an argument of the core and reaches the same argument.

Proposition 8. *Let us consider the set $E = \{(a, b) \in \mathcal{C} | a \in X$ and $b \notin X\}$ of attacks that come from an argument of X and attack an argument of $\mathcal{A}'$. Then, the set $E' = \{W \subseteq \mathcal{C}$ and W maximal $|\forall(w_i, w_j), (w_k, w_l) \in W, wi \approx w_k, w_j = w_l, wi, w_k \in X$ and $w_j, w_l \notin X\}$ is a partition of E. The function $f : \mathcal{C}' \to E'$ that associates to each attack $(a', b') \in \mathcal{C}'$ a set of attacks $W \in E'$ with $\forall(w_i, w_j) \in W, w_i \approx a'$ and $w_j = b'$ is surjective.*

Proof. Let us consider $W \in E'$ and an element $(w_i, w_j) \in W$. Then since c' is a core of $\mathcal{AS}$ for $\approx_1$ (resp. $\approx_2$), we have that $\exists! z \in \bar{w}_{i\approx_2} \cap \mathcal{A}'$ (resp. $\bar{w}_{i\approx_1} \cap \mathcal{A}'$). Furthermore, using Proposition 2, we get that $(z, w_j) \in \mathcal{C}'$. $\square$

This proposition means that the modification of the ranking is induced mainly by a quantitative loss. We now introduce the notion of graph isomorphism which will be used to clone our argumentation frameworks.

Definition 10. Let G_1, G_2 be two oriented graphs such that $V(G_1)$ denotes the set of vertices of G_1 and $E(G_1)$ the set of its arcs. We say that $\gamma : V(G_1) \to V(G_2)$ is an isomorphism iff $\forall(x, y) \in E(G_1)$, $(\gamma(x), \gamma(y)) \in E(G_2)$. For simplicity purposes, we will also write $G2 = \gamma(G_1)$.

Using the previous Proposition 8, we can have a better understanding as to why some arguments have better ranking in a core than in $\mathcal{AS}$ with some ranking semantics. This is because arguments of c' that have equivalent arguments in X (for $\approx_2$ or $\approx_1$) have their attacks amplified by those arguments. Of course, depending of the ranking semantics, having more attackers does not always mean that the ranking of the argument is worst. This concept corresponds to the CP postulate defined in [12].

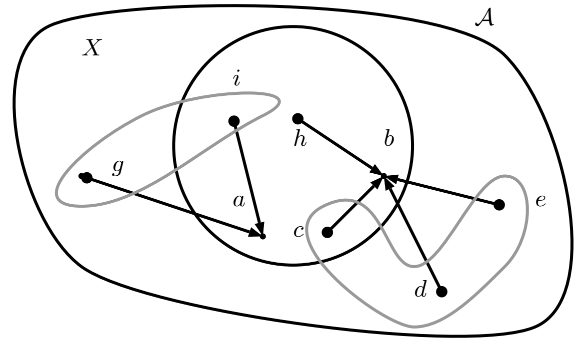


Figure 2: Representation of an argumentation framework $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ and one of its core $c' = (\mathcal{A}', \mathcal{C}')$.

Definition 11. [12] Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF, S a semantics and $\preceq^S_{\mathcal{AS}}$ the ranking obtained after applying S on $\mathcal{AS}$. S satisfy CP iff $\forall a, b \in \mathcal{A}, |Att(a)| < |Att(b)| \Rightarrow b \preceq^S_{\mathcal{AS}} a$ and $a \not\preceq^S_{\mathcal{AS}} b$.

Note that the *burden-based semantics* [2] and the *discussion-based semantics* [2] both satisfy the CP postulate.

We are now interested in the impact of arguments removed by a core on other arguments still inside this core.

Definition 12. Let $\mathcal{AS}$ be an AF and c' one of its core. We denote by $J_{c'}$ (or J if the core is obvious) the set of arguments of the core that have at least one attacker that belongs to X. More precisely, $J = \{a \in \mathcal{A}' | \exists(e, a) \in \mathcal{C}'$ and $f((e, a)) \neq \emptyset\}$.

Example 2. *Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF and $c' = (\mathcal{A}', \mathcal{C}')$ a core of $\mathcal{AS}$ for an equivalence relation. In this example depicted in Figure 2, we have $\mathcal{A} = \{a, b, c, d, e, i, g, h\}$ and $\mathcal{C} = \{(i, a), (g, a), (c, b), (d, b), (e, b), (h, b)\}$. Suppose that $\bar{i} = \{i, g\}$ and $\bar{c} = \{c, d, e\}$. The core c' is such that $\mathcal{A}' = \{a, i, c, b, h\}$. In this case, $J = \{a, b\}$.*

The next proposition states that every argument of the core that is attacked by an argument of X is ranked better in the core.

Proposition 9. *Let $\mathcal{AS}$ be an AF, c' a core of $\mathcal{AS}$ for $\approx_1$ or $\approx_2$, γ an isomorphism s.t. $\mathcal{AS}' = \mathcal{AS} \cup \gamma(c')$, S a ranking that satisfies CP and $\preceq^S_{\mathcal{AS}'}$ the ranking obtained on $\mathcal{AS}'$ using S. Thus, $\forall b \in J, b \preceq^S_{\mathcal{AS}'} \gamma(b)$ and $\gamma(b) \not\preceq^S_{\mathcal{AS}'} b$.*

Proof. Let (a, b) be an attack of c' such that $f((a, b)) \neq \emptyset$. It means that there exists an argument $a' \in X$ such that $(a', b) \in \mathcal{C}$. We thus have $|Att(\gamma(b))| < |Att(b)|$ and since S satisfies CP, $b \preceq^S_{\mathcal{AS}'} \gamma(b)$ and $\gamma(b) \not\preceq^S_{\mathcal{AS}'} b$. $\square$

In Proposition 9, we showed that some arguments of the core may be ranked higher. We now proceed further in this direction by introducing a sufficient condition for characterising the ranking of such arguments.

Proposition 10. *Let $a, b \in J$, if S satisfies CP and $Att(a) - \sum_{e \in Att(a) \cap \mathcal{C}'} |f((e, a))| < Att(b) - \sum_{e \in Att(b) \cap \mathcal{C}'} |f((e, b))|$ then $\gamma(b) \preceq^S_{\mathcal{AS}'} \gamma(a)$ and $\gamma(a) \not\preceq^S_{\mathcal{AS}'} \gamma(b)$.*

PROOF. We have that for all arguments a in $\mathcal{A}'$, $Att(a) - \sum_{e \in Att(a) \cap \mathcal{C}'} |f((e,a))| = |Att(a) \cap \mathcal{A}'|$. Thus, we can say that if $|Att(a) \cap \mathcal{A}'| < |Att(b) \cap \mathcal{A}'|$ then $|Att(\gamma(a))| < |Att(\gamma(b))|$. Since S is a semantics that satisfy CP, $\gamma(b) \preceq^S_{\mathcal{AS}'} \gamma(a)$ and $\gamma(a) \not\preceq^S_{\mathcal{AS}'} \gamma(b)$. $\square$

EXAMPLE 2 (CONT.). *We have that $f((i,a)) = \{(g,a)\}$, $f((c,b)) = \{(d,b),(e,b)\}$ and $f((h,b)) = \emptyset$. Thus, we can compute that $Att(a) - \sum_{e \in Att(a) \cap \mathcal{C}'} |f((e,a))| = 1$ and $Att(b) - \sum_{e \in Att(b) \cap \mathcal{C}'} |f((e,b))| = 4-2 = 2$. We conclude that under a semantics S satisfying CP, $b \preceq^S_{c'} a$ and $a \preceq^S_{c'} b$.*

Unchanged rank. We now give a sufficient condition for an argument to keep the same rank. The basic notion behind this is that arguments that are not attacked by others do not undergo a change in their rank. This is true if the NaE postulate is satisfied, namely if all the non-attacked argument have the same rank.

Definition 13. [12] Let $AS = (\mathcal{A}, \mathcal{C})$ be an AF and S a ranking semantics. S satisfy the NaE postulate iff $\forall a, b \in \mathcal{A}$ s.t. $Att(a) = Att(b) = \emptyset$, we have $a \preceq^S_{\mathcal{AS}} b$ and $b \preceq^S_{\mathcal{AS}} a$.

Note that the *burden-based semantics*, *discussion-based semantics*, *the Categoriser* [8], the *ranking-based semantics SAF* [21], the *Tuples* [15] and *the Matt & Toni semantics* [22] satisfy the NaE postulate.

PROPOSITION 11. *Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF, $a \in \mathcal{A}$ s.t. $Att(a) = \emptyset$, $c' = (\mathcal{A}', \mathcal{C}')$ a core of $\mathcal{AS}$ s.t. $a \in \mathcal{A}'$ and γ an isomorphism s.t. $\mathcal{AS}' = \mathcal{AS} \cup \gamma(c')$. If S is a semantics that satisfies NaE then $a \preceq^S_{\mathcal{AS}'} \gamma(a)$ and $\gamma(a) \preceq^S_{\mathcal{AS}'} a$.*

PROOF. We know that the core c' has fewer arguments and attacks than $\mathcal{AS}$. Thus, the argument a is not attacked either in c' or $\gamma(c')$. Furthermore, since S satisfies NaE, $\gamma(a)$ and a are equivalent. $\square$

Rank decrease. In the next proposition, we introduce a sufficient condition for an argument of the core to have its rank decreased. This condition is true only if the semantics used for the ranking satisfies the CP and SCT postulates. The SCT postulate basically says that if the attackers of an argument b are at least as numerous and acceptable as those of an argument a and either the attackers of b are strictly more numerous or acceptable than those of a, then a is strictly more acceptable than b.

Definition 14. [12] Let $AS = (\mathcal{A}, \mathcal{C})$ be an AF and S a ranking semantics. S satisfy SCT iff $\forall a, b \in \mathcal{A}$ s.t. there is an injective mapping $g : Att(a) \rightarrow Att(b)$ with $\forall a' \in Att(a), a' \preceq^S_{\mathcal{AS}} g(a')$ and ($|Att(a)| < |Att(b)|$ or $\exists a' \in Att(a)$, $a' \preceq^S_{\mathcal{AS}} g(a'), g(a') \not\preceq^S_{\mathcal{AS}} a'$) then $b \preceq^S_{\mathcal{AS}} a$ and $a \not\preceq^S_{\mathcal{AS}} b$.

Note that the *burden-based semantics*, *discussion-based semantics*, *the Categoriser* and the *ranking-based semantics SAF* satisfy the SCT postulate.

The idea behind the next proposition is that if an argument has all of its attackers increase in rank, then its rank is obviously reduced.

PROPOSITION 12. *Let $\mathcal{AS} = (\mathcal{A}, \mathcal{C})$ be an AF, $c' = (\mathcal{A}', \mathcal{C}')$ a core of $\mathcal{AS}$, $a \notin J$ an argument of $\mathcal{A}'$ and $\mathcal{AS}' = \mathcal{AS} \cup \gamma(c')$. If S is a semantics that satisfies CP and SCT and $Att(a) \subseteq J$ then $\gamma(a) \preceq^S_{\mathcal{AS}'} a$ and $a \not\preceq^S_{\mathcal{AS}'} \gamma(a)$.*

PROOF. Since $a \notin J$, we have that $Att(\gamma(a)) = \{\gamma(a') | a' \in Att(a)\}$ and thus $|Att(a)| = |Att(\gamma(a))|$. Now, since $Att(a) \subseteq J$, we have that $\forall b \in Att(b), b \preceq^S_{\mathcal{AS}'} \gamma(b)$ and $\gamma(b) \not\preceq^S_{\mathcal{AS}'} b$ (using Proposition 9). Finally, using the SCT postulate, we conclude that $\gamma(a) \preceq^S_{\mathcal{AS}'} a$ and $a \not\preceq^S_{\mathcal{AS}'} \gamma(a)$. $\square$

5. DISCUSSION

The work presented in this paper is of direct interest for logical-founded agent interaction since logical instantiations of ranking semantics is one of the most recent promising avenues for argumentation and have been little addressed in literature.

Classically, logical argumentation allows for a more practical and application-oriented use of argumentation theory. However, one usual caveat of such a framework is the appearance of a large number of redundant arguments. In this paper, we used the notion of core to reduce the size of argumentation graphs, and study whether this simplification has an impact on the total order over the set of arguments outputted by a particular class of ranking semantics. More precisely, we first gave two notions of core based on equivalence and/or equality of arguments' supports and conclusions in the context of the existential rules fragment. We provided theoretical results extending the state of the art regarding core equivalence in the context of logical argumentation framework. We then studied how ranking semantics behaved w.r.t. notions of core and we showed that, depending on the used notion of core, the obtained argument ranking can be different. Finally, in light of this result, we introduced the notion of core-induced modification and gave a characterisation for semantics satisfying several postulates.

Our contribution is the first approach that formally study the impact of the notion of core on ranking semantics. As such, several avenues are contemplated. In particular, while we showed that depending of the core, the obtained rankings can be different, we envision to study the effect of ranking semantics on cores coming from new and existing equivalence relations in the literature. Our approach, using the ARL as general logical setting, is general enough to capture various rankings semantics via the postulate satisfaction. For instance, the Categoriser and Social Abstract Argumentation both satisfy SCT but neither satisfy CP thus we cannot characterise the ranking decrease. Categoriser and Social Abstract Argumentation satisfy NAE, thus we can characterise such arguments w.r.t. unchanged rank.

In our future work, we plan to exhaustively cover all postulates in [12] in order to complete the change landscape.

We also plan to investigate how acceptance of ranking based semantics for existential rules can be applied practically for OBDA related applications. Following previous work on classic semantics [26] we are interested in providing reasoning workflows in practice.

Finally, while this work focuses on existential rules we can envisage this work in a greater context where ranking semantics could be used for a finer acceptability notion akin to human reasoning [10].

REFERENCES

[1] L. Amgoud. Postulates for logic-based argumentation systems. *Int. J. Approx. Reasoning*, 55(9):2028–2048, 2014.

[2] L. Amgoud and J. Ben-Naim. Ranking-Based Semantics for Argumentation Frameworks. In *Scalable Uncertainty Management - 7th International Conference, SUM 2013, Washington, DC, USA, September 16-18, 2013. Proceedings*, pages 134–147, 2013.

[3] L. Amgoud and J. Ben-Naim. Argumentation-based Ranking Logics. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey, May 4-8, 2015*, pages 1511–1519, 2015.

[4] L. Amgoud, P. Besnard, and S. Vesic. Equivalence in logic-based argumentation. *Journal of Applied Non-Classical Logics*, 24(3):181–208, 2014.

[5] A. Arioua and M. Croitoru. A dialectical proof theory for universal acceptance in coherent logic-based argumentation frameworks. In *ECAI 2016: 22nd European Conference on Artificial Intelligence*, volume 285, pages 55–63. IOS Press, 2016.

[6] J.-F. Baget, F. Garreau, M.-L. Mugnier, and S. Rocher. Revisiting Chase Termination for Existential Rules and their Extension to Nonmonotonic Negation. *CoRR*, abs/1405.1071, 2014.

[7] J.-F. Baget, M. Leclère, M.-L. Mugnier, and E. Salvat. On rules with existential variables: Walking the decidability line. *Artif. Intell.*, 175(9-10):1620–1654, 2011.

[8] P. Besnard and A. Hunter. A logic-based theory of deductive arguments. *Artif. Intell.*, 128(1-2):203–235, 2001.

[9] P. Besnard and A. Hunter. *Elements of Argumentation.* MIT Press, 2008.

[10] P. Bisquert, M. Croitoru, F. D. de Saint-Cyr, and A. Hecham. Formalizing cognitive acceptance of arguments: Durum wheat selection interdisciplinary study. *Mind and Machines (to appear)*, 2016.

[11] A. Bondarenko, F. Toni, and R. A. Kowalski. An Assumption-Based Framework for Non-Monotonic Reasoning. In *LPNMR*, pages 171–189, 1993.

[12] E. Bonzon, J. Delobelle, S. Konieczny, and N. Maudet. A Comparative Study of Ranking-Based Semantics for Abstract Argumentation. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 914–920, 2016.

[13] A. Calì, G. Gottlob, and T. Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. In *PODS 2009*, pages 77–86, 2009.

[14] M. Caminada and L. Amgoud. An Axiomatic Account of Formal Argumentation. In *Proceedings, The Twentieth National Conference on Artificial Intelligence and the Seventeenth Innovative Applications of Artificial Intelligence Conference, July 9-13, 2005, Pittsburgh, Pennsylvania, USA*, pages 608–613, 2005.

[15] C. Cayrol and M.-C. Lagasquie-Schiex. Graduality in Argumentation. *J. Artif. Intell. Res. (JAIR)*, 23:245–297, 2005.

[16] M. Croitoru, R. Thomopoulos, and S. Vesic. Introducing Preference-Based Argumentation to Inconsistent Ontological Knowledge Bases. In *PRIMA 2015: Principles and Practice of Multi-Agent Systems*, pages 594–602, 2015.

[17] M. Croitoru and S. Vesic. What Can Argumentation Do for Inconsistent Ontology Query Answering? In *SUM 2013*, pages 15–29, 2013.

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

[19] A. J. García and G. R. Simari. Defeasible Logic Programming: An Argumentative Approach. *TPLP*, 4(1-2):95–138, 2004.

[20] G. Gottlob, T. Lukasiewicz, and A. Pieris. Datalog+/-: Questions and Answers. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014*, 2014.

[21] J. Leite and J. Martins. Social Abstract Argumentation. In *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 2287–2292, 2011.

[22] P.-A. Matt and F. Toni. A Game-Theoretic Measure of Argument Strength for Abstract Argumentation. In *Logics in Artificial Intelligence, 11th European Conference, JELIA 2008, Dresden, Germany, September 28 - October 1, 2008. Proceedings*, pages 285–297, 2008.

[23] S. Modgil and H. Prakken. The ASPIC+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.

[24] M. Thomazo. Compact Rewritings for Existential Rules. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, pages 1125–1131, 2013.

[25] M. Thomazo and S. Rudolph. Mixing Materialization and Query Rewriting for Existential Rules. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014)*, pages 897–902, 2014.

[26] B. Yun and M. Croitoru. An Argumentation Workflow for Reasoning in Ontology Based Data Access. In *Computational Models of Argument - Proceedings of COMMA 2016, Potsdam, Germany, 12-16 September, 2016.*, pages 61–68, 2016.

[27] H. Zhang, Y. Zhang, and J.-H. You. Expressive Completeness of Existential Rule Languages for Ontology-Based Query Answering. In *IJCAI 2016*, pages 1330–1337, 2016.