On Computational Problems for Infinite Argumentation Frameworks: Classifying Complexity via Computability*

Uri Andrews[1,†], Luca San Mauro[2,*,†]

[1]Department of Mathematics, University of Wisconsin, USA
[2]Department of Philosophy, University of Bari, Italy

Abstract
This paper investigates infinite argumentation frameworks. We introduce computability theoretic machinery as a robust method of evaluating, in the infinite setting, the complexity of the main computational issues arising from admissible, complete, and stable semantics: in particular, for each of these semantics, we measure the complexity of credulous and skeptical acceptance of arguments, and that of determining existence and uniqueness of extensions. We also propose a way of using Turing degrees to classify, for a given infinite argumentation framework, the exact difficulty of computing an extension in a given semantics and show that these problems give rise to a rich class of complexities.

Keywords
infinite argumentation frameworks, computability theory, complexity, admissible extensions, stable extensions, complete extensions, Turing degrees

1. Introduction

Abstract argumentation theory is a fundamental research area in AI, providing a powerful paradigm for reasoning about knowledge representation and multi-agent systems. Historically, the focus has predominantly been on finite argumentation frameworks (AFs), leaving the infinite case relatively unexplored. As noted in [1], this oversight poses significant theoretical, conceptual, and practical limitations.

Firstly, infinite frameworks align naturally with Dung's seminal approach [2], whose results do not presuppose finiteness. Secondly, representing argumentation scenarios in an infinite manner captures the inherently nonmonotonic nature of argumentation, where arguments can always be challenged by the emergence of new information, making any fixed limit on the space arguments or attacks somewhat artificial. Thirdly, infinite AFs often arise in practical contexts, such as logic programming [3] and the logical analysis of multi-agent or distributed systems [4]

SAFA'24: Fifth International Workshop on Systems and Algorithms for Formal Argumentation, September 17, 2024, Hagen, Germany

*An expanded version of this article will be submitted to NMR 2024 and FCR 2024.

*Corresponding author.
†These authors contributed equally.
andrews@math.wisc.edu (U. Andrews); luca.sanmauro@gmail.com (L. San Mauro)
https://math.wisc.edu/~andrews (U. Andrews); https://www.lucasanmauro.com/ (L. San Mauro)

(the substantial introduction of [1] provides other concrete examples of applications of infinite AFs, e.g., to multiagent negotiations).

Fortunately, recent years have seen a growing interest in infinite AFs, with special focus on how the existence and interplay of various semantics—well-understood for finite AFs—are affected in the infinite realm (see, e.g., [5, 6, 7, 8, 9]). This increasing recognition underscores the importance of infinite AFs for a broad understanding of argumentation theory.

However, the literature still lacks a comprehensive framework for systematically exploring all logical aspects of infinite AFs, particularly regarding their core computational issues. A significant research avenue in finite AFs has been determining the algorithmic complexity of tasks associated with finding acceptable collections of arguments (up to suitable collection of semantics), with numerous complexity theoretic results highlighting their inherent computational difficulty (see, e.g., [10, 11, 12]). To our knowledge, no analog study has been conducted for infinite AFs.

This paper addresses this gap by initiating a systematic study of the complexity of computational problems in infinite AFs. For this endeavor, we bring into the subject of argumentation theory the machinery of computability theory, which may be regarded as an infinitary companion of computational complexity theory and abounds with concepts and hierarchies for measuring the complexity of computing or defining countably infinite objects.

The application of computability theoretic tools outside of mathematical logic is a well-established idea. Over the past decades, computability theory has been applied to a wide array of mathematical disciplines, and computability theoretic concepts have found applications in other formal subjects, such as theoretical computer science, economics, and linguistics (see, e.g., [13, 14, 15]).

The present paper, we argue, provides compelling evidence of the benefits of viewing infinite AFs through a computability theoretic lens. We assess the complexity of many computational problems—both established and novel—within our framework, illustrating their undecidability while providing precise measures of their complexity.

Organization of the paper

Section 2 briefly reviews the main semantic concepts from argumentation theory that are relevant to this paper, along with the fundamental computational problems associated with them. In Section 3, we introduce the key notions of computability theory employed in the work and we define the concept of computable AFs and the computational issues emerging from it. Finally, in Section 4, we provide lower bounds for the complexity of our computational problems. Our results are collected in Table 2.

2. Argumentation theoretic background

To keep our paper self-contained, we now briefly review some key concepts of Dung-style argumentation theory, focusing on the semantics notions considered in this paper and the fundamental computational problems associated with them (the surveys [16, 17] offer an overview of these topics).

An *argumentation framework* (AF) $\mathcal{F}$ is a pair $(A_\mathcal{F}, R_\mathcal{F})$ consisting of a set $A_\mathcal{F}$ of arguments and an attack relation $R_\mathcal{F} \subseteq A_\mathcal{F} \times A_\mathcal{F}$. If some argument a attacks some argument b, we may write $a \rightarrowtail b$ instead of $(a, b) \in R_\mathcal{F}$. Collections of arguments $S \subseteq A_\mathcal{F}$ are called *extensions*. For an extension S, the symbols S^+ and S^- denote, respectively, the arguments that S attacks and the arguments that attack S:

$$S^+ = \{x : (\exists y \in S)(y \rightarrowtail x)\};$$
$$S^- = \{x : (\exists y \in S)(x \rightarrowtail y)\}.$$

S *defends* an argument a, if any argument that attacks a is attacked by some argument in S (i.e., $\{a\}^- \subseteq S^+$). The *characteristic function* of $\mathcal{F}$ is the following mapping $f_\mathcal{F}$ which sends subsets of $A_\mathcal{F}$ to subsets of $A_\mathcal{F}$:

$$f_\mathcal{F}(S) := \{x : \ x \text{ is defended by } S\}.$$

All AFs investigated in this paper are infinite.

A *semantics* σ assigns to every AF $\mathcal{F}$ a set of extensions $\sigma(\mathcal{F})$ which are deemed as acceptable. A huge number of semantics, fueled by different motivations, have been proposed and analyzed. Here, we focus on three prominent choices, whose computational aspects are well-understood in the finite setting: admissible, complete, and stable semantics (abbreviated by *ad*, *co*, *stb*, respectively).

Let $\mathcal{F} = (A_\mathcal{F}, R_\mathcal{F})$ be an AF. Denote by *cf*$(\mathcal{F})$ the collection of extensions of $\mathcal{F}$ which are *conflict-free* (i.e., $S \in$ *cf*$(\mathcal{F})$ iff $a \not\rightarrowtail b$, for all $a, b \in S$). Then, for $S \in$ *cf*$(\mathcal{F})$,

- $S \in$ *ad*$(\mathcal{F})$ iff S is self-defending (i.e., $S \subseteq f_\mathcal{F}(S)$);
- $S \in$ *co*$(\mathcal{F})$ iff S is a fixed point of f_F (i.e., $S = f_\mathcal{F}(S)$);
- $S \in$ *stb*$(\mathcal{F})$, iff S attacks all arguments outside of it (i.e., $S^+ = A_\mathcal{F} \setminus S$).

In discussing the complete extensions, we will also briefly mention the grounded extension, which is the unique smallest fixed point of f_F; in any AF, the grounded extension always exists [2, Theorem 3].

For a given semantics σ, the following are well-known computational problems related to σ:

- Cred$_\sigma$ (for *credulous* acceptance) is the decision problem whose accepting instances are the pairs $(\mathcal{F}, a)$ so that $a \in S$ for *some* $S \in \sigma(\mathcal{F})$;
- Skept$_\sigma$ (for *skeptical* acceptance) is the decision problem whose accepting instances are the pairs $(\mathcal{F}, a)$ so that $a \in S$ for *all* $S \in \sigma(\mathcal{F})$;
- Exist$_\sigma$ is the decision problem whose accepting instances are the AFs $\mathcal{F}$ so that $\sigma(\mathcal{F}) \neq \emptyset$;
- NE$_\sigma$ is the decision problem whose accepting instances are the AFs $\mathcal{F}$ so that $\sigma(\mathcal{F}) \setminus \{\emptyset\} \neq \emptyset$;
- Uni$_\sigma$ is the decision problem whose accepting instances are the AFs $\mathcal{F}$ so that $|\sigma(\mathcal{F})| = 1$.

In formal argumentation theory, evaluating the computational complexity of the aforementioned problems for various semantics has been a noteworthy research thread for more than 20 years [17]. Table 1 collects known complexity results for the admissible, stable, and complete semantics. This analysis refers only to finite AFs. In the next section, we introduce our computability theoretic perspective that allows us to tackle complexity issues concerning infinite AFs.

σ	Cred_σ	Skept_σ	Exist_σ	NE_σ	Uni_σ
ad	**NP**-c	trivial	trivial	**NP**-c	**coNP**-c
stb	**NP**-c	**coNP**-c	**NP**-c	**NP**-c	**DP**-c
co	**NP**-c	**P**-c	trivial	**NP**-c	**coNP**-c

Table 1
Computational problems for finite AFs. $\mathcal{C}$-c denotes completeness for the class $\mathcal{C}$.

3. Computational problems for AFs through the lens of computability theory

In this section, we introduce computable AFs and we revisit the computational problems of the last section through the lens of computability theory. We aim at conveying the main ideas without delving into too many technical details. A more formal and comprehensive exposition of the fundamentals of computability theory can be found, e.g., in [18, 19]. We begin by establishing standard notation and terminology for some combinatorial notions that appear frequently in our proofs.

3.1. Sequences, strings, and trees

As is common in computability theory, we denote the set of natural numbers by ω. Since there is no risk of ambiguity, we simply refer to the elements of ω as numbers. The symbol ω^ω denotes the set of all functions from ω to ω. For our purposes, it is convenient to represent elements of ω^ω as infinite sequences of numbers; we denote by 0^∞ the infinite sequence consisting of only 0's (or, equivalently, the constant function to 0). The restriction of an infinite sequence $\pi \in \omega^\omega$ to its first n-many bits is denoted by $\pi \upharpoonright_n$.

We use standard notation and terminology about *strings*: The set of all finite strings of numbers is denoted by $\omega^{<\omega}$. The symbol λ denotes the empty string. The concatenation of strings σ, τ is denoted by $\sigma^\frown\tau$. The length of a string σ is denoted by $|\sigma|$. If there is ρ so that $\sigma^\frown\rho = \tau$, we say that σ is a *prefix* of τ and we write $\sigma \preceq \tau$. Similarly, if $\pi \in \omega^{<\omega}$ and $\sigma = \pi \upharpoonright_n$ for some n, we write $\sigma \prec \pi$.

In order to formulate our problems as subsets of ω, it will be convenient to encode pairs of numbers into single numbers. The pairing function does this. Fix $p : \omega \times \omega \to \omega$ to be a computable bijection. We adopt the common habit of denoting $p(x, y)$ by $\langle x, y\rangle$.

The encodings discussed in Section 4 heavily rely on the difficulty of calcuting paths through trees. As is common in computability theory, we say that a *tree* is a set $\mathcal{T} \subseteq \omega^{<\omega}$ closed under prefixes. We picture trees growing upwards, with $\sigma^\frown i$ to the left of $\sigma^\frown j$, whenever $i < j$. A *path* $\pi \in \omega^\omega$ through a tree $\mathcal{T} \subseteq \omega^{<\omega}$ is an infinite sequence so that $\pi \upharpoonright_n \in \mathcal{T}$, for all numbers n. The set of paths through a tree $\mathcal{T}$ is denoted by $[\mathcal{T}]$. $\mathcal{T}$ is *well-founded* if $[\mathcal{T}] = \emptyset$ and otherwise is *ill-founded*. Note that we follow the standard terminology in computability theory requiring that paths be infinite. Indeed, if one were to allow paths to be finite, then these notions trivialize, since one could computably find a path through any given computable tree. For example, the set of strings

$$\mathcal{T} := \{\lambda\} \cup \{\sigma, \sigma^\frown 1 : (\forall n < |\sigma|)(\sigma(n) = 0)\}$$

is an ill-founded tree with $[\mathcal{T}] = \{0^\infty\}$. If $\mathcal{T}$ contains strings of arbitrary length, then $\mathcal{T}$ has *infinite height*. Note that there are trees of infinite height which are well-founded, e.g., $\mathcal{T} = \{n^\frown\sigma : |\sigma| \leq n\}$.

3.2. Computable argumentation frameworks

A basic problem that one encounters when attempting to calibrate the algorithmic complexity of infinite AFs is that of describing infinite objects in a finitary way. Computability theory offers a wide range of tools designed for this endeavour. Here, we will concentrate on AFs that are *computably presentable*, in the sense that there are Turing machines (or, equivalently, modern computer programs) that, in finitely many steps, decide whether a given pair of arguments belongs to the attack relation.

Notation. Let $(\Phi_e)_{e\in\omega}$ be a uniformly computable enumeration of all computable functions from ω to $\{0, 1\}$.

Definition 3.1. *A number e is a* computable index for an AF $\mathcal{F} = (A_\mathcal{F}, R_\mathcal{F})$, *if there is a computable bijection $f : \omega \to A_\mathcal{F}$ so that*

$$\Phi_e(\langle x, y\rangle) = \begin{cases} 1 & \textit{if } f(x) \rightarrowtail f(y) \\ 0 & \textit{otherwise.} \end{cases}$$

An AF $\mathcal{F}$ is computably presented, *if it has a computable index $e \in \omega$.*

We use the notation $\mathcal{F}_e$ to refer to the AF with computable index e (note that every computable AF possesses infinitely many computable indices.). We let a_n refer to the element of $A_{\mathcal{F}_e}$ given by $f(n)$.

Remark 3.2. *The collection of computable indices for AFs just defined is noncomputable (in particular, any index e for a non-total computable function Φ_e cannot be a computable index for an argumentation framework). There are alternative indexings that circumvent this issue; yet, adopting another indexing would not alter the complexity of the computational problems we analyze, though it would make the proofs slightly more cumbersome. Hence, we opt for the simplicity of Definition 3.1.*

The benefit of dealing with computable AFs is that the complexity of the decision problems associated with them do not arise due to complexity of the argumentation framework itself, but rather reflects the inherent complexity of the decision problem. Further, the computational problems associated with computable AFs can be naturally represented as subsets of ω, which are suitable to be classified by computability theoretic means:

Definition 3.3. *For a semantics σ:*

1. $\mathit{Cred}_\sigma^\infty := \{\langle e, a\rangle \in \omega : (\exists S \in \sigma(\mathcal{F}_e))(a \in S)\}$;
2. $\mathit{Skept}_\sigma^\infty := \{\langle e, a\rangle \in \omega : (\forall S \in \sigma(\mathcal{F}_e))(a \in S)\}$;

3. *Exist*$_\sigma^\infty := \{e : (\exists S \subseteq A_{\mathcal{F}_e}))(S \in \sigma(\mathcal{F}_e))\}$;

4. *NE*$_\sigma^\infty := \{e : (\exists S \in \sigma(\mathcal{F}_e))(S \neq \emptyset)\}$;

5. *Uni*$_\sigma^\infty := \{e : (\exists! S \subseteq A_{\mathcal{F}_e})(S \in \sigma(\mathcal{F}_e))\}$.

We also introduce new semantics which make sense only in the infinite setting. This is motivated by the idea that, given an infinite AF, we might hope for our accepted sets to give us infinitely much information:

1. $S \in$ *infad*$(\mathcal{F})$ if and only if $S \in$ *ad*$(\mathcal{F})$ and S is infinite;
2. $S \in$ *infco*$(\mathcal{F})$ if and only if $S \in$ *co*$(\mathcal{F})$ and S is infinite;
3. $S \in$ *infstb*$(\mathcal{F})$ if and only if $S \in$ *stb*$(\mathcal{F})$ and S is infinite.

In a sense, these new semantics give a measure of how much conflict lies in a given AF. For example, if *infad*$(\mathcal{F}) = \emptyset$ for an AF $\mathcal{F}$, then the size of any of its admissible extensions are negligible in comparison to the size of $\mathcal{F}$, suggesting that the attack relation in $\mathcal{F}$ prevents simultaneously accepting any significant fraction of $\mathcal{F}$-arguments.

As an illustration of why we might want to accept only infinite extensions, we consider that a given infinite AF may contain a single argument b so that b attacks every other argument, and every other argument attacks b. We imagine that b is a statement of extreme solipsism denying the truth of any other statement. While $\{b\}$ is a stable extension, it represents a negligible fraction of arguments, and we may prefer not to accept it. In an infinite AF, any finite set is as negligible as $\{b\}$, so we may prefer to accept only infinite extensions.

The complexity classes that most naturally match the problems of Definition 3.3 are those of the Σ_1^1 and Π_1^1 sets. The Σ_1^1 sets are formally defined as those subsets of ω that are definable in the language of second-order arithmetic using a single second-order existential quantifier ranging over subsets of ω followed by number quantifiers and the first order functions and relations $(+, \cdot, <, 0, 1, \in)$; for more details, see [18, §16]. Π_1^1 sets are the complements of Σ_1^1 sets.

Proposition 3.4. *For $\sigma \in \{ad, stb, co, infad, infstb, infco\}$, Cred$_\sigma^\infty$, Exist$_\sigma^\infty$, NE$_\sigma^\infty$, are Σ_1^1.*

Proof. We first consider $\sigma \in \{ad, stb, co\}$. To define $\mathrm{Cred}_\sigma^\infty$, we see from Definition 3.3: $\mathrm{Cred}_\sigma^\infty := \{\langle e, a\rangle \in \omega : (\exists S \in \sigma(\mathcal{F}_e)(a \in S))\}$ uses a single existential quantifier over sets S. This is similarly true for the definitions of $\mathrm{Exist}_\sigma^\infty$ and $\mathrm{NE}_\sigma^\infty$ in Definition 3.3. Thus, it suffices to see that the condition $S \in \sigma(\mathcal{F}_e)$ can be defined with only quantification over arguments (which are encoded as numbers), not needing quantification over sets of arguments. Note that the definition of S^+ and S^- uses only quantifiers over arguments. Thus, the definition of $f_{\mathcal{F}}(S)$ given by $a \in f_{\mathcal{F}}(S)$ if and only if $\{a\}^- \subseteq S^+$ uses only quantifiers over arguments. Finally, $S \in$ *ad*$(\mathcal{F}), S \in$ *stb*$(\mathcal{F}), S \in$ *co*$(\mathcal{F})$ are all defined from $f_{\mathcal{F}}(S)$ and S^+ using only quantifiers over arguments.

In the case of $\sigma \in \{infad, infstb, infco\}$, we need to also observe that S being infinite is defined via $\forall n \exists m (a_m \in S \wedge m > n)$, which uses only quantifiers over numbers. □

Proposition 3.5. *For* $\sigma \in \{ad, stb, co, infad, infstb, infco\}$*, Skept*$^\infty_\sigma$ *is* Π^1_1 *and, for* $\sigma \in \{ad, co\}$*, Uni*$^\infty_\sigma$ *is* Π^1_1*.*

Proof. The definition of Skept$^\infty_\sigma$ in Definition 3.3 uses a single universal set-quantifier followed by only number quantifiers in the definition of $\sigma(\mathcal{F}_e)$.

For $\sigma \in \{ad, co\}$, $e \in$ Uni$^\infty_\sigma$ if and only if there are not two different σ extensions (as there is always at least one σ extension). This is defined by the negation of the following formula:

$$(\exists S_1 \exists S_2)(\exists x \in S_1 \setminus S_2) \wedge S_1 \in \sigma(\mathcal{F}_e) \wedge S_2 \in \sigma(\mathcal{F}_e)).$$

Note that $\exists S_1 \exists S_2$ can be replaced by a single existential quantifier by encoding the pair (S_1, S_2) as a single set $\{\langle 1, x\rangle : x \in S_1\} \cup \{\langle 2, y\rangle : y \in S_2\}$. This shows that Uni$^\infty_\sigma$ is the complement of a Σ^1_1 set, thus is Π^1_1. □

Remark 3.6. The above argument does not suffice to show that Uni$^\infty_{stb}$ is also Π^1_1, since some AFs have no stable extension. The most obvious definition says there exists one stable extension and there does not exist two, which gives a definition which is a conjunction of a Σ^1_1 and a Π^1_1 condition, i.e., a so-called d-Σ^1_1 definition. This is analogous to the fact that in the finite case Uni$_{stb}$ is **DP**-complete. Similarly, the argument above does not show that Uni$^\infty_\sigma$ is Π^1_1 for $\sigma \in \{infad, infstb, infco\}$. It is true that Uni$^\infty_\sigma$ is Π^1_1 for $\sigma \in \{stb, infad, infstb, infco\}$, but we will not include a proof in this paper.

We note that knowing that a problem is Σ^1_1 does not necessarily mean the problem is complicated. This only gives an upper bound for its complexity. Sometimes, a simpler definition is achievable. As an example, we consider Cred$_{cf} := \{\langle a, e\rangle : (\exists S \in cf(\mathcal{F}_e))(a \in S)\}$. Though the given definition is Σ^1_1, to know if an argument a belongs to a conflict-free extension of $\mathcal{F}_e$, it suffices to check whether a is non-self-defeating, i.e., $a \not\rightarrow a$, which is equivalent to checking the computable fact that $\Phi_e(\langle f^{-1}(a), f^{-1}(a)\rangle) = 0$. In contrast, we will show that for the computational problems associated to the admissible, stable, and complete semantics, the use of the quantifier ranging over all sets cannot be avoided.

We will heavily rely on the following fundamental theorem by Kleene which offers a natural way of representing Σ^1_1 sets:

Theorem 3.7 (Kleene [20])**.** *A set* $X \subseteq \omega$ *is* Σ^1_1 *if and only if there is a computable sequence of computable trees* $(\mathcal{T}^X_n)_{n\in\omega}$ *so that* $n \in X$ *iff* $\mathcal{T}^X_n$ *is ill-founded.*

We call $(\mathcal{T}^X_n)_{n\in\omega}$ a *tree-sequence* for X. As a corollary of Kleene's theorem, one obtains that the problem of deciding which computable trees in $\omega^{<\omega}$ are ill-founded (or well-founded) is as hard as any other Σ^1_1 (resp., Π^1_1) problem.

Theorem 3.7 gives a reason to consider the Σ^1_1 sets as the natural infinite analogs of the **NP** problems. Namely, given an ill-founded computable tree $\mathcal{T}$ and a sequence π which is a path through $\mathcal{T}$, it's relatively simple to check that $\pi \in [\mathcal{T}]$ (it requires checking infinitely many simple facts: $\pi \upharpoonright_n \in \mathcal{T}$, for each n), but finding a sequence $\pi \in [T]$—or even knowing whether there exists a sequence $\pi \in [T]$—is a far harder problem.

Our main goal is to exactly characterize the complexity of the computational problems described in Definition 3.3. To do so, we need to show that they are complete for their respective complexity classes. The following definition formalizes this notion.

Definition 3.8. *Let Γ be a complexity class (e.g., $\Gamma \in \{\Sigma^1_1, \Pi^1_1\}$). A set $V \subseteq \omega$ is* Γ-hard*, if for every $X \in \Gamma$ there is a computable function $f : \omega \to \omega$ so that $x \in X$ if and only if $f(x) \in V$. If V is Γ-hard and belongs to Γ, then it is* Γ-complete.

Proposition 3.9. *It follows from Theorem 3.7 that the set of indices for ill-founded computable trees is a Σ^1_1-complete set. Similarly, the set of indices for well-founded computable trees is a Π^1_1-complete set.*

The following result is far less obvious, but will be useful below to examine Uni^∞_σ.

Theorem 3.10 ([21, Theorem 18.11])**.** *The set UB of indices for computable trees with exactly one path is a Π^1_1-complete set.*

Remark 3.11. The hardness in Theorem 3.10 is quite easy. We can reduce the question of whether a tree $\mathcal{T}$ is well-founded to whether a tree $\mathcal{T}'$ has two paths, where $\mathcal{T}'$ always has at least one path, by simply giving $\mathcal{T}'$ one more path than $\mathcal{T}$ (e.g. $\mathcal{T}' = \{1^\frown\sigma : \sigma \in \mathcal{T}\} \cup \{\sigma : (\forall n < |\sigma|)\sigma(n) = 0\}$). The fact that UB is itself Π^1_1 is the subtle part of Theorem 3.10.

Theorem 3.7 along with Definition 3.8 suggest a natural approach for gauging the complexity of the computational problems of Definition 3.3. Namely, given another Σ^1_1 (or Π^1_1) set X, we translate the question asking whether $n \in X$ to the question of if the tree $\mathcal{T}^X_n$ is ill-founded (resp., well-founded), and then we need to computably find an instance of our computational problem which should be accepted if and only if $\mathcal{T}^X_n$ is ill-founded (resp., well-founded). This involves coding a tree, or more precisely, the collection of paths through a tree into the σ extensions in an argumentation framework. We do exactly this in Section 4.

Table 2 collects our results regarding complexities of the computational problems examined for computable argumentation frameworks.

σ	$\text{Cred}^\infty_\sigma$	$\text{Skept}^\infty_\sigma$	$\text{Exists}^\infty_\sigma$	NE^∞_σ	Uni^∞_σ
ad	Σ^1_1-c 4.4,3.4	trivial	trivial	Σ^1_1-c 4.4 3.4	Π^1_1-c 4.5,3.5
stb	Σ^1_1-c 4.4,3.4	Π^1_1-c 4.6,3.5	Σ^1_1-c 4.4, 3.4	Σ^1_1-c 4.4,3.4	Π^1_1-c 4.5, †
co	Σ^1_1-c 4.4, 3.4	Π^1_1-c $*$, 3.5	trivial	Σ^1_1-c 4.4, 3.4	Π^1_1-c 4.5,3.5
infad	Σ^1_1-c 4.4,3.4	Π^1_1-c 4.7,3.5	Σ^1_1-c 4.4, 3.4	Σ^1_1-c 4.4, 3.4	Π^1_1-c 4.5,†
infstb	Σ^1_1-c 4.4,3.4	Π^1_1-c 4.6,3.5	Σ^1_1-c 4.4, 3.4	Σ^1_1-c 4.4, 3.4	Π^1_1-c 4.5, †
infco	Σ^1_1-c 4.4,3.4	Π^1_1-c 4.6,3.5	Σ^1_1-c 4.4, 3.4	Σ^1_1-c 4.4, 3.4	Π^1_1-c 4.5, †

Table 2
Computational problems for computable AFs. $\mathcal{C}$-c denotes completeness for the class $\mathcal{C}$. The numbers in each cell of the table refer to the Theorem number providing the lower bound and upper bounds for the result in that cell. The asterisk in the Skept^∞_{co} cell reflects that this lower bound is not proved in this paper. Rather, the Π^1_1-hardness for Skept^∞_{co} is deferred to future work focusing on the grounded semantics. Note that $\langle e, a\rangle \in \text{Skept}^\infty_{co}$ if and only if a is in the unique grounded extension in $\mathcal{F}_e$. Similarly, the dagger in several Uni^∞_σ cells reflect that the upper bounds in these cases are not proved in this paper. (See Remark 3.6).

Remark 3.12. As noted before, the Σ^1_1 sets are natural analogs in the infinitary setting of the **NP** sets, and the Π^1_1 sets are the natural analogs of the **coNP** sets. With the exception of Skept^∞_{co}

and $\text{Uni}^{\infty}_{stb}$, Table 2 follows this translation from Table 1 for the first three rows. These two results mark surprising differences in the infinite setting.

The trivial entries are due to the fact that $\emptyset$ is always an admissible extension and the grounded extension is always a complete extension.

3.3. Spectra of σ extensions

We propose a way to more fully understand the complexity of the problem of finding a σ extension in a given AF $\mathcal{F}$.

Definition 3.13. *For each $e \in \omega$ and semantics σ, let $Spec^{\neg\emptyset}_{\sigma}(\mathcal{F}_e)$ be the set of Turing degrees of non-empty sets $X \subseteq \omega$ so that $\{a_n : n \in X\}$ is a σ extension in $\mathcal{F}_e$.*

The notion $\text{Spec}^{\neg\emptyset}_{\sigma}(\mathcal{F}_e)$ exactly captures the difficulty of computing a non-empty σ extension in $\mathcal{F}_e$. We will be relating the problem of computing a σ extension in $\mathcal{F}_e$ to the problem of finding a path through a particular tree. So, we define the analogous notion of the spectrum of a tree.

Definition 3.14. *Given any computable tree $\mathcal{T}$, we let Spec($\mathcal{T}$) be set of Turing degrees of paths $X \in [\mathcal{T}]$.*

Our main result in this direction is the following:

Theorem 3.15. *For $\sigma \in \{ad, stb, co, infad, infstb, infco\}$ and for any computable tree $\mathcal{T}$, there exists a computable AF $\mathcal{F}_e$ so that $Spec^{\neg\emptyset}_{\sigma}(\mathcal{F}_e) = Spec(\mathcal{T})$.*

When $\sigma \in \{ad, stb, infad, infstb\}$, future work will show the converse, namely that for every e, there is a computable tree so that $\text{Spec}^{\neg\emptyset}_{\sigma}(\mathcal{F}_e) = \text{Spec}(\mathcal{T})$. Table 3 collects our results on Spectra of extensions.

σ	$\text{Spec}^{\neg\emptyset}_{\sigma}$
ad	Exactly Spec($\mathcal{T}$)
stb	Exactly Spec($\mathcal{T}$)
co	Any Spec($\mathcal{T}$)
infad	Exactly Spec($\mathcal{T}$)
infstb	Exactly Spec($\mathcal{T}$)
infco	Any Spec($\mathcal{T}$)

Table 3
In this paper, we show that for any computable tree $\mathcal{T}$, there is a computable argumentation framework $\mathcal{F}_e$ so that $\text{Spec}(\mathcal{T}) = \text{Spec}^{\neg\emptyset}_{\sigma}(\mathcal{F}_e)$. When $\sigma \in \{ad, stb, infad, infstb\}$, future work will show the converse. Namely that for every e, there is a computable tree so that $\text{Spec}^{\neg\emptyset}_{\sigma}(\mathcal{F}_e) = \text{Spec}(\mathcal{T})$. We do not know how to attain a corresponding upper bound for the complete or infinite complete cases.

We now discuss some consequences of these characterizations on the problem of, given a computable argumentation framework, computing some σ extension. The hyperarithmetical sets are, in a very general sense, considered the collection of constructible subsets of the natural numbers. Formally, a set is hyperarithmetical if and only if it is both Σ^1_1 and Π^1_1. The hyperarithmetical degrees are particularly useful as a yardstick of complexity because a set X is hyperarithmetical if and only if it is computed from a set Y which can be reached by (transfinitely) iterating the halting jump operator. Thus, the number of iterations of the halting jump needed to compute X yields a useful yardstick for the complexity of X. For more information about the hyperarithmetical hierarchy, see [19, Chapter 5].

Proposition 3.16. *For each $\sigma \in \{ad, stb, co, infad, infstb, infco\}$, there is a computable argumentation framework $\mathcal{F}_e$ which has continuum many non-empty σ extensions, yet no hyperarithmetical non-empty σ extension.*

(We take this to mean that there is no uniform way to construct—even with arbitrary access to the halting jump operator—a σ extension).

Proof. There exists a computable tree with uncountably many paths yet no hyperarithmetical path [18, Corollary XLI(b)]. Applying Theorem 3.15 to this tree yields a computable argumentation framework with uncountably many non-empty σ extensions, yet no hyperarithmetical non-empty σ extension. □

In the case of Proposition 3.16, there are σ extensions that are not particularly computationally powerful. They are not hyperarithmetical, but they also compute no hyperarithmetical sets. We can think of them as on the side of the hyperarithmetical hierarchy, thus simply not measured by the yardstick. This is always the case if an infinite AF has continuum many σ extensions. On the other hand, if a computable argumentation framework has a unique σ extension, the picture is quite different. In forthcoming work, we will settle the following conjecture.

Conjecture 3.17. *Let σ be in $\{ad, stb, co, infad, infstb, infco\}$ and suppose that $\mathcal{F}_e$ is a computable argumentation framework with a unique non-empty σ extension. Then, the non-empty σ extension of $\mathcal{F}_e$ is hyperarithmetical.*

On the other hand, we can show that there is no bound in the hyperarithmetical hierarchy on how complicated this extension might be.

Theorem 3.18. *Let σ be in $\{ad, stb, co, infad, infstb, infco\}$ and let H be a hyperarithmetical set. Then, there exists a computable AF $\mathcal{F}_e$ with a single non-empty σ extension X so that X computes H.*

Proof. This follows from Theorem 3.15 by encoding a tree with a single path π so that π computes H. Such a tree is known to exist for any hyperarithmetical H [18, Corollary XLIV(d)]. □

4. Encoding a tree into an argumentation framework

This section is devoted to our hardness results: we provide lower bounds for the complexity of our computational problems. In particular, given a tree $\mathcal{T} \subseteq \omega^{<\omega}$, we will define an argumentation framework $\mathcal{F}^{\mathcal{T}} = (A^{\mathcal{T}}, R^{\mathcal{T}})$. The set of arguments $A^{\mathcal{T}}$ of $\mathcal{F}^{\mathcal{T}}$ is computable and consists of $\{a_\sigma : \sigma \in \mathcal{T}\} \cup \{b_\sigma : \sigma \in \mathcal{T}\}$. The attack relation $R^{\mathcal{T}}$ of $\mathcal{F}^{\mathcal{T}}$ contains all and only the following edges: For all $\sigma \in \mathcal{T}$,

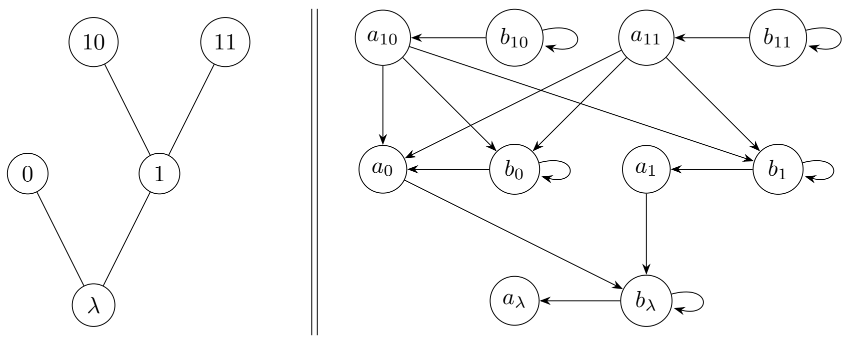

Figure 1: Example of our encoding of trees into AFs: the left-side represents the tree $\{\lambda, 0, 1, 10, 11\}$, the right-side is the resulting AF. When applied to trees T of infinite height, $[T]$ will be encoded into the stable, complete, and admissible extensions of $\mathcal{F}_T$. Let's stress that, for the example shown in the figure, the only admissible extension of $\mathcal{F}_\mathcal{T}$ is the empty one.

1. $b_\sigma \rightarrowtail b_\sigma$;
2. $b_\sigma \rightarrowtail a_\sigma$;
3. $a_\sigma \rightarrowtail b_\tau$, if $|\sigma| = |\tau| + 1$;
4. $a_\sigma \rightarrowtail a_\tau$, if $|\sigma| = |\tau| + 1$ and $\tau \not\preceq \sigma$.

Figure 1 gives an example of our encoding for a finite tree. We next consider which extensions in $\mathcal{F}^\mathcal{T}$ are admissible, stable, or complete.

Notation. For $\pi \in [\mathcal{T}]$ and $n \in \omega$, let S^n_π be the set $\{a_\sigma : \sigma \prec \pi \text{ and } |\sigma| \geq n\}$.

Lemma 4.1. *A non-empty extension S of $\mathcal{F}^\mathcal{T}$ is admissible iff S is exactly S^n_π for some $\pi \in [\mathcal{T}]$ and $n \in \omega$.*

Proof. ($\Rightarrow$): Suppose that $S \neq \emptyset$ belongs to *ad*($\mathcal{F}^\mathcal{T}$). First, observe that no b_σ can be in S, as all such arguments are self-defeating and S must be conflict-free. Next, observe that, if $a_\tau \in S$, then there must be some i so that $a_{\tau^\frown i} \in S$: this is because some element of S must defend a_τ from b_τ and such an element must be an a_σ with $|\sigma| = |\tau| + 1$. But it must have $\tau \prec \sigma$ as otherwise a_σ would attack a_τ.

Finally, take ρ of minimal length so $a_\rho \in S$. Then the previous paragraph shows that S contains $S^{|\rho|}_\pi$ for some $\pi \in [\mathcal{T}]$ with $\rho \prec \pi$. Since ρ was chosen of minimal length, no a_τ with τ shorter can be in S. Moreover, no a_τ with $|\tau| \geq |\rho|$ and $\tau \not\prec \pi$ can be in S, as otherwise S would not be conflict-free. Thus, $S = S^{|\rho|}_\pi$.

($\Leftarrow$): Any element which attacks $a_{\pi\restriction_n}$ is itself attacked by either $a_{\pi\restriction_{n+1}}$ or $a_{\pi\restriction_{n+2}}$, so $S^n_\pi \subseteq f_{\mathcal{F}^\mathcal{T}}(S)$. ☐

Lemma 4.2. *An extension S of $\mathcal{F}^\mathcal{T}$ is stable iff S is exactly S^0_π for some $\pi \in [\mathcal{T}]$.*

Proof. (⇒): Suppose that $S \in stb(\mathcal{F}^{\mathcal{T}})$. Then, since S is admissible, we know $S = S^n_\pi$ for some $\pi \in [\mathcal{T}]$ and $n \in \omega$. Since b_λ is the only the argument that attacks a_λ and $b_\lambda \notin S$, it must be the case that $a_\lambda \in S$. Thus, $n = 0$.

(⇐): Observe that S^0_π is conflict-free and any other argument of $F^{\mathcal{T}}$ is contained in $(S^0_\pi)^+$. Thus, S^0_π is a stable extension of $\mathcal{F}^{\mathcal{T}}$. □

Lemma 4.3. *A non-empty extension S of $\mathcal{F}^{\mathcal{T}}$ is complete iff S is S^0_π for some $\pi \in [\mathcal{T}]$.*

Proof. (⇒): Suppose that $S \neq \emptyset$ belongs to $co(\mathcal{F}^{\mathcal{T}})$. Since complete extensions are admissible, we see that $S = S^n_\pi$, for some $X \in [\mathcal{T}]$ and $n \in \omega$. But observe that if $n > 0$, then S would not be complete: indeed, a_σ with $\sigma = \pi \upharpoonright_{n-1}$ would be defended by S but not in S. Thus, n must be equal to 0.

(⇐): This follows since S^0_π is stable and all stable extensions are complete. □

We are now in a position to obtain hardness results for the computational problems described in Definition 3.3.

Theorem 4.4. *The following hold:*

1. *for $\sigma \in \{ad, stb, co, infad, infstb, infco\}$, NE^∞_σ is Σ^1_1-hard;*
2. *for $\sigma \in \{stb, infad, infstb, infco\}$, $Exist^\infty_\sigma$ is Σ^1_1-hard;*
3. *for $\sigma \in \{ad, stb, co, infad, infstb, infco\}$, $Cred^\infty_\sigma$ is Σ^1_1-hard.*

Proof. 1. Let $X \in \Sigma^1_1$ and let $(\mathcal{T}^X_n)_{n\in\omega}$ be a tree-sequence for X, as given by Theorem 3.7. To show Σ^1_1-hardness, we need to produce a computable function f so that $n \in X$ if and only if $f(n) \in \mathrm{NE}^\infty_\sigma$. We let $f(n)$ be a computable index for $\mathcal{F}^{\mathcal{T}^X_n}$. Then Lemmas 4.1, 4.2, and 4.3 prove that $n \in X$ if and only if $\mathcal{T}^X_n$ is ill-founded if and only if $\mathcal{F}^{\mathcal{T}^X_n}$ has a non-empty σ extension for each $\sigma \in \{ad, stb, co, infad, infstb, infco\}$.

2. For each of these σ, the empty set is not a σ extension, so $\mathrm{Exist}^\infty_\sigma = \mathrm{NE}^\infty_\sigma$, which we showed above is Σ^1_1-hard.

3. In the proof of 1. above, we reduced a given Σ^1_1 set X to $\mathrm{NE}^\infty_\sigma$ by sending n to $\mathcal{F}^{\mathcal{T}^X_n}$. Note that $\mathcal{F}^{\mathcal{T}^X_n}$ has a non-empty σ extension if and only a_λ is in some σ extension. Thus sending n to the $\langle e, a_\lambda\rangle$ where e is a computable index for $\mathcal{F}^{\mathcal{T}^X_n}$ shows that $\mathrm{Cred}^\infty_\sigma$ is Σ^1_1-hard. □

Theorem 4.5. *For $\sigma \in \{ad, stb, co, infad, infstb, infco\}$, Uni^∞_σ is Π^1_1-hard.*

Proof. We first consider $\sigma \in \{ad, co\}$. Let $X \in \Pi^1_1$ and let $(\mathcal{T}^{\omega\smallsetminus X}_n)_{n\in\omega}$ be a tree-sequence for its complement. For $n \in \omega$, consider the sequence of AFs $\mathcal{F}^{\mathcal{T}^{\omega\smallsetminus X}_n}$. Note that $\emptyset$ is an admissible extension in any AF and since every argument in $\mathcal{F}^{\mathcal{T}^{\omega\smallsetminus X}_n}$ is attacked, $\emptyset$ is also a complete extension. Thus, $\mathcal{F}^{\mathcal{T}^{\omega\smallsetminus X}_n}$ has a unique σ extension if and only if $\mathcal{T}^{\omega\smallsetminus X}_n$ is well founded if and only if $n \in X$, which shows that Uni^∞_{ad} and Uni^∞_{co} are Π^1_1-hard.

For the other σ, $\emptyset$ is not a σ extension. We use Theorem 3.10 to show Π^1_1-hardness. Let X be any Π^1_1 set. Then we get from Remark 3.11 a sequence of trees $\mathcal{T}'_n$ so that $0^\infty \in [\mathcal{T}'_n]$ for each n, and $\{n : \mathcal{T}'_n \text{ has only one path}\}$ is Π^1_1-hard. It follows from Lemmas 4.1, 4.2, and 4.3 that this is if and only if $\mathcal{F}^{\mathcal{T}'_n}$ has a unique σ extension, which shows that Π^1_1-hardness of $\mathrm{Uni}^\infty_\sigma$. □

Theorem 4.6. *For any $\sigma \in \{stb, infstb, infco\}$, Skept$_\sigma^\infty$ is Π_1^1-hard.*

Proof. Let X be a Π_1^1 set. Then we get from Remark 3.11 a sequence of trees $\mathcal{T}_n'$ so that $0^\infty \in [\mathcal{T}_n']$ for each n, and $\{n : \mathcal{T}_n' \text{ has only one path}\}$ is Π_1^1-hard. Then, note that $\langle e, a_0 \rangle \in \text{Skept}_\sigma^\infty$ where e is a computale index for $\mathcal{T}_n'$ if and only if $\mathcal{T}_n'$ only has paths π with $\pi(0) = 0$ if and only if $\mathcal{T}_n'$ has only one path (see the definition of $\mathcal{T}_n'$ in Remark 3.11) if and only if $n \in X$. This shows the Π_1^1-hardness of Skept$_\sigma^\infty$. □

We note that the above argument does not work for *infad* since, even if π is the only path through a tree T, each S_π^n is an infinite admissible extension in $\mathcal{F}^\mathcal{T}$ and $\bigcap_n S_\pi^n = \emptyset$, so in any $\mathcal{F}^\mathcal{T}$, Skept$_{infad}^\infty = \emptyset$.

Theorem 4.7. *Skept$_{infad}^\infty$ is Π_1^1-hard.*

Proof. Let X be a Π_1^1 set. We get from Remark 3.11 a sequence of trees $\mathcal{T}_n'$ so that each has one path $0^\infty \in \mathcal{T}_n'$ and has another path extending 1 if and only if $n \notin X$.

For each $n \in \omega$, we construct an AF $\mathcal{G}_n = (A_{\mathcal{G}_n}, R_{\mathcal{G}_n})$ slightly larger than $\mathcal{F}^{\mathcal{T}_n'}$. In particular $A_{\mathcal{G}_n} = A_{\mathcal{F}^{\mathcal{T}_n'}} \cup \{x_0, y_0, x_1, y_1\}$. We let $(w, z) \in R_{\mathcal{G}_n}$ if

- $w, z \in A_{\mathcal{F}^{\mathcal{T}_n'}}$ and $(x, y) \in R_{\mathcal{F}^{\mathcal{T}_n'}}$
- $w = y_i$ and $z = y_i$
- $w = x_i$ and $z = x_{1-i}$
- $w = x_i$ and $z = y_i$
- $w = y_i$ and $z = a_\sigma$ where $\sigma(0) = i$
- $w = a_\sigma$ with $\sigma(0) = i$ and $z = y_{1-i}$

Lemma 4.8. *An infinite $S \subseteq A_{\mathcal{G}_n}$ is an admissible extension if and only if it equals a set $S_\pi^k \cup \{x_{\pi(0)}\}$ for some $\pi \in [\mathcal{T}_n']$ and $k \in \omega$.*

Proof. Let U be an infinite admissible extension. Note first that arguments y_0, y_1 and b_σ cannot be in U since they are self-defeating. Then, since U is infinite, U must contain elements a_σ for $\sigma \in \mathcal{T}_n'$. By the same argument as in Lemma 4.1, we get that $U \cap A_{\mathcal{F}^{\mathcal{T}_n'}} = S_\pi^n$ for some $\pi \in [\mathcal{T}_n']$. But then since each element of S_n^π is attacked by $y_{\pi(0)}$, we must have $x_{\pi(0)} \in U$ to defend them. This excludes $x_{1-\pi(0)}$ from U since U is conflict-free.

It is straightforward to check that each of these are in fact admissible extensions. □

Finally, note that x_0 is in every infinite admissible extension of $\mathcal{G}_n$ if and only if there is no $\pi \in [\mathcal{T}_n']$ which extends 1 if and only if $n \in X$, showing that Skept$_{infad}^\infty$ is Π_1^1-hard. □

Theorem 4.9. *For $\sigma \in \{ad, stb, co, infad, infstb, infco\}$ and for any computable tree $\mathcal{T}$, there exists a computable AF $\mathcal{F}_e$ so that Spec$_\sigma^{\neg\emptyset}(\mathcal{F}_e) =$ Spec$(\mathcal{T})$.*

Proof. Observe that for the AF $\mathcal{F}_e = \mathcal{F}^\mathcal{T}$, it follows from Lemmas 4.1, 4.2, and 4.3 that the non-empty σ extensions are all infinite and are in the same Turing degrees as the paths through $\mathcal{T}$. □

5. Conclusion and future work

In this paper, we initiated a systematic exploration of the complexity issues inherent to infinite argumentation frameworks. To pursue this direction, we employed computability-theoretic techniques which are ideally suited for assessing the complexity of infinite mathematical objects. Our focus was on the credulous and skeptical acceptance of arguments, as well as the existence and uniqueness of extensions, for admissible, complete, and stable semantics. The computational problems we examined were found to be maximally complex, properly belonging to the complexity classes of Σ^1_1 and Π^1_1 sets. We also introduced and explored new semantics that are meaningful exclusively in the infinite setting, concerning the existence of infinite extensions that satisfy a given semantics σ.

It is natural to conceive of an argumentative scenario with arguments being added as time proceeds, such as the ongoing accumulation of scientific studies. Then, infinite frameworks naturally emerge as the union of the frameworks observed at each finite time. A key question, then, is how the acceptance of arguments within the infinite framework $\mathcal{F}$ can be related to the acceptance within the finite frameworks $(\mathcal{F}_t)$ which have appeared by time t. Our results show that, for complexity reasons alone, credulous and skeptical acceptance of arguments in $\mathcal{F}$ cannot be understood in terms of any kind of limiting procedure applied to the same problems in $\mathcal{F}_t$.

A plethora of intriguing questions regarding the complexity of infinite AFs remains open. In forthcoming extensions of this work, we shall fill the gaps that we left behind (such as the entries marked with a dagger in Table 2). Next, we will show that the techniques introduced here enable the construction of a single argumentation framework witnessing our hardness results, thereby proving that solving these problems is not only challenging for the entire class of argumentation frameworks but also remains difficult for an individual, specific framework.

Finally, future research will extend our analysis to analogous problems associated with other key semantics for AFs, including grounded, preferred, and ideal semantics. Given that the definitions of these semantics are more intricate than those we examined here, we anticipate the need for additional techniques to thoroughly analyze them.

Acknowledgments

Andrews was partially supported by NSF grant DMS-2348792. San Mauro is a member of INDAM-GNSAGA.

References

[1] P. Baroni, F. Cerutti, P. E. Dunne, M. Giacomin, Automata for infinite argumentation structures, Artificial Intelligence 203 (2013) 104–150.

[2] P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artificial intelligence 77 (1995) 321–357.

[3] A. J. García, G. R. Simari, Defeasible logic programming: An argumentative approach, Theory and practice of logic programming 4 (2004) 95–138.
[4] P. Baroni, M. Giacomin, G. Guida, Self-stabilizing defeat status computation: dealing with conflict management in multi-agent systems, Artificial Intelligence 165 (2005) 187–259.
[5] B. Verheij, Deflog: on the logical interpretation of prima facie justified assumptions, Journal of Logic and Computation 13 (2003) 319–346.
[6] M. Caminada, N. Oren, Grounded semantics and infinitary argumentation frameworks, in: Proceedings of the 26th Benelux Conference on Artificial Intelligence, BNAIC, 2014, pp. 25–32.
[7] R. Baumann, C. Spanring, Infinite argumentation frameworks: On the existence and uniqueness of extensions, in: Advances in Knowledge Representation, Logic Programming, and Abstract Argumentation: Essays Dedicated to Gerhard Brewka on the Occasion of his 60th Birthday, Springer, 2015, pp. 281–295.
[8] P. Baroni, F. Cerutti, P. E. Dunne, M. Giacomin, Computing with infinite argumentation frameworks: The case of afras, in: Theorie and Applications of Formal Argumentation: First International Workshop, TAFA 2011. Barcelona, Spain, July 16-17, 2011, Springer, 2012, pp. 197–214.
[9] S. Bistarelli, F. Santini, et al., Weighted argumentation., FLAP 8 (2021) 1589–1622.
[10] P. E. Dunne, Coherence in finite argument systems, Artificial Intelligence 141 (2002) 187–203.
[11] P. E. Dunne, The computational complexity of ideal semantics, Artificial Intelligence 173 (2009) 1559–1591.
[12] W. Dvořák, S. Woltran, Complexity of semi-stable and stage semantics in argumentation frameworks, Information Processing Letters 110 (2010) 425–430.
[13] V. Brattka, P. Hertling, Handbook of computability and complexity in analysis, Springer, 2021.
[14] K. V. Velupillai, Computable foundations for economics, Routledge, 2012.
[15] G. Jäger, J. Rogers, Formal language theory: refining the chomsky hierarchy, Philosophical Transactions of the Royal Society B: Biological Sciences 367 (2012) 1956–1970.
[16] P. Baroni, M. Giacomin, Semantics of abstract argument systems, Argumentation in artificial intelligence (2009) 25–44.
[17] P. E. Dunne, M. Wooldridge, Complexity of abstract argumentation, Argumentation in artificial intelligence (2009) 85–104.
[18] H. Rogers Jr, Theory of recursive functions and effective computability, MIT press, 1987.
[19] C. J. Ash, J. Knight, Computable structures and the hyperarithmetical hierarchy, Elsevier, 2000.
[20] S. C. Kleene, Arithmetical predicates and function quantifiers, Transactions of the American Mathematical Society 79 (1955) 312–340.
[21] A. Kechris, Classical descriptive set theory, volume 156, Springer Science & Business Media, 2012.