An Argumentation Calculus for Equilibrium Logic

Tobias Geibinger and Thomas Eiter

Institute for Logic and Computation, TU Wien, Austria
{tobias.geibinger,thomas.eiter}@tuwien.ac.at

Abstract. In this paper, we introduce a characterisation of answer set entailment through a combination of monotonic proof theory and abstract argumentation.
As the name suggests, answer set entailment is related to Answer Set Programming (ASP). In particular it encompasses the questions of which formulas hold in some or all answer sets. We study this entailment for equilibrium logic, which is a logical characterisation of ASP.
Our argumentation calculus AAC proves a sound and complete axiomatisation of cautious and brave entailment in equilibrium logic.
The motivation behind this preliminary work is to study notions of explainability for answer set entailment, which, for example, can be used to formalise inconsistency. By building on top of abstract argumentation, it might be possible to use explanation techniques from that area in ASP. Furthermore, while nonmonotonic proof theory is quite challenging, our characterisation shows that argumentation calculi are achievable.

1 Introduction

Answer Set Programming (ASP) is a symbolic rule-based reasoning formalism that has been used for various AI applications in numerous domains [21, 24], among them scheduling [2, 1, 36], product configuration [15], life sciences [23], health insurance [7], traffic flow simulation [18], or psychology [28], to mention a few. ASP allows for a declarative encoding of problems in a succinct manner. Solutions for them are obtained from *answer sets*, which result from the evaluation of the encoding using an ASP solver. In this work, we study answer set entailment, which intuitively formalises which sentences hold in every answer set of a program. This concept itself is already of practical interest, and has been applied for diagnosis [17] and planning [20]. Notably, answer set entailment is at the core of abductive explanations [20].

Given the practical usage of ASP, questions of explainability have been raised. Those questions or problems generally concern themselves with either answering why certain literals hold in a particular answer set or why there is no answer set at all, cf. Fandinno and Schulz [25] for a survey. Both these questions can be formulated through answer set entailment. However, simply stating a valid entailment is in general not an explanation on its own and needs to be justified. Such justifications can be obtained via a formal proof system, incorporating the logical base of ASP in its inference rules.

Although ASP has some efficient solvers available [29, 26, 3] and its model-theoretic properties have been studied extensively, the same cannot be said for proof-theoretic investigations. Given the nonmonotonic nature of ASP, obtaining a proof calculus is non-trivial, but some attempts have been made [27, 12, 34, 19].

In this work, we introduce an argumentation calculus for answer set entailment, which combines a monotonic sequent calculus with abstract argumentation. This sound and complete characterisation is defined for equilibrium logic, which generalises ASP to arbitrary propositional theories.

The calculus is intended to be a first step towards an investigation of argumentation based explanations for answer set entailment. However, its concrete study is still a topic for future work.

2 Preliminaries

We consider propositional *equilibrium logic (EL)* [32, 33] over a propositional language $\mathcal{L}$ over atoms Atoms with the usual logical connectives $\wedge, \vee, \supset, \neg$ and logical constant $\bot$.

The semantics of EL is based on the *logic of here and there* (also called *HT logic*), which is defined over the same language. An interpretation in HT logic is a pair $\langle H, T\rangle$ where $H \subseteq T \subseteq At$. Whether an HT interpretation $\langle H, T\rangle$ is a *model* of a formula φ (denoted by $\langle H, T\rangle \models \varphi$) is inductively defined as follows:

- $\langle H, T\rangle \not\models \bot$
- $\langle H, T\rangle \models p$ iff $p \in H$
- $\langle H, T\rangle \models \varphi \wedge \psi$ iff $\langle H, T\rangle \models \varphi$ and $\langle H, T\rangle \models \psi$
- $\langle H, T\rangle \models \varphi \vee \psi$ iff $\langle H, T\rangle \models \varphi$ or $\langle H, T\rangle \models \psi$
- $\langle H, T\rangle \models \varphi \supset \psi$ iff $T \models \varphi \supset \psi$, and $\langle H, T\rangle \not\models \varphi$ or $\langle H, T\rangle \models \psi$
- $\langle H, T\rangle \models \neg\varphi$ iff $T \not\models \varphi$ and $\langle H, T\rangle \not\models \varphi$

Note that whenever $H = T$, $\langle H, T\rangle \models \varphi$ corresponds to the satisfaction relation of classical propositional logic. By slight abuse of notation, we use $T \models \varphi$ to refer to $\langle T, T\rangle \models \varphi$ and call such models *classical* or *total*.

From the above semantics, two useful observations follow. The first is that $\langle H, T\rangle \models \varphi$ implies $\langle T, T\rangle \models \varphi$. This property is also called *persistence*. The other important property is that $\langle H, T\rangle \models \neg\varphi$ iff $T \models \neg\varphi$, i.e., negation is only evaluated over T.

An HT model $\langle H, T\rangle$ of a formula φ is an *equilibrium model* iff $H = T$ and for any other HT interpretation $\langle H', T\rangle$ such that $H' \subset H$, $\langle H', T\rangle \not\models \varphi$. The latter conditions is also referred to as *stability*. We denote an equilibrium model $\langle T, T\rangle$ by T whenever convenient and we use a single set I to refer to classical interpretations. As usual, a set of formulas is called a *theory*, and we say that an interpretation is a model of a theory if it satisfies all contained formulas.

ASP programs are generally sets of rules of the form $H \leftarrow B$, where H and B are sets of literals. In EL, such a rule is encoded as an implication $\bigwedge B \supset \bigvee H$ and a program is then a theory consisting of such an implication for each rule.

As has been shown by Pearce [32], the equilibrium models of such an encoded program amount to its answer sets.

From now on, whenever we talk about a *program*, we refer to a theory which encodes an ASP program as described above.

We use $\Gamma \models \varphi$ to denote that a theory classically entails φ and $\Gamma \models_{HT} \varphi$ for entailment in HT.

We will assume some sound and complete HT calculus, e.g. the sequent calculus from Mints [30], such that $\Gamma \Rightarrow \varphi$ is derivable iff $\Gamma \models_{HT} \varphi$ holds. Mints' calculus consists of the axioms $\Gamma, \varphi \Rightarrow \Delta, \varphi$ and $\Gamma, \neg\varphi, \varphi \Rightarrow \Delta$ and the usual rules for $\wedge$ and $\vee$. For $\supset$, we have the rules:

$$\frac{\Gamma, \neg\varphi \Rightarrow \Delta \qquad \Gamma \Rightarrow \Delta, \varphi, \neg\psi \qquad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \varphi \supset \psi \Rightarrow \Delta} \supset_l$$

$$\frac{\Gamma, \varphi \Rightarrow \Delta, \psi \qquad \Gamma, \neg\varphi \Rightarrow \Delta, \neg\psi}{\Gamma \Rightarrow \Delta, \varphi \supset \psi} \supset_r$$

Furthermore, the calculus requires the following nested rules for $\neg$:

$$\frac{\Gamma, \neg\varphi \Rightarrow \Delta \qquad \Gamma, \neg\psi \Rightarrow \Delta}{\Gamma, \neg(\varphi \wedge \psi) \Rightarrow \Delta} \neg\wedge_l \qquad \frac{\Gamma \Rightarrow \Delta, \neg\varphi, \neg\psi}{\Gamma \Rightarrow \Delta, \neg(\varphi \wedge \psi)} \neg\wedge_r$$

$$\frac{\Gamma, \neg\varphi, \neg\psi \Rightarrow \Delta}{\Gamma, \neg(\varphi \vee \psi) \Rightarrow \Delta} \neg\vee_l \qquad \frac{\Gamma \Rightarrow \Delta, \neg\varphi \qquad \Gamma \Rightarrow \Delta, \neg\psi}{\Gamma \Rightarrow \Delta, \neg(\varphi \vee \psi)} \neg\vee_r$$

$$\frac{\Gamma, \neg\psi \Rightarrow \Delta, \neg\varphi}{\Gamma, \neg(\varphi \supset \psi) \Rightarrow \Delta} \neg\supset_l \qquad \frac{\Gamma, \neg\varphi \Rightarrow \Delta \qquad \Gamma \Rightarrow \Delta, \neg\psi}{\Gamma \Rightarrow \Delta, \neg(\varphi \supset \psi)} \neg\supset_r$$

$$\frac{\Gamma \Rightarrow \Delta, \neg\varphi}{\Gamma, \neg\neg\varphi \Rightarrow \Delta} \neg\neg_l \qquad \frac{\Gamma, \neg\varphi \Rightarrow \Delta}{\Gamma \Rightarrow \Delta, \neg\neg\varphi} \neg\neg_r$$

Regarding entailment relations in equilibrium logic, one usually considers two variants. *Brave entailment*, denoted by $\Gamma \approx_b \varphi$, holds if there is some equilibrium model I of Γ where $I \models \varphi$. The other relations is *skeptical entailment* $\Gamma \approx_s \varphi$, which holds if φ holds in every equilibrium model of Γ.

We follow the seminal work of Dung [16] and utilise argumentation frameworks $\mathcal{AF} = (\mathsf{Arg}, \mathsf{Atck})$ which are directed graphs consisting of arguments $a, b, c, \ldots \in \mathsf{Arg}$ and an attack relation $\mathsf{Atck} \subseteq \mathsf{Arg} \times \mathsf{Arg}$.

A set $A \subseteq \mathsf{Arg}$ is called an *extension*, which is *conflict-free* if for each $(a, b) \in \mathsf{Atck}$ it holds that if $a \in A$ then $b \notin A$; and *stable* if it is conflict-free and for every $b \in \mathsf{Arg} \setminus A$, there is $a \in A$ such that $(a, b) \in \mathsf{Atck}$.

3 Argumentation Calculus

We now come the main contribution of this work, our argumentation calculus AAC for equilibrium logic.

First, some additional notation. We consider $\mathcal{L}^{\mathsf{def}} = \{\sim\ell \mid \ell \in \mathcal{L}^{\mathsf{lit}}\}$ where $\mathcal{L}^{\mathsf{lit}} = \mathsf{Atoms} \cup \{\neg a \mid a \in \mathsf{Atoms}\}$ and add the following rule to the HT calculus:

$$\frac{\neg\phi, \Delta \Rightarrow \Gamma}{\sim\phi, \Delta \Rightarrow \Gamma} \sim_l$$

Intuitively, $\sim\varphi$ means that φ is "assumed" to be false.

The calculus AAC is then based on the following argumentation framework. Let Γ be a propositional theory, then $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ is defined as:

- $\Sigma \Rightarrow \varphi \in \mathsf{Arg}$ iff $\Sigma \Rightarrow \varphi$ is HT-derivable and $\Sigma \subseteq \Gamma \cup \mathcal{L}^{\mathsf{def}}$;
- $(a, b) \in \mathsf{Atck}$ iff
  (a) $a : \Gamma \Rightarrow \varphi \in \mathsf{Arg}$ and $b : \sim\varphi, \Sigma \Rightarrow \psi \in \mathsf{Arg}$, or
  (b) $a : \Gamma \Rightarrow \bot \in \mathsf{Arg}$ and $b \in \mathsf{Arg}$.

What we will now show is that by looking at the stable extensions of $\mathcal{AF}(\Gamma)$ we can characterize the consequences of Γ.

The framework consists of arguments which are derivable sequents in HT. Such an argument is potentially attacked, whenever it requires the assumption that a literal is false and it's attackers are the arguments – i.e. sequents – which derive said literal.

A special case is any argument which derives $\bot$, such an argument attacks all other arguments including itself. The latter ensures that any classically inconsistent theory has no stable extension.

Lemma 1. *Let Γ be a propositional theory. If Γ has no classical models, then $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ has no stable extensions.*

Proof. If Γ has no classical models, then $\Gamma \Rightarrow \bot$ is derivable, as the existence of an HT model $\langle J, I\rangle$ implies that I is a classical model. By the definition above, there is a set of arguments $A \subseteq \mathsf{Arg}$ s.t. for each $a : \Sigma \Rightarrow \varphi \in A$, it holds that $\varphi = \bot$ and $\Sigma \cap \mathcal{L}^{\mathsf{def}} = \emptyset$. The latter implies that no argument in A is attacked by any argument outside of A, but we have $(a, b) \in \mathsf{Atck}$ for each $a, b \in A$ and in particular for $a = b$. Suppose now that there is some stable extension E. If $E \cap A \neq \emptyset$, then there is some $a \in E \cap A$ which attacks itself and E is thus not conflict-free and not stable. Otherwise, if $E \cap A = \emptyset$, then there is $a \in A$ such that $a \notin E$ but by construction, a is not attacked by any argument in E. Hence, E cannot be stable.

Lemma 2. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ with stable extension E and formula φ. If there is an argument $\Sigma \Rightarrow \neg\varphi$ in E, then there is no argument $\Theta \Rightarrow \varphi$ in E and vice versa.*

Proof. Towards a contradiction, suppose $\Sigma \Rightarrow \varphi$ and $\Theta \Rightarrow \neg\varphi$ in are E According to Lemma 1, E being a stable extension implies that Γ is classically consistent. Now, suppose $\sim\neg p \in \Sigma$ and $\sim p \in \Theta$. The latter implies that $\Theta \Rightarrow \neg p$ is derivable and thus in E as any attack would also attack $\Theta \Rightarrow \neg a$. However, $\Theta \Rightarrow \neg p$ attacks $\Sigma \Rightarrow a$ since we assume $\sim\neg p \in \Sigma$. This contradicts E being a stable extension. Furthermore, the case where $\sim\neg p \in \Theta$ and $\sim p \in \Sigma$ follows mutatis mutandis.

Hence, the two arguments do not contain contradictory assumptions and since Γ is classically consistent, it follows that $\Sigma \cup \Theta$ is consistent in the sense that $\Sigma, \Theta \Rightarrow \bot$ cannot be derived.

Considering our assumptions that $\Sigma \Rightarrow \varphi$ and $\Theta \Rightarrow \neg\varphi$ are in E and thus derivable, we obtain that $\Sigma, \Theta \Rightarrow \varphi$ and $\Sigma, \Theta \Rightarrow \neg\varphi$ can both be derived due to monotonicity. The latter implies $\Sigma, \Theta \Rightarrow \bot$ can be derived. Contradiction.

We will start by showing how $\mathcal{AF}(\Gamma)$ captures brave entailment, for which we give some auxiliary results.

Lemma 3. *Given a propositional theory Γ and interpretation I. If $I \models \Gamma$ and $\Gamma' \Rightarrow p$ for every $p \in I$, where $\Gamma' = \Gamma \cup \{\sim a \mid a \notin I\} \cup \{\sim\neg a \mid a \in I\}$, then I is a stable model of Γ.*

Proof. Towards a contradiction, suppose I is not a stable model. Then, there is some $\langle J, I\rangle \models \Gamma$ s.t. $J \subset I$. However, $\Gamma' \Rightarrow p$ for every $p \in I$ implies $J = I$. Contradiction.

Intuitively, the above states that if the theory has a classical model and every atom in the model is derivable, then the model is stable. The other direction of this statement is captured by the next lemma.

Lemma 4. *Given a propositional theory Γ with stable model I. Then, $\Gamma' \Rightarrow \varphi$ is derivable iff $I \models \varphi$, where $\Gamma' = \Gamma \cup \{\sim a \mid a \notin I\} \cup \{\sim\neg a \mid a \in I\}$.*

Proof. It can be seen that Γ' has only one classical model which is I (interpreting $\sim$ like a normal negation). Furthermore, since I is a stable model of Γ, $\langle I, I\rangle$ is also the only HT model of Γ and thus of Γ'. Since the HT calculus is complete, $\Gamma' \Rightarrow \varphi$ is derivable iff $I \models \varphi$.

A further important result we need to establish, is that extensions do not contain contradictory derivations.

Lemma 5. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ with stable extension E. If there is an argument $\Sigma \Rightarrow a$ in E, where a is an atom, then there is no argument $\Theta \Rightarrow \neg a$ in E.*

Proof. Towards a contradiction, suppose $\Sigma \Rightarrow a$ and $\Theta \Rightarrow \neg a$ in are E According to Lemma 1, E being a stable extension implies that Γ is classically consistent. Now, suppose $\sim\neg p \in \Sigma$ and $\sim p \in \Theta$. The latter implies that $\Theta \Rightarrow \neg p$ is derivable and thus in E as any attack would also attack $\Theta \Rightarrow \neg a$. However, $\Theta \Rightarrow \neg p$ attacks $\Sigma \Rightarrow a$ since we assume $\sim\neg p \in \Sigma$. This contradicts E being a stable extension. Furthermore, the case where $\sim\neg p \in \Theta$ and $\sim p \in \Sigma$ follows mutatis mutandis.

Hence, the two arguments do not contain contradictory assumptions and since Γ is classically consistent, it follows that $\Sigma \cup \Theta$ is consistent in the sense that $\Sigma, \Theta \Rightarrow \bot$ cannot be derived.

Considering our assumptions that $\Sigma \Rightarrow a$ and $\Theta \Rightarrow \neg a$ are in E and thus derivable, we obtain that $\Sigma, \Theta \Rightarrow a$ and $\Sigma, \Theta \Rightarrow \neg a$ can both be derived due to monotonicity. The latter implies $\Sigma, \Theta \Rightarrow \bot$ can be derived. Contradiction.

With those results in place, we can show that brave entailment, i.e. the existence of an answer set where a certain formula holds, implies the existence of a stable extension containing a derivation of said formula.

Lemma 6. *Given a propositional theory Γ and formula φ. If there is a stable model I of Γ such that $I \models \varphi$, then there is a stable extension E of $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ such that $\Sigma \Rightarrow \varphi \in E$.*

Proof. Let $\Gamma' = \Gamma \cup \{\sim a \mid a \notin I\} \cup \{\sim\neg a \mid a \in I\}$. Clearly, $\Gamma' \models_{HT} \varphi$ follows from $I \models \varphi$ and thus $\Gamma' \Rightarrow \varphi$. Consider the following extension E, which includes every argument $\Sigma \Rightarrow \psi$ where $\Sigma \cap \mathcal{L}^{\mathsf{def}} = \emptyset$. We call those the classical arguments Furthermore, E includes $\Sigma \Rightarrow \psi$ where $\Sigma \subseteq \Gamma \cup \{\sim a \mid a \notin I\} \cup \{\sim\neg a \mid a \in I\}$ which are dubbed the assumptive arguments. Note that I being a stable model of Γ implies that I is a classical model. Hence, we do not have any attacks between classical arguments, since there is no argument of form $\Sigma \Rightarrow \bot$. Suppose we have an attack between two arguments $a : \Sigma_1 \Rightarrow \psi$ and $b : \Sigma_2, \sim\psi \Rightarrow \chi$ of E. By construction, either (i) $\psi =\sim p$ or (ii) $\psi =\sim\neg p$. If (i), then $p \notin I$ but $\Gamma' \Rightarrow a$ which contradicts I being a stable model of Γ by Lemma 4. Similarly, (ii) implies $p \in I$ but $\Gamma' \Rightarrow \neg p$ which also implies that I is not a stable model of Γ. Hence, E is conflict-free and it remains to show that every argument not in E is attacked. Any argument $b \notin E$ necessarily contains some assumption $\sim\psi$ such that either (a) $\psi =\sim p$ or (b) $\psi =\sim\neg p$. If (a), then by construction, $p \in I$ and thus $\Gamma' \Rightarrow p \in E$ which attacks b. If (b), then $p \notin I$ which implies $\Gamma' \Rightarrow \neg p \in E$. Clearly the latter argument attacks b which is thus attacked in every case.

For the other direction, we proceed by first establishing that every stable extension of $\mathcal{AF}(\Gamma)$ characterizes some equilibrium model of Γ.

Lemma 7. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ with stable extension E and $I = \mathsf{Atoms} \setminus \{p \in \mathsf{Atoms} \mid \sim p \Rightarrow \neg p \in E\}$. Then, $I = \{p \in \mathsf{Atoms} \mid \Sigma \Rightarrow p \in E$ for some $\Sigma\}$.*

Proof. We start by showing $I \subseteq \{p \in \mathsf{Atoms} \mid \Sigma \Rightarrow p \in E\}$ Towards a contradiction, assume there is some $p \in I$ for which there there is no $\Sigma \Rightarrow p \in E$. The latter implies that $\sim p \Rightarrow \neg p$ is not attacked by any argument of E and thus either E is not stable or the argument is in E. The former is a contradiction but the latter implies $p \notin I$ by construction and thus also a contradiction.

Remains to show that $I \supseteq \{p \in \mathsf{Atoms} \mid \Sigma \Rightarrow p \in E\}$. Again towards a contradiction, suppose there is some $p \in \mathsf{Atoms}$ such that $\Sigma \Rightarrow p \in E$ for some Σ, but $p \notin I$. The latter implies $\sim p \Rightarrow \neg p \in E$ and by Lemma 5, E is not stable. Contradiction.

Lemma 8. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ with stable extension E and $I = \mathsf{Atoms} \setminus \{p \in \mathsf{Atoms} \mid \sim p \Rightarrow \neg p \in E\}$. Then, for every argument $\Theta \Rightarrow \varphi$ in E, it holds that $\Theta \subseteq \Sigma$, where $\Sigma = \Gamma \cup \{\sim a \mid a \in \mathsf{Atoms}, a \notin I\} \cup \{\sim\neg a \mid a \in I\}$.*

Proof. Assume towards a contradiction that there is some argument $\Theta \Rightarrow \varphi$ in E s.t. $\Theta \not\subseteq \Sigma$. By construction the latter implies that there is some p such that (a) $\sim p \in \Sigma$ and $\sim\neg p \in \Theta$ or (b) $\sim\neg p \in \Sigma$ and $\sim p \in \Theta$.

Suppose (a) holds, then $\sim p \in \Sigma$ implies that $\sim p \Rightarrow \neg p$ is in E by construction. Thus, $\Theta \Rightarrow \varphi$ is attacked since $\sim\neg p \in \Theta$ which contradicts E being conflict-free and thus stable.

If (b) holds, then $\sim\neg p \in \Sigma$ implies $p \in I$ and thus $\Delta \Rightarrow p \in E$ for some Δ by Lemma 7. However, $\Delta \Rightarrow p$ attacks $\Theta \Rightarrow \varphi$ since $\sim p \in \Theta$ and thus contradicts E being a stable extension as it contains both arguments.

Now we can show that having a stable extension implies the existence of a stable model.

Lemma 9. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ with stable extension E and $I = \mathsf{Atoms} \setminus \{p \in \mathsf{Atoms} \mid \sim p \Rightarrow \neg p \in E\}$. Then, I is a stable model of Γ and $I \models \varphi$ for every formula φ iff there is some argument $\Delta \Rightarrow \varphi$ in E.*

Proof. First, $\Sigma \Rightarrow \varphi$ is not in E, which means it is attacked by some argument in E. However, if the attacking argument is of form $\Theta \Rightarrow p$, then by construction $\sim p \Rightarrow p \in E$ and would also be attacked, which contradicts E being stable. Similarly, if the attacking argument is $\Theta \Rightarrow \neg p$, then there is some argument $\Pi \Rightarrow p$ since $p \in I$ and thus E is not stable by Lemma 5. Hence, the argument $\Sigma \Rightarrow \varphi$ is in E.

By Lemma 7 for each $p \in I$, $\Theta \Rightarrow p \in E$ for some Δ. Now, by Lemma 8, $\Theta \Rightarrow p \in E$ implies $\Theta \subseteq \Sigma$ where $\Sigma = \Gamma \cup \{\sim a \mid a \in \mathsf{Atoms}, a \notin I\} \cup \{\sim\neg a \mid a \in I\}$. Due to monotonicity of the base calculus, $\Sigma \Rightarrow p$ is derivable which implies that I is a stable model of Γ by Lemma 3 if $I \models \Gamma$. The latter holds since otherwise, $\Sigma \Rightarrow \bot$ and since $\Sigma \Rightarrow \varphi$ is in E, also $\Sigma \Rightarrow \bot$ is in E and attacks all other arguments contradicting that E is stable.

Similarly, $\Delta \Rightarrow \varphi$ in E implies $\Sigma \Rightarrow \varphi$ and thus $I \models \varphi$ by Lemma 4. Also, the argument $\Delta \Rightarrow \varphi$ is in E where $\Delta = \Sigma$.

The following results then follows from the lemmas proving both directions.

Corollary 1. *Given a propositional theory Γ and formula φ. Then, $\Gamma \approx_b \varphi$ iff there is a stable extension E of $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ such that $\Sigma \Rightarrow \varphi \in E$.*

Proof. Follows from Lemmas 6 and 9.

We further want to establish that we can characterize skeptical entailment, where the consequence has to hold in every stable model, as well. In order to achieve this, we simply have to show that there is a one-to-one correspondence between the stable extensions and stable models.

Lemma 10. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ and formula φ. If for every stable extension E of $\mathcal{AF}(\Gamma)$, there is an argument $\Sigma \Rightarrow \varphi \in E$, then $I \models \varphi$ for every stable model I of Γ.*

Proof. Towards a contradiction, suppose for every stable extension E of $\mathcal{AF}(\Gamma)$, there is an argument $\Sigma \Rightarrow \varphi \in E$, but $I \not\models \varphi$ for some stable model I of Γ. Clearly, $I \not\models \varphi$ implies $I \models \neg\varphi$ and by Lemma 6 there is some stable extension E s.t. $\Sigma \Rightarrow \neg\varphi \in E$. By Lemma 2, $\Theta \Rightarrow \varphi \notin E$ for every Θ which contradicts our initial assumption.

Lemma 11. *Given a propositional theory Γ and $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ and formula φ. If for every stable model I of Γ it holds that $I \models \varphi$, then there is an argument $\Sigma \Rightarrow \varphi \in E$ in every stable extension E of $\mathcal{AF}(\Gamma)$.*

Proof. Towards a contradiction, suppose $I \models \varphi$ for every stable model I of Γ, but there is a stable extension E of $\mathcal{AF}(\Gamma)$ such that there is no argument $\Sigma \Rightarrow \varphi$ in E. By Lemma 9, $I = \mathsf{Atoms} \setminus \{p \in \mathsf{Atoms} \mid \sim p \Rightarrow \neg p \in E\}$ is a stable model of Γ and thus $I \models \varphi$ by assumption. Now, again by Lemma 9, there is an argument $\Sigma \Rightarrow \varphi \in E$. Contradiction.

From those lemmas, it directly follows that AAC captures skeptical entailment.

Corollary 2. *Given a propositional theory Γ and formula φ. Then, $\Gamma \approx_s \varphi$ iff for every stable extension E of $\mathcal{AF}(\Gamma) = (\mathsf{Arg}, \mathsf{Atck})$ there is an argument $\Sigma \Rightarrow \varphi \in E$.*

Proof. Follows from Lemmas 10 and 11.

To summarize, we have shown the following.

Theorem 1. *$\Gamma \approx_b \phi$ iff there is a stable extension A of $\mathcal{AF}(\Gamma)$ such that $\Gamma \Rightarrow \phi \in A$, and $\Gamma \approx_s \phi$ iff for each stable extension A of $\mathcal{AF}(\Gamma)$ it holds that $\Gamma \Rightarrow \phi \in A$.*

We conclude this section with some examples.

Example 1. Consider program $P_1 = \{a \leftarrow \neg b;\ b \leftarrow \neg a\}$ with answer sets $AS(P_1) = \{\{a\}, \{b\}\}$. Then $\mathcal{AF}(P_1)$ has two stable extensions E_1 and E_2.

The first extension E_1 contains, among others, an argument $a_1 : a \leftarrow \neg b, b \leftarrow \neg a, \sim b \Rightarrow a$, whereas E_2 contains an argument $a_2 : a \leftarrow \neg b, b \leftarrow \neg a, \sim a \Rightarrow b$. Clearly, those arguments attack each other in $\mathcal{AF}(P_1)$ and by the theorem above both $P_1 \approx_b a$ and $P_1 \approx_b b$ hold.

Example 2. Consider program $P_2 = \{a \leftarrow b;\ b \leftarrow a\}$ with answer sets $AS(P_2) = \{\emptyset\}$. Hence, it holds that $P_2 \approx_s \neg a$. Now, $\mathcal{AF}(P_2)$ has one stable extension E, which contains an argument $\sim a \Rightarrow a$. Given that a cannot be derived, no matter what literals we assume to be false, this argument is never attacked.

Example 3. Consider program $P_3 = \{p \leftarrow \neg p\}$ with answer sets $AS(P_3) = \emptyset$ and thus $P_3 \approx_s \perp$. Now, $\mathcal{AF}(P_3)$ contains the self-attacking argument $p \leftarrow \neg p, \sim p \Rightarrow p$ and has no stable extension.

Example 4. Let $P_4 = \{a \wedge \neg a\}$ which has no answer sets – and not even classical models – and thus $P_4 \approx_s \perp$. Now, $\mathcal{AF}(P_4)$ contains the all-attacking argument $a \wedge \neg a \Rightarrow \perp$ and thus has no stable extension.

4 Related Work

Regarding related work, we group it into work on other proof systems for equilibrium logic or ASP, encoding stable model semantics into argumentation frameworks, and related argumentation calculi.

For the latter, one approach was directly given by Dung [16]. However, this encoding only captures so-called supported models and while every stable model is a supported model, the converse only holds for a specific class of programs. Namely, tight programs [22]. A complete characterisation of stable models semantics was achieved by Bondarenko et al. using an extension of Argumentations Frameworks, namely, Assumption-based Argumentation (ABA) Frameworks [13]. In difference to standard AFs, ABA frameworks are defined over a deductive system which is represented as a pair $(\mathcal{L}, \mathcal{R})$, where $\mathcal{L}$ is the language and $\mathcal{R}$ is a set of deduction rules. An ABA framework is then a tuple $\langle T, Ab, \rangle$, where $T, Ab \subseteq \mathcal{L}$, $Ab \neq \emptyset$ indicating a knowledge base and a set of abducibles, and $\ : Ab \to \mathcal{L}$ maps any $\alpha \in Ab$ to its complement $\overline{\alpha}$. In an ABA framework, arguments are essentially pairs (Δ, φ), where $\Delta \subseteq Ab$, $\varphi \in \mathcal{L}$, and φ can be derived from $T \cup \Delta$ using the rules in $\mathcal{R}$.

Hence, in stands to reason that we could also encode our approach for equilibrium logic in an ABA framework, where $\mathcal{R}$ are the rules of the HT calculus including the $(\sim_l)$-rule and the abducibles Ab are $\mathcal{L}^{\mathsf{def}}$ with $\overline{\sim l} = l$. However, this idea does not capture attacks in AAC from an argument $\Gamma \Rightarrow \bot$ and simply adding $\bot$ to Ab would create too many arguments. This means that with this simple ABA encoding, we cannot capture that the AAC framework for a classically inconsistent theory, has no stable extension.

While this marks a difference in our approach and a potential ABA-based one, it might still be possible to achieve the overall goal of characterising answer set entailment and using AFs for explainability. However, such investigations are topic of future work.

Several proof systems for ASP can be found in the literature. Bonatti [11] introduced a resolution calculus for answer set entailment and provided another calculus for *brave entailment* [12]. The latter is different from the notion of entailment we study in the sense that the consequence does not need to hold in all models but rather in at least one. The calculi differ from their classical counterparts and operate on pairs of sets of literals and suffer from some drawbacks: they support only normal logic programs and rely on a rather complicated notion of *counter-supports* which is not axiomatised by the calculus but computed externally.

Gebser et al. [27] presented a tableau calculus which is aimed at axiomatising satisfiability and how solvers obtain their solutions and thus differs in motivation from our work. However, it can also be used for answer set entailment by translation of the entailment into an inconsistent program. While their calculus supports a large number of advanced ASP language features like disjunction and weight constraints, it does not cover arbitrary nested formulas like our calculus does. Furthermore, some rules in their calculus heavily rely on global syntactic notions which only work for programs.

Proofs for ASP also appear in the context of proof logging that ASP solvers provide to justify whenever they report unsatisfiability [4, 14]. However, those proofs are generally not geared towards human interpretability but rather verification and are not very concise.

Proof-like systems have been used to provide explanations as to why certain atoms are, or are not, in a given answer set [35, 5]. While the explanations given by those approaches share some similarity with a formal proof, they are not actual complete proof systems and cannot be applied without a model. Furthermore, they are limited to the basic ASP language.

For equilibrium logic, there are less systems. Recently, Eiter and Geibinger [19] introduced a sequent calculus for equilibrium logic, which generally behaves like a standard sequent calculus, but also includes a complex rule relying on an HT refutation calculus which axiomatises non-entailment. The latter is sometimes required for excluding unfounded sets.

Pearce et al. [34] introduced a tableau system for equilibrium logic. Their approach works in several stages. In the first stage, a dedicated calculus for total HT models is used to enumerate all models. With another dedicated calculus, those total models are then checked for stability in the second stage. Finally, in the last stage, it is checked via tableaux whether the given formula holds in the previously obtained total and stable models.

The fix-point characterisation for equilibrium logic [33] does not axiomatise the inference relation over equilibrium models, but rather only provides a definition of stable models via HT entailment and model selection. The characterisation can be used in conjunction with an HT sequent calculus [31] to justify the atoms in an equilibrium model in a proof-theoretic manner.

Argumentation Calculi have been studied for default reasoning [9] and deontic reasoning [10, 6], which essentially characterises Input/Output logic. The latter system has further been applied for explainability [8].

5 Conclusion and Future Work

We introduced the argumentation calculus AAC for equilibrium logic, which soundly and completely characterises answer set entailment. The characterisation is based on stable extensions of an argumentation framework, which consist of all monotonic derivations we can make given a set of assumptions. Attacks are then intuitively defined as contradictory derivations.

As we have mentioned in the beginning, the motivation for this work is to investigate explanation approaches for argumentation frameworks in the context of equilibrium logic. With the cornerstone now set, such studies can be conducted in future work. Further future work includes alternative encodings of answer set entailment in argumentation and related frameworks, like the previously mentioned ABA ,and the investigation of which formalism proves the best basis for explanation.

References

1. Abels, D., Jordi, J., Ostrowski, M., Schaub, T., Toletti, A., Wanko, P.: Train Scheduling with Hybrid ASP. In: Proceedings of the 15th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2019). LNCS, vol. 11481, pp. 3–17. Springer (2019)
2. Ali, R., El-Kholany, M.M.S., Gebser, M.: Flexible job-shop scheduling for semiconductor manufacturing with hybrid answer set programming (application paper). In: Proceedings of the 25th International Symposium on Practical Aspects of Declarative Languages (PADL 2023). pp. 85–95. Springer, Cham (2023)
3. Alviano, M., Dodaro, C., Faber, W., Leone, N., Ricca, F.: WASP: A native ASP solver based on constraint learning. In: Proceedings of the 12th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2013). LNCS, vol. 8148, pp. 54–66. Springer (2013). https://doi.org/10.1007/978-3-642-40564-8_6
4. Alviano, M., Dodaro, C., Fichte, J.K., Hecher, M., Philipp, T., Rath, J.: Inconsistency proofs for ASP: the ASP - DRUPE format. Theory and Practice of Logic Programming **19**(5-6), 891–907 (2019). https://doi.org/10.1017/S1471068419000255, https://doi.org/10.1017/S1471068419000255
5. Alviano, M., Hahn, S., Sabuncu, O., Weichelt, H.: Answer set explanations via preferred unit-provable unsatisfiable subsets. In: Proceedings of the 17th International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR 2024). LNCS, vol. 15245, pp. 187–199. Springer (2024). https://doi.org/10.1007/978-3-031-74209-5_15
6. Arieli, O., van Berkel, K., Straßer, C.: Defeasible normative reasoning: A proof-theoretic integration of logical argumentation. In: Wooldridge, M.J., Dy, J.G., Natarajan, S. (eds.) Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada. pp. 10450–10458. AAAI Press (2024). https://doi.org/10.1609/AAAI.V38I9.28913, https://doi.org/10.1609/aaai.v38i9.28913
7. Beierle, C., Dusso, O., Kern-Isberner, G.: Using answer set programming for a decision support system. In: Proceedings of the 8th International Conference (LPNMR 2005). LNCS, vol. 3662, pp. 374–378. Springer (2005). https://doi.org/10.1007/11546207_30
8. van Berkel, K., Straßer, C.: Towards deontic explanations through dialogue. In: Kampik, T., Cyras, K., Rago, A., Cocarascu, O. (eds.) Proceedings of the 2nd International Workshop on Argumentation for eXplainable AI co-located with the 10th International Conference on Computational Models of Argument (COMMA 2024), Hagen, Germany, September 16, 2024. CEUR Workshop Proceedings, vol. 3768, pp. 29–40. CEUR-WS.org (2024), https://ceur-ws.org/Vol-3768/paper8.pdf
9. van Berkel, K., Straßer, C., Zhou, Z.: Towards an argumentative unification of default reasoning. In: Reed, C., Thimm, M., Rienstra, T. (eds.) Computational Models of Argument - Proceedings of COMMA 2024, Hagen, Germany, September 18-20, 2024. Frontiers in Artificial Intelligence and Applications, vol. 388, pp. 313–324. IOS Press (2024). https://doi.org/10.3233/FAIA240331, https://doi.org/10.3233/FAIA240331
10. van Berkel, K., Straßer, C.: Reasoning with and about norms in logical argumentation. In: Toni, F., Polberg, S., Booth, R., Caminada, M., Kido, H. (eds.)

Frontiers in Artificial Intelligence and Applications: Computational Models of Argument, proceedings (COMMA22). vol. 353, pp. 332 – 343. IOS press (2022). https://doi.org/10.3233/FAIA220164
11. Bonatti, P.A.: Resolution for skeptical stable model semantics. Journal of Automated Reasoning **27**(4), 391–421 (2001). https://doi.org/10.1023/A:1011960831261, https://doi.org/10.1023/A:1011960831261
12. Bonatti, P.A., Pontelli, E., Son, T.C.: Credulous resolution for answer set programming. In: Proceedings of the 23rd AAAI Conference on Artificial Intelligence (AAAI 2008). pp. 418–423. AAAI Press (2008), http://www.aaai.org/Library/AAAI/2008/aaai08-066.php
13. Bondarenko, A., Dung, P.M., Kowalski, R.A., Toni, F.: An abstract, argumentation-theoretic approach to default reasoning. Artif. Intell. **93**, 63–101 (1997). https://doi.org/10.1016/S0004-3702(97)00015-5, https://doi.org/10.1016/S0004-3702(97)00015-5
14. Chew, L., de Colnet, A., Szeider, S.: ASP-QRAT: A conditionally optimal dual proof system for ASP. In: Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning (KR 2024) (2024). https://doi.org/10.24963/KR.2024/24
15. Comploi-Taupe, R., Francescutto, G., Schenner, G.: Applying incremental answer set solving to product configuration. In: Proceedings of the 26th ACM International Systems and Software Product Line Conference (SPLC 2022). pp. 150–155. ACM (2022). https://doi.org/10.1145/3503229.3547069
16. Dung, P.M.: 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). https://doi.org/10.1016/0004-3702(94)00041-X, https://doi.org/10.1016/0004-3702(94)00041-X
17. Eiter, T., Faber, W., Leone, N., Pfeifer, G.: The diagnosis frontend of the dlv system. AI Communications **12**(1-2), 99–111 (1999), http://content.iospress.com/articles/ai-communications/aic171
18. Eiter, T., Falkner, A.A., Schneider, P., Schüller, P.: ASP-based signal plan adjustments for traffic flow optimization. In: Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020). Frontiers in Artificial Intelligence and Applications, vol. 325, pp. 3026–3033. IOS Press (2020). https://doi.org/10.3233/FAIA200478
19. Eiter, T., Geibinger, T.: A sequent calculus for answer set entailment. In: Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2025, Montreal, Canada, August 16-22, 2025. pp. 4463–4472. ijcai.org (2025). https://doi.org/10.24963/IJCAI.2025/497, https://doi.org/10.24963/ijcai.2025/497
20. Eiter, T., Gottlob, G., Leone, N.: Abduction from logic programs: Semantics and complexity. Theoretical Computer Science **189**(1-2), 129–177 (1997). https://doi.org/10.1016/S0304-3975(96)00179-X, https://doi.org/10.1016/S0304-3975(96)00179-X
21. Erdem, E., Gelfond, M., Leone, N.: Applications of answer set programming. AI Magazine **37**(3), 53–68 (2016). https://doi.org/10.1609/aimag.v37i3.2678
22. Erdem, E., Lifschitz, V.: Tight logic programs. Theory and Practice of Logic Programming **3**(4-5), 499–518 (2003). https://doi.org/10.1017/S1471068403001765, https://doi.org/10.1017/S1471068403001765

23. Erdem, E., Oztok, U.: Generating explanations for biomedical queries. Theory and Practice of Logic Programming **15**(1), 35–78 (2015). https://doi.org/10.1017/S1471068413000598
24. Falkner, A., Friedrich, G., Schekotihin, K., Taupe, R., Teppan, E.C.: Industrial applications of answer set programming. KI - Künstliche Intelligenz **32**(2), 165–176 (2018). https://doi.org/10.1007/s13218-018-0548-6
25. Fandinno, J., Schulz, C.: Answering the "why" in answer set programming - A survey of explanation approaches. Theory and Practice of Logic Programming **19**(2), 114–203 (2019). https://doi.org/10.1017/S1471068418000534, https://doi.org/10.1017/S1471068418000534
26. Gebser, M., Kaminski, R., Kaufmann, B., Schaub, T.: Multi-shot ASP solving with clingo. Theory and Practice of Logic Programming **19**(1), 27–82 (2019). https://doi.org/10.1017/S1471068418000054
27. Gebser, M., Schaub, T.: Tableau calculi for logic programs under answer set semantics. ACM Transactions on Computational Logic **14**(2), 15:1–15:40 (2013). https://doi.org/10.1145/2480759.2480767, https://doi.org/10.1145/2480759.2480767
28. Inclezan, D.: An application of answer set programming to the field of second language acquisition. Theory and Practice of Logic Programming **15**(1), 1–17 (2015). https://doi.org/10.1017/S1471068413000653
29. Leone, N., Pfeifer, G., Faber, W., Eiter, T., Gottlob, G., Perri, S., Scarcello, F.: The DLV system for knowledge representation and reasoning. ACM Transactions on Computational Logic **7**(3), 499–562 (2006). https://doi.org/10.1145/1149114.1149117
30. Mints, G.: Cut-free formulations for a quantified logic of here and there. Ann. Pure Appl. Log. **162**(3), 237–242 (2010). https://doi.org/10.1016/J.APAL.2010.09.009, https://doi.org/10.1016/j.apal.2010.09.009
31. Mints, G.: Cut-free formulations for a quantified logic of here and there. Annals of Pure and Applied Logic **162**(3), 237–242 (2010). https://doi.org/10.1016/J.APAL.2010.09.009, https://doi.org/10.1016/j.apal.2010.09.009
32. Pearce, D.: A new logical characterisation of stable models and answer sets. In: Selected Papers of the 2nd International Workshop on Non-Monotonic Extensions of Logic Programming (NMELP 1996). LNCS, vol. 1216, pp. 57–70. Springer (1996). https://doi.org/10.1007/BFB0023801
33. Pearce, D.: Equilibrium logic. Annals of Mathematics and Artificial Intelligence **47**, 3–41 (2006)
34. Pearce, D., de Guzmán, I.P., Valverde, A.: A tableau calculus for equilibrium entailment. In: Proceedings of the 9th International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX 2000). LNCS, vol. 1847, pp. 352–367. Springer (2000). https://doi.org/10.1007/10722086_28
35. Pontelli, E., Son, T.C., El-Khatib, O.: Justifications for logic programs under answer set semantics. Theory and Practice of Logic Programming **9**(1), 1–56 (2009). https://doi.org/10.1017/S1471068408003633, https://doi.org/10.1017/S1471068408003633
36. Yli-Jyrä, A., Rankooh, M.F., Janhunen, T.: Pruning redundancy in answer set optimization applied to preventive maintenance scheduling. In: Proceedings of the 25th International Symposium on Practical Aspects of Declarative Languages (PADL 2023). pp. 279–294. Springer, Cham (2023)