Some Types of Equivalence for Logic Programs and Equilibrium Logic*

David Pearce[1] and Agustín Valverde[2]

[1] Universidad Rey Juan Carlos (Madrid, Spain)
Email: d.pearce@escet.urjc.es
[2] Universidad de Málaga (Spain)
Email: a_valverde@ctima.uma.es

Abstract. For a given semantics, two logic programs Π_1 and Π_2 can be said to be equivalent if they have the same intended models and strongly equivalent if for any program X, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are equivalent. Eiter and Fink have recently studied and characterised under answer set semantics a further, related property of *uniform* equivalence, where the extension X is required to be a set of atoms. We extend their main results to propositional theories in equilibrium logic and describe a tableaux proof system for checking the property of uniform equivalence. We also show that no new forms of equivalence are obtained by varying the logical form of expressions in the extension X.

1 Introduction

Concepts of program equivalence are important in both the theory and practice of logic programming. In terms of theory, knowing in general terms when logic programs are equivalent provides important information about their mathematical properties. In practical terms, knowing that two programs are equivalent may mean in certain contexts that one can be replaced by the other without loss. In answer set programming (ASP), the property of having the same answer sets can be viewed as the simplest kind of equivalence. Two programs with this property respond to queries in the same way: they have the same credulous and the same skeptical consequences. But they need not be intersubstitutable without loss in all contexts. To guarantee this property in the most general case, a notion of *strong* equivalence is needed. Two programs Π_1 and Π_2 are said to be *strongly equivalent* iff for any program X, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are equivalent, ie have the same answer sets. Strong equivalence in ASP is a powerful property that turns out to be easier to verify than ordinary equivalence. Lifschitz, Pearce and Valverde [10] showed that in answer set semantics programs are strongly equivalent if and only if they are equivalent in a certain non-classical logic called here-and-there (with strong negation), which we denote here by $\mathbf{N}_5$.[3] In the case of ordinary equivalence one has the harder task to verify that Π_1 and Π_2 have the same minimal $\mathbf{N}_5$ models of a special type, called equilibrium models, introduced in [11]. They correspond to answer sets.

* Partially supported by Junta de Andalucia project TIC-115.

[3] $\mathbf{N}_5$ is also a maximal logic with this property. Other logics capturing strong equivalence are described in [8].

Besides strong equivalence one may consider weaker concepts that still permit one program to be substituted for another in certain well-defined settings. One such notion is that of *uniform* or u-equivalence, defined as above but restricted to the case where X is a set of atoms. This concept is of interest when one is dealing with a fixed set of rules, or intensional knowledge base, and a varying set of facts or extensional knowledge component. It may also be relevant in other applications of ASP, eg in planning and diagnosis where there is a fixed background theory, and plans (resp. diagnostic explanations) are sequences (resp. sets) of atomic propositions. Uniform equivalent background theories will generate equivalent plans (resp. explanations).

The u-equivalence of logic programs under answer set semantics has recently been studied by Eiter and Fink [3]. For finite programs they show that u-equivalence can be neatly characterised in terms of certain maximal models. A weaker semantic property is demonstrated for the infinite case where such maximal models are not guaranteed to exist. They also look at several special classes of programs, such as Horn and head-cycle free programs, and provide complexity results for the general and several special cases. Here we extend the work of Eiter and Fink in several directions. First, their characterisations of uniform equivalence are proved for disjunctive programs using a notion of SE-model, introduced in [13]. They observe that this is essentially equivalent to the models of here-and-there logic and that the results generalise to programs with strong negation and even nested expressions. We shall prove the main characterisation results directly for theories in equilibrium logic using here-and-there models. This simplifies the proofs, generalises the results to the full propositional language and yields by the well-known properties of equilibrium logic the corresponding results for programs with strong negation and nested expressions without further ado. In the case of the characterisation applicable to infinite theories (Theorem 4 below), we strengthen the results slightly by simplifying part of the sufficiency condition. Secondly, we consider the question whether in the definition of uniform equivalence placing other restrictions on the logical form of sentences in the extension X yields new forms of equivalence lying 'between' uniform and strong equivalence. The answer is no. If rules involving implication are permitted in X, then strong equivalence is the appropriate concept. Otherwise, for implication-free formulas of any other logical type, the equivalence in question is equivalent to uniform. Thirdly, we consider a proof system for checking the property of uniform equivalence. For this, since we express equivalence using ordinary logical models in $\mathbf{N}_5$, we can adapt a tableau proof system for $\mathbf{N}_5$ that was studied in an earlier paper [12].

In [3] several examples are given of logic programs that are uniform but not strongly equivalent. A feature of equilibrium logic is that it allows one to represent programs with *conditional* rules, ie expressions of the form $\alpha \rightarrow (\beta \rightarrow \gamma)$ or $(\alpha \rightarrow \beta) \rightarrow \gamma$. It is interesting to consider when such rules are equivalent to ordinary programs with nested expressions, eg when can one replace $(p \rightarrow q) \rightarrow r$ by $(\neg p \vee q) \rightarrow r$. In general the last two expressions are u-equivalent but not strongly equivalent. In 4.2 we consider some cases of this kind.

2 Equilibrium Logic

We work throughout in the nonclassical logic of here-and-there with strong negation $\mathbf{N}_5$ and its nonmonotonic extension, equilibrium logic [11], which generalises answer set semantics for logic programs to arbitrary propositional theories, see eg [10]. We give only a very brief overview of equilibrium logic here. For more details the reader is referred to [11, 10, 12] and the logic texts cited below.

Formulas of $\mathbf{N}_5$ are built-up in the usual way using the logical constants: $\wedge$, $\vee$, $\rightarrow$, $\neg$, $\sim$, standing respectively for conjunction, disjunction, implication, weak (or intuitionistic) negation and strong negation. The axioms and rules of inference for $\mathbf{N}_5$ are those of intuitionistic logic (see eg [14]) together with:

1. the axiom schema $(\neg\alpha \rightarrow \beta) \rightarrow (((\beta \rightarrow \alpha) \rightarrow \beta) \rightarrow \beta)$, which characterises the 3-valued here-and-there logic of Heyting [7], and Gödel [4] (hence it is sometimes known as Gödel's 3-valued logic).
2. the following axiom schemata involving strong negation taken from the calculus of Vorob'ev [15, 16] (where '$\alpha \leftrightarrow \beta$' abbreviates $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$):

N1. $\sim(\alpha \rightarrow \beta) \leftrightarrow \alpha \wedge \sim\beta$	**N2.** $\sim(\alpha \wedge \beta) \leftrightarrow \sim\alpha \vee \sim\beta$
N3. $(\sim\alpha \vee \beta) \leftrightarrow \sim\alpha \wedge \sim\beta$	**N4.** $\sim\sim\alpha \leftrightarrow \alpha$
N5. $\sim\neg\alpha \leftrightarrow \alpha$	**N6.** (for atomic α) $\sim\alpha \rightarrow \neg\alpha$

The model theory of $\mathbf{N}_5$ is based on the usual Kripke semantics for Nelson's constructive logic $\mathbf{N}$(see eg. [5, 14]), but $\mathbf{N}_5$ is complete for Kripke frames $\mathcal{F} = \langle W, \leq\rangle$ (where as usual W is the set of point or worlds and $\leq$ is a partial-ordering on W) having exactly two worlds say h ('here') and t ('there') with $h \leq t$. As usual a *model* is a frame together with an assignment i that associates to each element of W a set of *literals*,[4] such that if $w \leq w'$ then $i(w) \subseteq i(w')$. An assignment is then extended inductively to all formulas via the usual rules for conjunction, disjunction, implication and (weak) negation in intuitionistic logic together with the following rules governing strongly negated formulas:

$$\begin{array}{l}
\sim(\varphi \wedge \psi) \in i(w) \text{ iff } \sim\varphi \in i(w) \text{ or } \sim\psi \in i(w) \\
\sim(\varphi \vee \psi) \in i(w) \text{ iff } \sim\varphi \in i(w) \text{ and } \sim\psi \in i(w) \\
\sim(\varphi \rightarrow \psi) \in i(w) \text{ iff } \varphi \in i(w) \text{ and } \sim\psi \in i(w) \\
\sim\neg\varphi \in i(w) \text{ iff } \sim\sim\varphi \in i(w) \text{ iff } \varphi \in i(w)
\end{array}$$

It is convenient to represent an $\mathbf{N}_5$-model as an ordered pair $\langle H, T\rangle$ of sets of literals, where $H = i(h)$ and $T = i(t)$ under a suitable assignment i. By $h \leq t$, it follows that $H \subseteq T$. Again, by extending i inductively we know what it means for an arbitrary formula φ to be true in a model $\langle H, T\rangle$.

A formula φ is true in a here-and-there model $\mathcal{M} = \langle H, T\rangle$ in symbols $\mathcal{M} \models \varphi$, if it is true at each world in $\mathcal{M}$. A formula φ is said to be *valid* in $\mathbf{N}_5$, in symbols $\models \varphi$, if it is true in all here-and-there models. Logical consequence for $\mathbf{N}_5$ is understood as follows: φ is said to be an $\mathbf{N}_5$-consequence of a set Π of formulas, written $\Pi \models \varphi$, iff

[4] We use the term 'literal' to denote an atom, or atom prefixed by strong negation.

for all models $\mathcal{M}$ and any world $w \in \mathcal{M}$, $\mathcal{M}, w \models \Pi$ implies $\mathcal{M}, w \models \varphi$. Equivalently this can be expressed by saying that φ is true in all models of Π. Further properties of $\mathbf{N}_5$ are studied in [9].

Equilibrium models are special kinds of minimal $\mathbf{N}_5$ Kripke models. We first define a partial ordering $\trianglelefteq$ on $\mathbf{N}_5$ models that will be used both to characterise the equilibrium property as well as the property of uniform equivalence.

Definition 1. *Given any two models $\langle H, T\rangle$, $\langle H', T'\rangle$, we set $\langle H, T\rangle \trianglelefteq \langle H', T'\rangle$ if $T = T'$ and $H \subseteq H'$.*

Definition 2. *Let Π be a set of $\mathbf{N}_5$ formulas and $\langle H, T\rangle$ a model of Π.*

1. *$\langle H, T\rangle$ is said to be* total *if $H = T$.*
2. *$\langle H, T\rangle$ is said to be an* equilibrium *model if it is minimal under $\trianglelefteq$ among models of Π, and it is total.*

In other words a model $\langle H, T\rangle$ of Π is in equilibrium if it is total and there is no model $\langle H', T\rangle$ of Π with $H' \subset H$. Equilibrium logic is the logic determined by the equilibrium models of a theory. It generalises answer set semantics in the following sense. For all the usual classes of logic programs, including normal, extended, disjunctive and nested programs, equilibrium models correspond to answer sets [11, 10]. The 'translation' from the syntax of programs to $\mathbf{N}_5$ propositional formulas is the trivial one, eg. a ground rule of an (extended) disjunctive program of the form

$$K_1 \vee \ldots \vee K_k \leftarrow L_1, \ldots L_m, notL_{m+1}, \ldots, notL_n$$

where the L_i and K_j are literals corresponds to the $\mathbf{N}_5$ sentence

$$L_1 \wedge \ldots \wedge L_m \wedge \neg L_{m+1} \wedge \ldots \wedge \neg L_n \rightarrow K_1 \vee \ldots \vee K_k$$

A set of $\mathbf{N}_5$ sentences is called a *theory*. Two theories are *equivalent* if they have the same equilibrium models.

3 Uniform Equivalence

We recall the definition of uniform equivalence and give a new proof of Theorem 3 of [3] for propositional theories in equilibrium logic. Additional motivation for the study of uniform equivalence and references to previous work is given in [3].

Definition 3. *Two theories Π_1 and Π_2 are said to be* uniform equivalent, *or u-equivalent for short, iff for any (empty or non-empty) set X of literals, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are equivalent, ie have the same equilibrium models.*

Note that if the theories are logic programs, this means they have the same answer sets.

We begin with some simple terminology. A model $\langle H, T\rangle$ is said to be *incomplete* if it is not total, ie. if $H \subset T$. A model $\langle H, T\rangle$ of a theory Π is said to be *maximal incomplete* (or just *maximal*) if it is incomplete and is maximal among models of Π under the ordering $\trianglelefteq$. In other words a model $\langle H, T\rangle$ of Π is maximal if for any model

$\langle H', T \rangle$ of Π, if $H \subset H'$ then $H' = T$. It is clear that if a theory Π is finite and has an incomplete model $\langle H, T \rangle$, then it has a maximal incomplete model $\langle H', T \rangle$ such that $H \subseteq H'$. However maximal models need not exist in the case that Π is an infinite theory. The following is straightforward.

Lemma 1. *If two theories Π_1 and Π_2 are u-equivalent, then they have the same total models.*

Note that theories with the same total models are equivalent in classical logic with strong negation (see Gurevich [5]).

Lemma 2. *If two finite theories Π_1 and Π_2 have the same total models and the same maximal incomplete models, then they are equivalent.*

Proof. Equilibrium models are total models with no incomplete 'submodels'.

Lemma 3. *If two finite theories Π_1 and Π_2 have the same maximal and total models then they are uniform equivalent.*

Proof. From Lemma 2 we have seen that theories Π_1 and Π_2 with the same total and maximal models are equivalent. It remains to show that they are also uniform equivalent. Thus, assume that Π_1 and Π_2 have the same total and maximal models and are therefore equivalent. Suppose for the contradiction that they are not u-equivalent. Then for some set X of literals, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are not equivalent, say the former has an equilibrium model $\langle T, T \rangle$ that is not an equilibrium model of $\Pi_2 \cup X$. Since the Π_1 and Π_2 have the same total models, clearly $\langle T, T \rangle \models \Pi_2 \cup X$. By assumption there is a model $\langle H, T \rangle$ of $\Pi_2 \cup X$ with $H \subset T$. Clearly $X \subseteq H$. Keeping T fixed, extend H to a maximal incomplete model $\langle H', T \rangle$ of Π_2 in T. It is evident that $\langle H', T \rangle$ is not a maximal model (or even model) of Π_1. If it were, since $X \subseteq H'$, it would be an incomplete model of $\Pi_1 \cup X$, contradicting the assumption that $\langle T, T \rangle$ is in equilibrium. Consequently, if two theories are not u-equivalent, they differ on some maximal model, contradicting the initial assumption. □

Lemma 4. *If two finite theories Π_1 and Π_2 are uniform equivalent, then they have the same maximal and total models.*

Proof. By Lemma 1, u-equivalent theories have the same total models. We will show that if they differ on maximal models, then they are not u-equivalent. Thus, suppose that Π_1 and Π_2 differ on some maximal incomplete model. Suppose for instance that Π_1 has a maximal incomplete model $\langle H, T \rangle$ that is not a maximal incomplete model of Π_2. We distinguish two cases as follows. Case (i): there is a model $\langle H', T \rangle$ of Π_2 with $H \subseteq H'$. Case (ii): there is no such model of Π_2. In each case we define non-equivalent extensions of Π_1 and Π_2.

Case (i). Since by assumption $\langle H, T \rangle$ is not a maximal model of Π_2, we can choose H' such that $H \subset H'$. Now select any element $A \in H' - H$. Set $X = H \cup \{A\}$. Then clearly $\langle T, T \rangle \models \Pi_1 \cup X$ and for any model $\langle J, T \rangle$ of $\Pi_1 \cup X$, clearly $H \subset J$. Hence it follows from the maximality of $\langle H, T \rangle$ among models of Π_1 that $\langle T, T \rangle$ must be an equilibrium model of $\Pi_1 \cup X$. But by inspection $\langle H', T \rangle$ is a model of $\Pi_2 \cup X$

and so $\langle T, T\rangle$ is not an equilibrium model of $\Pi_2 \cup X$, showing that Π_1 and Π_2 are not u-equivalent.

Case (ii). There is no model $\langle H', T\rangle$ of Π_2 with $H \subseteq H'$. In this case set $X = H$ and consider $\Pi_1 \cup X$ and $\Pi_2 \cup X$. Since Π_1 and Π_2 have the same total models it is clear that $\langle T, T\rangle \models \Pi_2 \cup X$. Moreover it is an equilibrium model of $\Pi_2 \cup X$ since, by assumption, there is no incomplete model $\langle H', T\rangle$ of Π_2 with $H \subset H'$. But clearly $\langle T, T\rangle$ is not an equilibrium model of $\Pi_1 \cup X$, since $\langle H, T\rangle \models \Pi_1$ and hence $\langle H, T\rangle \models \Pi_1 \cup H$. □

So we have shown Theorem 3 of [3] for arbitrary, finite theories:

Theorem 1. *Two finite theories are uniform equivalent if and only if they have the same total and maximal incomplete models.*

3.1 An extension

u-equivalence is typically of interest when one has a fixed set of program rules or a deductive database (the intensional part) and a variable set of facts or atomic propositions (or their strong negations) changing over time (the extensional part). Now suppose we allow the extensional part to contain other kinds of formulas, including say disjunctions and integrity constraints. What kinds of equivalences are obtained in such cases? We know from [10] that adding even the simplest kinds of proper rules with implication, of the form $p_i \rightarrow p_j$, brings us to the full case of strong equivalence. The next 'strongest' case would be to allow arbitrary (extended) boolean formulas in $(\wedge, \vee, \neg, \sim)$ but without implication.

Definition 4. *Two theories Π_1 and Π_2 are* uniform equivalent in the extended sense, *or u^+-equivalent, iff for any (empty or non-empty) set X of (implication-free) formulas in $(\wedge, \vee, \neg, \sim)$, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are equivalent, ie have the same equilibrium models.*

It turns out that the two notions are equivalent. To see this we need a simple lemma whose proof is left to the reader.

Lemma 5. *Let φ be any formula in $(\wedge, \vee, \neg, \sim)$. If $\langle H, T\rangle \models \varphi$ and $H \subseteq H' \subseteq T$ then $\langle H', T\rangle \models \varphi$.*

Theorem 2. *Two finite theories are u^+-equivalent if and only if they are u-equivalent.*

Proof. By definition, u-equivalence is a special case of u^+-equivalence. In particular if $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are equivalent for all X comprising boolean formulas, then they are clearly equivalent for all literal X. So u^+-equivalence implies u-equivalence. For the other direction, suppose Π_1 and Π_2 are u-equivalent, then by Lemma 4, they have the same maximal and total models. We show that this implies their u^+-equivalence. We proceed exactly as in the proof of Lemma 3. But for the contradiction we now suppose that for some set X of *boolean formulas*, $\Pi_1 \cup X$ and $\Pi_2 \cup X$ are not equivalent, say that $\langle T, T\rangle$ is an equilibrium model of $\Pi_1 \cup X$ but not of $\Pi_2 \cup X$. Again since they have the same total models we know that $\langle T, T\rangle \models \Pi_2 \cup X$, but since it is not an equilibrium model, there is a model $\langle H, T\rangle$ of $\Pi_2 \cup X$ with $H \subset T$. Let $\langle H', T\rangle$ be any maximal model of Π_2 such that $H \subset H'$. We need only check that $\langle H', T\rangle \models X$.

But this follows immediately from Lemma 5 above. So, as before, $\langle H', T\rangle \not\models \Pi_1$, since $\langle T, T\rangle$ is an equilibrium model of $\Pi_1 \cup X$. □

This result shows that on finite programs uniform and strong equivalence are the only concepts of their kind in ASP. In particular, varying the logical form of formulas permitted in the extension X of a program does not produce any new notion of equivalence: strong equivalence is the appropriate concept when proper rules containing implication are permitted, while all other cases are covered by uniform equivalence.

4 Many-valued semantics for $\mathbf{N}_5$

The Kripke semantics for $\mathbf{N}_5$ logic can be easily characterised using a many-valued approach, specifically with a five-valued logic. In this section we define this interpretation and then describe a five-valued tableau system to check inference.

The set of truth values in the many-valued characterisation is $\mathbf{5} = \{-2, -1, 0, 1, 2\}$ and 2 is the designated value. The connectives are interpreted as follows: $\wedge$ is the minimum function, $\vee$ is the maximum function, $\sim x = -x$,

$$x \rightarrow y = \begin{cases} 2 & \text{if either } x \leq 0 \text{ or } x \leq y \\ y & \text{otherwise} \end{cases} \quad \text{and} \quad \neg x = \begin{cases} 2 & \text{if } x \leq 0 \\ -x & \text{otherwise} \end{cases}$$

Any $\mathbf{N}_5$ model σ as a truth-value assignment can trivially be converted into a Kripke model $\langle H, T\rangle$, and *vice versa*. For example, if σ is an assignment and p is a propositional variable, then the corresponding Kripke model, denoted by $\mathcal{M}_\sigma$, is determined by the equivalences:

$$\begin{array}{llll} \sigma(p) = 2 \text{ iff } p \in H & & \sigma(p) = -1 \text{ iff } \sim p \in T, \sim p \notin H \\ \sigma(p) = 1 \text{ iff } p \in T, p \notin H & & \sigma(p) = -2 \text{ iff } \sim p \in H \\ \sigma(p) = 0 \text{ iff } p \notin T, \sim p \notin T & & \end{array}$$

The many-valued semantics and the Kripke semantics for $\mathbf{N}_5$ are equivalent. In other words, if Π is a set of formulas in $\mathbf{N}_5$ and ψ is a formula, then $\Pi \models \psi$ iff for every assignment σ in $\mathbf{N}_5$, if $\sigma(\varphi) = 2$ for every $\varphi \in \Pi$, then $\sigma(\psi) = 2$. Note too that assignments or truth-value interpretations can also be considered partially ordered by the $\trianglelefteq$ relation. We then say for example that an assignment σ is greater than or equal to an assignment τ, if $\mathcal{M}_\tau \trianglelefteq \mathcal{M}_\sigma$.

4.1 Tableau systems for $\mathbf{N}_5$

In [12] we introduced tableaux systems to study several properties in $\mathbf{N}_5$. Specifically, one system to check validity, another one to generate total models and another system based on auxiliary tableaux to check the equilibrium property for a specific model.[5] The systems are describe using *signed-formulas*, following the approach of [6]. The formulas in the tableau systems are labelled with sets of truth values: $S{:}\varphi$, $S \subset \mathbf{5}$; an assignment σ in $\mathbf{N}_5$ is a model of $S{:}\varphi$ if $\sigma(\varphi) \in S$. The initial tableau determines the

[5] Note that the tableaux system for $\mathbf{N}_5$ is already adequate for checking strong equivalence.

goal of the system; to study the satisfiability of a set of formulas $\{\varphi_1, \dots, \varphi_n\}$ and generate its models we use the following:

$$\textbf{Initial tableau for satisfiability} \quad \left\{ \begin{array}{l} \{2\}{:}\varphi_1 \\ \dots \\ \{2\}{:}\varphi_n \end{array} \right.$$

So we look for assignments, σ such that $\sigma(\varphi_i) = 2$ for all i. The **expansion rules** are common for every system and they must comply with the following property: *a model for any branch must be a model for the initial tableau and every model of the initial tableau is a model of some branch.* We show the rules for the connective $\rightarrow$ in Figure 1, the other connectives, $\wedge$, $\vee$, $\sim$ and $\neg$ are *regular* connectives, and the standard expansion rules can be applied [6].

1. If T is a tableau and T' is the tree obtained from T applying one of the expansion rules, then T' is a tableau. As usual in tableau systems for propositional logics, if a formula can be used to expand the tableau, then the tableau is expanded in every branch below the formula using the corresponding rule and the formula used to expand is marked and is no longer used.
2. A branch B in a tableau T is called *closed* if it contains a variable p with two signs, $S{:}p$, $S'{:}p$, such that $S \cap S' = \varnothing$, that is, the branch is unsatisfiable.
3. A branch B in a tableau T is called *finished* if the non-marked formulas are labelled propositional variables, *signed literals*. The branch is called *open* if it is non-closed and finished; in this case every model of the set of signed literals is a model of the initial set of formulas.
4. A tableau T is called *closed* if every branch is closed; in this case the initial set of formulas is unsatisfiable. The tableau is *open* if it has an open branch, (ie if it is non-closed). And it is *terminated* if every branch is either closed or open.

$$\frac{\{2\}{:}\varphi \rightarrow \psi}{\{-2,-1,0\}{:}\varphi \;\Big|\; \{2\}{:}\psi \;\Big|\; \begin{array}{c}\{-2,-1,0,1\}{:}\varphi \\ \{1,2\}{:}\psi\end{array}} \qquad \frac{\{-2,-1,0,1\}{:}\varphi \rightarrow \psi}{\begin{array}{c}\{1,2\}{:}\varphi \\ \{-2,-1,0\}{:}\psi\end{array} \;\Big|\; \begin{array}{c}\{2\}{:}\varphi \\ \{-2,-1,0,1\}{:}\psi\end{array}}$$

$$\frac{\{1,2\}{:}\varphi \rightarrow \psi}{\{-2,-1,0\}{:}\varphi \;|\; \{1,2\}{:}\psi} \qquad \frac{\{-2,-1,0\}{:}\varphi \rightarrow \psi}{\begin{array}{c}\{1,2\}{:}\varphi \\ \{-2,-1,0\}{:}\psi\end{array}} \qquad \frac{\{0,1,2\}{:}\varphi \rightarrow \psi}{\{-2,-1,0\}{:}\varphi \;|\; \{0,1,2\}{:}\psi}$$

$$\frac{\{-2,-1\}{:}\varphi \rightarrow \psi}{\begin{array}{c}\{1,2\}{:}\varphi \\ \{-2,-1\}{:}\psi\end{array}} \qquad \frac{\{-1,0,1,2\}{:}\varphi \rightarrow \psi}{\{-2,-1,0\}{:}\varphi \;|\; \{-1,0,1,2\}{:}\psi} \qquad \frac{\{-2\}{:}\varphi \rightarrow \psi}{\begin{array}{c}\{1,2\}{:}\varphi \\ \{-2\}{:}\psi\end{array}}$$

Fig. 1. Tableau expansion rules in $\mathbf{N}_5$ for $\rightarrow$

In the system, we look for non-closed terminated tableaux, that allows us to generate all the models of the initial set of formulas.

Theorem 3. *Let T be a non-closed terminated tableau for $\Pi = \{\varphi_1, \dots, \varphi_n\}$. Then, σ is a model of Π if and only if some branch of T is satisfiable by σ.*

4.2 Uniform equivalence

In the tableaux system for checking u^+-equivalence the following characterisation will be used. It is important to recall that it is valid also for infinite theories.

Theorem 4. *Two theories Π_1 and Π_2 with the same total models are u^+-equivalent if and only if the following conditions hold:*

(a) If $\langle H, T\rangle$ is an incomplete model of Π_1 then there exists H' such that $H \subseteq H' \subset T$ and $\langle H', T\rangle$ is a model of Π_2.
(b) If $\langle H, T\rangle$ is an incomplete model of Π_2 then there exists H' such that $H \subseteq H' \subset T$ and $\langle H', T\rangle$ is a model of Π_1.

Proof: ($\Leftarrow$) Assume that conditions (a) and (b) hold and Π_1 and Π_2 are not u^+-equivalent; let us assume that there exists a set of implication-free formulas, X, and an incomplete interpretation $\langle H, T\rangle$ such that $\langle T, T\rangle$ is an equilibrium model of $\Pi_1 \cup X$ and $\langle H, T\rangle \models \Pi_2 \cup X$. Obviously, $\langle H, T\rangle \models \Pi_2$ and $\langle H, T\rangle \models X$ and, by condition (b), there exists H' such that $H \subseteq H' \subset T$ and $\langle H', T\rangle \models \Pi_1$. Moreover, by lemma 5, $\langle H', T\rangle \models X$ and thus $\langle H', T\rangle \models \Pi_1 \cup X$, contradicting the equilibrium property of $\langle T, T\rangle$.

($\Rightarrow$) Assume that Π_1 and Π_2 are u^+-equivalent. We show (a); the proof for (b) is similar. Let H and T be such that $H \subset T$ and $\langle H, T\rangle \models \Pi_1$ and assume that for every H' such that $H \subseteq H' \subset T$ we have $\langle H', T\rangle \not\models \Pi_2$. Obviously, $\langle T, T\rangle \models H$, $\langle T, T\rangle \models \Pi_1$, thus $\langle T, T\rangle \models \Pi_2$, and $\langle T, T\rangle \models \Pi_2 \cup H$. Moreover, every model of H must be between $\langle H, T\rangle$ and $\langle T, T\rangle$ and none of them is a model of Π_2; so, there is no model for $\Pi_2 \cup H$ less than $\langle T, T\rangle$ and therefore this is in equilibrium. Since Π_1 and Π_2 are u^+-equivalent, $\langle T, T\rangle$ is also an equilibrium model of $\Pi_1 \cup H$, contradicting that $\langle H, T\rangle \models \Pi_1 \cup H$. □

This result is given for the case of disjunctive logic programs as Theorem 1 of [3]; note however that they state the conditions in (a) and (b) as 'iff' rather than 'if-then' conditions.

Auxiliary tableaux for u^+-equivalence checking An algorithm for u^+-equivalence checking is sketched as follows

1. The models of Π_1 and Π_2 are generated using the tableaux method in section 4.1.
2. If either some total model of Π_1 is not a model of Π_2 or some total model of Π_2 is not a model of Π_1, then Π_1 and Π_2 are not u^+-equivalent.
3. If Π_1 and Π_2 have the same total models, then we check if every incomplete model of Π_1 has a greater interpretation that is a model for Π_2, and also if every incomplete model of Π_2 has a greater interpretation that is a model for Π_1. For this we use auxiliary tableaux that are constructed for Π_1 and every incomplete model of Π_2, and for Π_2 and every incomplete model of Π_1.

The goal of the auxiliary tableau for a theory Π and an interpretation σ is to check if there is a model for Π greater than σ and so the initial tableau is the same but we need to add a new expansion rule to those introduced in section 4.1.

$$\frac{S{:}p}{(S \cap \omega_\sigma(p)){:}p}$$

where ω_σ is defined by: $\omega_\sigma(p) = \{\sigma(p)\}$ if $\sigma(p) \in \{-2, 0, 2\}$, $\omega_\sigma(p) = \{-2, -1\}$ if $\sigma(p) = -1$ and $\omega_\sigma(p) = \{1, 2\}$ if $\sigma(p) = 1$. With this new rule we restrict the admisible models for a branch to those that are greater than σ. Note that a new situation can arise in these auxiliary tableaux. After applying the new rule, the signed literal $\varnothing{:}p$ can appear; this literal is trivially unsatisfiable and the branch containing it is immediately closed. Thus we have the following property.

Theorem 5. *Given a theory Π and an assignment σ, there exists a model of Π greater than σ if and only if there is an open tableau for (Π, σ).*

Example: We are going to check that $\varphi_1 = r \to (\neg p \vee q)$ and $\varphi_2 = \neg r \vee (p \to q)$ are u^+-equivalent. The tableaux on the right allow us to generate the models of φ_1 and φ_2.

Comparing the sets of models, it is easy to conclude that the total models are the same for both formulas and we obtain two special models, σ and τ, defined as follows: $\sigma(p) = 2, \sigma(q) = 1, \sigma(r) = 1, \tau(p) = 1, \tau(q) = 1, \tau(r) = 2$; or, equivalently, $\sigma = \langle\{p\}, \{p.q, r\}\rangle$, $\tau = \langle\{r\}, \{p.q, r\}\rangle$.

$\{2\}{:}(r \to (\neg p \vee q))$ ✓

$\{-2,-1,0\}{:}r$ $\quad$ $\{2\}{:}(\neg p \vee q)$ ✓ $\quad$ $\{-2,-1,0,1\}{:}r$

$\{2\}{:}(\neg p)$ ✓ $\quad$ $\{2\}{:}q$ $\quad$ $\{1,2\}{:}(\neg p \vee q)$ ✓

$\{-2,-1,0\}{:}p$ $\quad$ $\{1,2\}{:}(\neg p)$ ✓ $\quad$ $\{1,2\}{:}q$

$\{-2,-1,0\}{:}p$

$\{2\}{:}(\neg r \vee (p \to q))$ ✓

$\{2\}{:}(\neg r)$ ✓ $\quad$ $\{2\}{:}(p \to q)$ ✓

$\{-2,-1,0\}{:}r$ $\quad$ $\{-2,-1,0\}{:}p$ $\quad$ $\{2\}{:}q$ $\quad$ $\{-2,-1,0,1\}{:}p$

$\{1,2\}{:}q$

These interpretations verify: $\sigma \models \varphi_1$, $\tau \models \varphi_2$, $\sigma \not\models \varphi_2$, $\tau \not\models \varphi_1$. Thus, φ_1 and φ_2 are not strongly equivalent, however they are uniform equivalent. To see this we construct the auxiliary tableaux for (φ_1, τ) and for (φ_2, σ). In the previous tableaux for φ_1 we apply the rules

$$\frac{S{:}p}{(S \cap \{1,2\}){:}p} \qquad \frac{S{:}q}{(S \cap \{1,2\}){:}q} \qquad \frac{S{:}r}{(S \cap \{2\}){:}r}$$

but the resulting auxiliary tableaux for (φ_1, τ) remains open

$\{2\}{:}(r \to (\neg p \vee q))$ ✓

$\{-2,-1,0\}{:}r$ ✓ $\quad$ $\{2\}{:}(\neg p \vee q)$ ✓ $\quad$ $\{-2,-1,0,1\}{:}r$

$\varnothing{:}r$ ✗ $\quad$ $\{2\}{:}(\neg p)$ ✓ $\quad$ $\{2\}{:}q$ $\quad$ $\{1,2\}{:}(\neg p \vee q)$ ✓

$\{-2,-1,0\}{:}p$ ✓ $\quad$ $\{1,2\}{:}(\neg p)$ ✓ $\quad$ $\{1,2\}{:}q$

$\varnothing{:}p$ ✗ $\quad$ $\{-2,-1,0\}{:}p$ ✓

$\varnothing{:}p$ ✗

For (φ_2, σ) the auxiliary tableau also remains open:

$$
\begin{array}{cccc}
 & \{2\}{:}(\neg r \vee (p \rightarrow q))\checkmark & & \\
\{2\}{:}(\neg r)\checkmark & \{2\}{:}(p \rightarrow q)\checkmark \quad \{2\}{:}q & & \{-2,-1,0,1\}{:}p\checkmark \\
\{-2,-1,0\}{:}r\checkmark & \{-2,-1,0\}{:}p\checkmark & & \{1,2\}{:}q \\
\varnothing{:}r & \varnothing{:}p & & \varnothing{:}p \\
\times & \times & & \times
\end{array}
$$

Similar examples are easy to generate. For instance, as mentioned in the introduction, one can readily verify that a rule such as $(p \rightarrow q) \rightarrow r$ with a conditional 'body' is uniform equivalent to the ordinary rule $(\neg p \vee q) \rightarrow r$. In [3] the complexity of uniform equivalence checking is studied. For the restriction to disjunctive programs the conclusion is that the problem is Π_2^P-complete. Our procedure based on tableau systems allow us to concluded that the problem for general theories is also Π_2^P-hard, because the generation of models is coNP-hard and the maximality checking (the oracle) is NP-hard. Therefore, the problem of uniform equivalence checking for general theories is also Π_2^P-complete.

5 Conclusions and Future Work

The uniform equivalence of logic programs is an important property that may, in specific contexts, allow one program to be substituted by another, perhaps syntactically simpler, program. In the general case, however, it is a harder property to check than strong equivalence. Here we have outlined a proof system for verifying uniform equivalence that applies to general propositional theories and consequently to any selected subclass of logic programs. The system is based on a semantical characterisation of uniform equivalence, similar to that of Eiter and Fink [3], but formulated for general propositional theories in terms of ordinary models in the logic $\mathbf{N}_5$ of here-and-there with strong negation. As a by-product we were able to show that varying the logical form of formulas in the program extensions does not change the properties of uniform equivalence, and hence there are no other types of equivalence of this kind situated 'between' uniform and strong equivalence.

We already remarked that uniform equivalence may be relevant in application areas of answer set programming, such as diagnosis and planning, where abductive methods are used. It is clear that if all and only atoms are permitted as abducibles, then uniform equivalent programs have the same (abductive) explanatory power, since every abductive explanation in one program is an equivalent explanation in the other.[6] Our results show that this is still the case when abducibles are allowed to be any implication-free formulas. However there appears to be a useful concept of abductive equivalence weaker than uniform equivalence. Two programs Π_1 and Π_2 would be equivalent in this weaker sense if for any X there exists a Y such that $\Pi_1 \cup X$ and $\Pi_2 \cup Y$ are equivalent, where X, Y are suitably restricted (but possibly different) sets of formulas (eg

[6] We are assuming here the absence of additional syntactic restrictions such as minimality conditions.

literals). It remains to be seen whether this concept is genuinely weaker and whether it can be characterised in simple, semantic terms.

Efficiency issues also arise from our basic algorithms. The exhaustive generation of models can be computationally hard; in the previous example, each formula has 111 distinct models, though ultimately we only need to manipulate two models. So, further refinements of the algorithm remain be investigated in the future [2, 1].

References

1. G. Aguilera Venegas, I. Perez de Guzman, M. Ojeda Aciego, and A. Valverde. Reducing signed propositional formulas. *Soft Computing—A Fusion of Foundations, Methodologies and Applications*, 2(4):157–166, 1999.
2. I.P. de Guzmán, M. Ojeda-Aciego, and A. Valverde. *Multiple-Valued Tableaux with Δ-reductions*. In *Proceedings of IC-AI'99*, pages 177–183, 1999.
3. T. Eiter and M. Fink. Uniform equivalence of logic programs under the stable model semantics. In *Int. Conf. in Logic Programming, ICLP'03*, Mumbay, India. Springer, 2003.
4. K. Gödel. Zum intuitionistischen aussagenkalkül. *Anzeiger der Akademie der Wissenschaften Wien, mathematisch, naturwissenschaftliche Klasse*, 69:65–66, 1932.
5. Y. Gurevich. Intuitionistic logic with strong negation. *Studia Logica*, 36(1–2):49–59, 1977.
6. R. Hähnle. *Automated Deduction in Multiple-Valued Logics*. Oxford University Press, 1993.
7. A. Heyting. Die formalen regeln der intuitionistischen logik. *Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch-mathematische Klasse*, pages 42–56, 1930.
8. D. De Jongh and L. Hendriks. Characterization of strongly equivalent logic programs in intermediate logics. *Theory and Practice of Logic Programming*, 3(3):259–270, 2003.
9. M. Kracht. On extensions of intermediate logics by strong negation. *Journal of Philosophical Logic*, 27(1):49–73, 1998.
10. V. Lifschitz, D. Pearce, and A. Valverde. Strongly equivalent logic programs. *ACM Transactions on Computational Logic*, 2(4):526–541, 2001.
11. D. Pearce. A new logical characterization of stable models and answer sets. In *Non-Monotonic Extensions of Logic Programming, NMELP 96*, LNCS 1216, pages 57–70. Springer, 1997.
12. D. Pearce, I.P. de Guzmán, and A. Valverde. A tableau calculus for equilibrium entailment. In *Automated Reasoning with Analytic Tableaux and Related Methods, TABLEAUX 2000*, LNAI 1847, pages 352–367. Springer, 2000.
13. H. Turner. Strong equivalence for logic programs and default theories (made easy). In *Proc. of the Logic Programming and Nonmonotonic Reasoning, LPNMR'01*, LNAI 2173, pages 81–92. Springer, 2001.
14. D. van Dalen. Intuitionistic logic. In *Handbook of Philosophical Logic, Volume III: Alternatives in Classical Logic*, Dordrecht, 1986. D. Reidel Publishing Co.
15. N. N. Vorob'ev. A constructive propositional calculus with strong negation (in Russian). *Doklady Akademii Nauk SSR*, 85:465–468, 1952.
16. N. N. Vorob'ev. The problem of deducibility in constructive propositional calculus with strong negation (in Russian). *Doklady Akademii Nauk SSR*, 85:689–692, 1952.