Strong and Uniform Equivalence Revisited

Zeynep G. Saribatur, Stefan Woltran

Institute of Logic and Computation, TU Wien

Abstract. We introduce a new notion of equivalence for logic programs that has strong and uniform equivalence as corner cases. We provide a characterisation in the spirit of SE-models together with a complexity analysis which sheds new light on the known difference between strong and uniform equivalence.

1 Introduction

Strong equivalence [2] between logic programs P and Q asks whether, for each further program R, the answer sets of $P \cup R$ and $Q \cup R$ coincide. We adapt this notion in the sense that the program R can be seperated into several parts over different alphabets. From a practical perspective, this can be motivated by a situation where P and Q receive input from different sources (each with their own alphabet) - the traditional notion of strong equivalence can be too restrictive in such a setting. From a theoretical perspective, as we will see, this allows to parameterize strong and uniform equivalence [1] (here R is restricted to facts) as special cases.

2 Preliminaries

We consider a countable universe $\mathcal{U}$ of atoms. A *signature* is any set of alphabets $\{A_1, \ldots, A_n\}$ where $\bigcup_i A_i = \mathcal{U}$. We consider the class of finite generalized disjunctive programs (i.e., with possibly double negated body atoms), simply referred as "programs", as well as the class of unary programs which allows only for facts $a \leftarrow$ and rules of the form $a \leftarrow b$. We say that a program P is given over a signature $\Sigma = \{A_1, \ldots, A_n\}$ if for each rule $r \in P$ there exists an i such that r has all its atoms from A_i. For an interpretation Y and alphabet A, we use $Y|_A$ as shorthand for $Y \cap A$. Answer sets of a program P (in symbols $AS(P)$) are defined via the GL-reduct as usual. We recall the notion of an SE-model of a program P: $(X, Y) \in SE(P)$, if $X \subseteq Y \subseteq \mathcal{U}$, $Y \models P$, and $X \models P^Y$. An SE-model (X, Y) of P is called UE-model of P if for each X' with $X \subset X' \subset Y$, $(X', Y) \notin SE(P)$. Strong (resp. uniform) equivalence between programs P and Q holds exactly if $SE(P) = SE(Q)$ (resp. $UE(P) = UE(Q)$) [2, 1].

3 Main Results

We define our new notion of equivalence. Proofs of the forthcoming results and some examples are provided in the appendix.

Definition 1. *Let P and Q be programs and Σ be a signature. We call P and Q Σ-equivalent ($P \equiv_\Sigma Q$) if for each R over Σ, $AS(P \cup R) = AS(Q \cup R)$ holds.*

It is easy to see that for the signature $\Sigma_s = \{\mathcal{U}\}$, Σ_s-equivalence coincides with strong equivalence. We will later observe that for $\Sigma_u = \{\{a\} \mid a \in \mathcal{U}\}$, Σ_u-equivalence coincides with uniform equivalence.

We now provide the objects for our forthcoming main characterization result (observe for signatures Σ_s and Σ_u the relation to SE- and UE-models).

Definition 2. *Let P be a program and $\Sigma = \{A_1, ..., A_n\}$ be a signature. An SE-model (X, Y) of P is called Σ-model of P (in symbols $(X, Y) \in SE_\Sigma(P)$) if, in addition, for each X' such that $X' \models P^Y$ and $X \subset X' \subset Y$, there exists an i such that $X|_{A_i} \subset X'|_{A_i} \subset Y|_{A_i}$.*

Theorem 1. *Given a signature Σ, and two programs P and Q, the following propositions are equivalent: (i) $P \equiv_\Sigma Q$; (ii) for each unary program R over Σ, $AS(P \cup R) = AS(Q \cup R)$; (iii) $SE_\Sigma(P) = SE_\Sigma(Q)$.*

Note that thanks to propery (ii), it follows directly that Σ_u-equivalence matches uniform equivalence, since unary rules of the form $a \leftarrow a$ have no semantical effect. We conclude with the following complexity result and recall that, while strong equivalence is known to be coNP-complete, uniform equivalence is Π_2^P-complete (in the following result, we tacitly consider $\mathcal{U}$ to be finite and a superset of the atoms occurring in P and Q).

Theorem 2. *Given programs P, Q, and signature $\Sigma = \{A_1, \dots A_n\}$ over universe $\mathcal{U}$, deciding $P \equiv_\Sigma Q$ is Π_2^P-complete. If we bound the dimension n of Σ to a fixed constant, the problem is* coNP*-complete.*

4 Discussion

We analysed the known differences between strong and uniform equivalence in terms of characterisation and computational complexity by providing a new equivalence schema. Compared to the approach [3], where restricting alphabets in heads and bodies of rules in the context program has been proposed, we did so by partitioning the rules into different alphabets via a given signature. Future work will tackle the relativized case where the signature $\Sigma = \{A_1, \dots, A_n\}$ does not cover the entire universe, i.e. where we can have $\bigcup_i A_i \subset \mathcal{U}$.

Acknowledgments

This work has been supported by the Austrian Science Fund (FWF) projects T-1315 and 10.557766/COE12.

References

1. Eiter, T., Fink, M.: Uniform equivalence of logic programs under the stable model semantics. In: International Conference on Logic Programming. pp. 224–238. Springer (2003)
2. Lifschitz, V., Pearce, D., Valverde, A.: Strongly equivalent logic programs. ACM Transactions on Computational Logic **2**(4), 526–541 (Oct 2001)
3. Woltran, S.: A common view on strong, uniform, and other notions of equivalence in answer-set programming. Theory Pract. Log. Program. **8**(2), 217–234 (2008). https://doi.org/10.1017/S1471068407003250, https://doi.org/10.1017/S1471068407003250

A Appendix

A.1 Examples

For the sake of our examples, we take a finite universe $\mathcal{U} = \{a, b, c, d\}$.

Consider signature $\Sigma = \{A_1, A_2\}$ with $A_1 = \{a, b\}$ and $A_2 = \{c, d\}$, and programs P, Q with

$$SE(P) = \{(\emptyset, abcd), (ab, abcd), (abcd, abcd)\}$$
$$SE(Q) = \{((ab, abcd), (abcd, abcd)\}$$

Hence, P and Q are not strongly equivalent. In order to construct an R such that $AS(P \cup R) \neq AS(Q \cup R)$, we need a rule like $c \leftarrow b$ which does not affect SE-model $(\emptyset, abcd)$ of P but "kills" SE-model $(ab, abcd)$. On the other hand, there is no chance to exclude $(ab, abcd)$ with rules over a, b (or c, d) only – i.e. over Σ – without killing the total model $(abcd, abcd)$ or removing $(\emptyset, abcd)$ at the same time. Indeed, P and Q are Σ-equivalent as can be checked via our characterisation. The SE-model $(\emptyset, abcd)$ is not Σ-model of P since $\{a, b\}_{A_1} = \{a, b, c, d\}|_{A_1}$ and $\emptyset|_{A_2} = \{a, b\}|_{A_2}$; thus the necessary condition for $(\emptyset, abcd)$ being Σ-model of P does neither hold for A_1 nor for A_2.

This shows that $\{\{a, b\}, \{c, d\}\}$-equivalence is indeed a different concept than $\{\{a, b, c, d\}\}$-equivalence (i.e., strong equivalence).

One might ask whether $\{\{a, b\}, \{c, d\}\}$-equivalence amounts to checking for relative strong equivalence with respect to $\{a, b\}$ and relativized strong equivalence wrt $\{c, d\}$ separately. Without going too much into detail, let us see another example. We use Σ as before and programs P, Q with the following SE-models.

$$SE(P) = \{(\emptyset, abcd), (a, abcd), (c, abcd), (abcd, abcd)\}$$
$$SE(Q) = \{((a, abcd), (c, abcd), (abcd, abcd)\}$$

It can be checked that $AS(P \cup R) \neq AS(Q \cup R)$ for $R = \{b \leftarrow a, d \leftarrow c\}$ – note that R satisfies the restriction for being a context program over Σ. In fact, it can be checked that $(\emptyset, abcd)$ remains Σ-model of P, thus the Σ-models of P and Q do not coincide, as expected. However, P and Q are strongly equivalent relative to both $\{a, b\}$ and $\{c, d\}$. This can be seen by using relativized SE-models. Here it appears that $(c, abcd)$ becomes $(\emptyset, abcd)$ for alphabet $\{a, b\}$, and $(a, abcd)$ becomes $(\emptyset, abcd)$ for alphabet $\{c, d\}$. Hence, for both alphabets, the relativized SE-models of P and Q coincide.

A.2 Proof of Theorem 1

We start with a technical lemma.

Lemma 1. *Given program P, signature Σ, and $X \subset Y$. If $(X, Y) \in SE(P)$, there exists X' with $X \subseteq X' \subset Y$ such that $(X', Y) \in SE_{\Sigma}(P)$.*

Proof. If $(X, Y) \in SE_\Sigma(P)$ we are done, so suppose this is not the case. Otherwise, by definition, there exists some X' with $X \subset X' \subset Y$ and $X' \models P^Y$. Hence $(X', Y) \in SE(P)$. It is clear that a subset-maximal such X' also satisfies the condition for $(X', Y) \in SE_\Sigma(P)$.

We now proceed with the proof of the statement, i.e. given a signature Σ, and two programs P and Q, the following propositions are equivalent:

(i) $P \equiv_\Sigma Q$;
(ii) for each unary program R over Σ, $AS(P \cup R) = AS(Q \cup R)$;
(iii) $SE_\Sigma(P) = SE_\Sigma(Q)$.

(i)⇒(ii) follows directly.

(ii)⇒(iii): We prove by contraposition. Suppose $SE_\Sigma(P) \neq SE_\Sigma(Q)$. W.l.o.g. assume $(X, Y) \in SE_\Sigma(P) \setminus SE_\Sigma(Q)$.

Case 1: $X = Y$: Then $Y \in AS(P \cup Y)$ but $Y \notin AS(Q \cup Y)$. Note that the set of facts Y fulfils the necessary condition for being a unary program over Σ.

Case 2: $X \subset Y$. Let us separate the following subcases:

Case 2.1: there is a X' with $X \subset X' \subset Y$ and $(X', Y) \in SE(Q)$ such that for all i, either $X|_{A_i} = X'|_{A_i}$ or $X'|_{A_i} = Y|_{A_i}$. For this case let

$$R = X' \cup \bigcup_i \{a \leftarrow b \mid a, b \in (Y \setminus X) \cap A_i\}.$$

We have $Y \in AS(P \cup R)$ but $Y \notin AS(Q \cup R)$. $Y \notin AS(Q \cup R)$ is clear since $X' \models Q^Y$ and $X' \models R = R^Y$. In order to show $Y \in AS(P \cup R)$, suppose the contrary. Then, there must exist Z with $X \subseteq Z \subset Y$ such that $Z \models P^Y$ and $Z \models R^Y = R$. From the latter we derive that $X \subset Z$ (since $X' \subseteq R$ and $X \subset X'$). However, since $(X, Y) \in SE_\Sigma(P)$, we know that for each such Z there exists an i such that $X|_{A_i} \subset Z|_{A_i} \subset Y|_{A_i}$. But then Z cannot be model of the subprogram $\{a \leftarrow b \mid a, b \in (Y \setminus X) \cap A_i\}$.

Case 2.2: otherwise, due to $(X, Y) \notin SE_\Sigma(Q)$, $(X, Y) \notin SE(Q)$. For this case let

$$R = X \cup \bigcup_i \{a \leftarrow b \mid a, b \in (Y \setminus X) \cap A_i\}.$$

We have $Y \in AS(Q \cup R)$ but $Y \notin AS(P \cup R)$. The latter is easily verified. To see the former, let X' with $X \subset X' \subset Y$. If $(X', Y) \in SE(Q)$, then there exists an i with $X|_{A_i} \subset X'|_{A_i}$ and $X'|_{A_i} \subset Y|_{A_i}$ (otherwise we would be in Case 2.1). But then, $X' \not\models \{a \leftarrow b \mid a, b \in (Y \setminus X) \cap A_i\}$ which is part of R.

Observe that in both cases R fullfils the necessary condition for being a unary program over Σ.

(iii)⇒(i): Let R be a context program over Σ, such that there is an Y with $Y \in AS(P \cup R)$ and $Y \notin AS(Q \cup R)$ (the other case symmetric). For $Y \notin AS(Q \cup R)$ we identify two cases:

Case 1: $Y \not\models Q \cup R$. Since $Y \models P \cup R$ we have $(Y,Y) \in SE_\Sigma(P)$ and $Y \not\models Q$. From the latter $(Y,Y) \notin SE_\Sigma(Q)$. Thus assume in what follows that total Σ-models of P and Q coincide.

Case 2: There is an $X \subset Y$ such that $X \models (Q \cup R)^Y = Q^Y \cup R^Y$. It follows that $X \not\models P^Y$, and thus $(X,Y) \notin SE_\Sigma(P)$. In case $(X,Y) \in SE_\Sigma(Q)$ we are done, so assume this is not the case. Since $X \models Q^Y$, there exists $X \subset X' \subset Y$ with $X' \models Q^Y$ such that for all i, either $X|_{A_i} = X'|_{A_i}$ or $X'|_{A_i} = Y|_{A_i}$. Observe that $X' \models R^Y$ (this can be seen as follows: if $X' \not\models R^Y$ there is a rule $r \in R$ with $X' \not\models r^Y$, i.e. $H(r) \cap X' = \emptyset$; since r is over some alphabet A_i, we know that thus either $H(r) \cap X = \emptyset$ or $H(r) \cap Y = \emptyset$; both a contradiction to previous assumptions). Thus $X' \not\models P^Y$, and thus $(X',Y) \notin SE_\Sigma(P)$. We can reiterate this argument now. By Lemma 1, at some point we arrive in a situation where $(X,Y) \notin SE_\Sigma(P)$ but $(X,Y) \in SE_\Sigma(Q)$.

A.3 Proof of Theorem 2

The lower bounds follow from known complexity results for strong and respectively uniform equivalence. For the upper bounds we make use of the charactersation of Theorem 1 and the result below. The bounds then follow by deciding the complementary problem via guessing a pair (X,Y) and checking whether $(X,Y) \in SE_\Sigma(P)$ and $(X,Y) \notin SE_\Sigma(Q)$ holds, or $(X,Y) \notin SE_\Sigma(P)$ and $(X,Y) \in SE_\Sigma(Q)$ holds.

Lemma 2. *Given P, (X,Y) and $\Sigma = \{A_1, ..., A_n\}$ a signature over $\mathcal{U}$. Deciding $(X,Y) \in SE_\Sigma(P)$ is (a) in* coNP *in general; (b) in P if n is bounded to constant.*

Proof. (a) The complementary problem is in NP — checking $(X,Y) \notin SE(P)$ can be done efficiently; for the additional condition it suffices to guess X' with $X' \models P^Y$ and $X \subset X' \subset Y$, and then loop over all i and check whether $X|_{A_i} = X'|_{A_i}$ or $X'|_{A_i} = Y|_{A_i}$ (note that the number of iterations is polynomial in input).

(b) We provide an algorithm for checking $(X,Y) \notin SE_\Sigma(P)$. We first check for potential violation of $(X,Y) \in SE(P)$ which is in P. Then we loop over all non-empty subsets I of the powerset of $\{1, \dots, n\}$ and for each such i we consider $X'_I = X \cup \bigcup_{i \in I}(Y \cap A_i)$. We then check whether in $X'_I \models P^Y$ and $X \subset X'_I \subset Y$. If we find such an X' we know $(X,Y) \notin SE_\Sigma(P)$. Since n is constant, the number of possible I is also constant. Moreover, all possible $X \subset X' \subset Y$, such that $X|_{A_i} = X'|_{A_i}$ or $X'|_{A_i} = Y|_{A_i}$ for all i are covered by some X'_I.