Towards Coinductive Theory Exploration in Horn Clause Logic: Position Paper

Ekaterina Komendantskaya
School of Mathematical and Computer Sciences
Heriot-Watt University, UK
ek19@hw.ac.uk

Yue Li
School of Mathematical and Computer Sciences
Heriot-Watt University, UK
yl55@hw.ac.uk

Coinduction occurs in two guises in Horn clause logic: in proofs of self-referencing properties and relations, and in proofs involving construction of (possibly irregular) infinite data. Both instances of coinductive reasoning appeared in the literature before, but a systematic analysis of these two kinds of proofs and of their relation was lacking. We propose a general proof-theoretic framework for handling both kinds of coinduction arising in Horn clause logic. To this aim, we propose a coinductive extension of Miller et al's framework of *uniform proofs* and prove its soundness relative to coinductive models of Horn clause logic.

1 Problem Statement

Coinductive proof methods have seen major developments in the last decade, and are reaching the point of maturity when coinductive proofs are used and implemented on par with inductive proofs. This step-change is facilitated by results from several research areas: coalgebra, fixed point theory, type theory, proof theory, automated deduction. In this abstract, we discuss a new coinductive approach to Horn clause logic.

A Horn clause fragment of FOL, named *fohc*, is given by the following syntax:

$$D ::= A \mid G \supset D \mid D \wedge D \mid \forall Var\ D$$
$$G ::= \top \mid G \wedge G \mid G \vee G \mid \exists Var\ G$$

where A stands for the set of atomic first-order formulae of a given signature, and D and G – for sets of definite Horn clauses and definite Horn goals, respectively. A *theory* Γ is a set of D-formulae.

First coinductive interpretation to Horn clause logic was given by Apt and van Emden in the 80s: The *greatest complete Herbrand model* for a theory Γ is the largest set of finite and infinite ground terms *coinductively entailed* by Γ's clauses.

Example 1 *Consider the three Horn clause theories Γ_1, Γ_2 and Γ_3 in Table 1. None of them has a meaningful inductive interpretation. However, they all have greatest (complete) Herbrand models, as Table 1 shows. These models define their coinductive interpretation. Notice how, depending on the clause structure, the models will differ: they may be given by finite sets of finite atomic formulae (for Γ_1), or infinite sets of finite and infinite formulae (Γ_2), or finite sets of infinite formulae (Γ_3). Note that Γ_3 is a prototypical example of a productive stream definition [3]: just substitute f by a stream constructor $cons(a, \_)$ to obatin a definition of the infinite stream of a's. Only one infinite term satisfies Γ_3.*

It has always been problematic to match the greatest complete Herbrand models with equally rich operational semantics. It is long known that infinite (SLD)-resolution derivations correspond to coinductive models [4]. Some infinite derivations may be terminated if a loop invariant (also known as *coinductive invariant*) is found. The problem is then to automate the discovery of coinductive invariants. To illustrate

T. Kahsai and G. Vidal (Eds.): 5th Workshop on
Horn Clauses for Verification and Synthesis 2018 (HCVS'18).
EPTCS 278, 2018, pp. 27–33, doi:10.4204/EPTCS.278.5

hohc theory:	Γ_1 : 1. $\forall x \;\; p(x) \supset p(x)$	Γ_2 : 2. $\forall x\; p(f\;x) \supset p(x)$	Γ_3 : 3. $\forall x\; p(x) \supset p(f\;x)$
greatest complete Herbrand model:	$\{\mathtt{p(a)}\}$	$\{\mathtt{p(a)}, \mathtt{p(f(a))}, \mathtt{p(f(f(a))}, \ldots, \mathtt{p(f(f\ldots)}\}$	$\{\mathtt{p(f(f\ldots)}\}$

Table 1: **Examples of greatest (complete) Herbrand models for fohc theories Γ_1, Γ_2, Γ_3.** We add an arbitrary constant symbol a to the signature, in order to have ground instances of formulae in the models.

how difficult this may prove to be, consider the following example. Given our three theories Γ_1, Γ_2 and Γ_3, suppose we want to prove a property $p(a)$ by coinduction.

Example 2 *For Γ_1, we will observe the following resolution steps:*

$$p(a) \stackrel{apply\ 1}{\longrightarrow} p(a) \stackrel{apply\ CI_1}{\longrightarrow} \checkmark$$

Clearly, $p(a)$ is the coinductive invariant (denoted as $CI_1 = p(a)$), the derivation is cyclic, and we can terminate soundly by noting this fact. Note how Γ_1's model in Table 1 agrees with this conclusion. Coinductive logic programming (CoLP) [2] handles such cases well: its method of loop detection is able to find that $p(a)$ is looping and thus find the correct coinductive hypothesis.

However, it is entirely possible that an environment Γ entails $p(a)$, yet $p(a)$ does not occur as an invariant in its infinite derivation.

Example 3 *Consider Γ_2. Trying to replicate the coinductive proof of Example 2 with coinductive invariant $p(a)$ would not work, as the coinductive invariant will not apply at any stage (the derivation does not have cycles):*

$$p(a) \stackrel{apply\ 2}{\longrightarrow} p(f\;a) \stackrel{apply\ 2}{\longrightarrow} p(f\;(f\;a)) \longrightarrow \ldots$$

A valid (as well as useful) coinductive invariant in this proof is $CI_2 = \forall x\; p(x)$. So, given a suitable calculus*, we can first coinductively prove $\Gamma_2 \vdash \forall x\; p(x)$, and then obtain $\Gamma_2 \vdash p(a)$ as a corollary. Note, however, that the formula $\forall x\; p(x)$ does not satisfy the syntax of a goal formula in* fohc*. And note also that loop-detection methods like CoLP [2] cannot handle such cases: no loop (i.e. no unifying subgoals) can be found in this derivation.*

Generally, discovering a suitable coinductive invariant may be a difficult task. Consider the following example, inspired by a similar example in [1].

Example 4 *Suppose we want to prove $p(a)$ given the theory Γ_4 :*

4.1. $\forall x\; p(f\;x) \wedge q(x) \supset p(x)$
4.2. $q(a)$
4.3. $\forall x\; q(x) \supset q(f\;x)$

Its greatest complete Herbrand model is given by:

$$\{\mathtt{p(a)}, \mathtt{p\;(f\;a)}, \mathtt{p(f\;(f\;a))}, \ldots, \mathtt{p(f(f(\ldots)))}$$
$$\mathtt{q(a)}, \mathtt{q(f\;a)}, \mathtt{q(f\;(f\;a))}, \ldots, \mathtt{q\;(f(f(\ldots)))}\;\}$$

Thus, $p(a)$ we seek to prove is coinductively valid.

It will give the following resolution trace:

$$p(a) \stackrel{apply\ 4.1}{\longrightarrow} p(f\;a) \wedge q(a) \stackrel{apply\ 4.2}{\longrightarrow} p(f\;a) \stackrel{apply\ 4.1}{\longrightarrow} p(f\;f\;a) \wedge q(f\;a) \stackrel{apply\ 4.3}{\longrightarrow} \ldots$$

The coinductive invariant $CI_1 = p(a)$ *will not apply here, despite* $p(a)$ *being in the model of* Γ_4*. Actually, neither* $CI_1 = p(a)$ *nor* $CI_2 = \forall x\ p(x)$ *would work as a suitable coinductive invariant. However, given a suitable calculus, we would be able to coinductively prove* $\Gamma_4 \vdash \forall x\ (q(x) \supset p(x))$*, from which* $\Gamma_4 \vdash p(a)$ *can be proven as a corollary. Again, note that* $CI_3 = \forall x\ (q(x) \supset p(x))$ *cannot be a goal formula in* fohc*, so we will need a different language for reasoning about coinductive invariant of the proof of* $\Gamma_4 \vdash p(a)$*.*

Finding a suitable coinductive invariant in a goal-directed proof search may require coming up with recursive terms on top of finding a suitable shape for the coinductive invariant, as the next example shows:

Example 5 *Given a theory* Γ_3 *from Table 1, the goal-directed search by resolution will result in a derivation:*

$$\underline{p(x)} \stackrel{apply\ 3,[x \mapsto f(x_1)]}{\longrightarrow} p(x_1) \stackrel{apply\ 3,[x_1 \mapsto f(x_2)]}{\longrightarrow} p(x_2) \longrightarrow \ldots$$

None of the sub-goals can serve as a suitable coinductive invariant. The correct coinductive invariant in this derivation is $p(\mathit{fix}\ \lambda x.f\ x)$*, where the fixpoint term* $\mathit{fix}\ \lambda x.f\ x$ *should be intuitively understood as a recursive definition for an infinite term* $(f(f\ldots))$*. Compare also with* Γ_3*'s model in Table 1, and its only inhabitant* `p(f(f...)`*).*

Thus, we would like to coinductively prove $\Gamma_3 \vdash p(\mathit{fix}\ \lambda x.f\ x)$ *in a suitable logic, and then get* $\Gamma_3 \vdash \exists x, p(x)$ *as a corollary. Yet again,* $p(\mathit{fix}\ \lambda x.f\ x)$ *is not a formula of* fohc*, because of the syntax of* $\mathit{fix}\ \lambda x.f\ x$ *is not in FOL.*

Taking the assumption that a theory Γ and a formula F are expressed in *fohc*, we can show that there are four different classes of coinductive proofs for $\Gamma \vdash F$, and they are all characterised by the logic in which the coinductive invariant of the goal-directed derivation of F can be expressed and proven. We take the uniform proofs of Miller, Nadathur et. al [5], and in particular the four uniform proof logics *fohc*, *fohh*, *hohc*, *hohh* (see Figure 1), as a basis for our classification of the expressivity of the coinductive invariants. For example, coinductive invariant of Example 2 belongs to *fohc*, coinductive invariants of Examples 3 and 4 – to *fohh*, and the coinductive invariant of Example 5 – to *fohc* enriched with fixpoint terms. Horn clauses defining irregular streams will require the syntax of *hohh* with fixpoint terms.

Example 6 *Theory* Γ_5 *defines an infinite irregular stream* $[0,(s\ 0),(s\ (s\ 0)),\ldots]$*:*

$$5.1.\ \forall xy \quad \mathit{from}\ (s\ x)\ y \supset \mathit{from}\ x\ (\mathit{scons}(x,y))$$

The infinite derivation for the above stream is given by

$$\mathit{from}\ 0\ y \stackrel{apply 5,[y \mapsto scons(0,y')]}{\rightarrow} \mathit{from}\ (s\ 0)\ y' \rightarrow \cdots$$

It cannot be handled by state-of-the-art coinductive theorem provers such as CoLP [6, 3], as the method of loop detection fails for this example (the subgoals do not unify).

In the next section, we will show that this example, too, can be handled by *coinductive uniform proofs* and falls under the classification of Figure 1. This classification thus provides foundations for automated exploration of coinductive invariants for proofs with coinductive theories expressed in Horn clause logic.

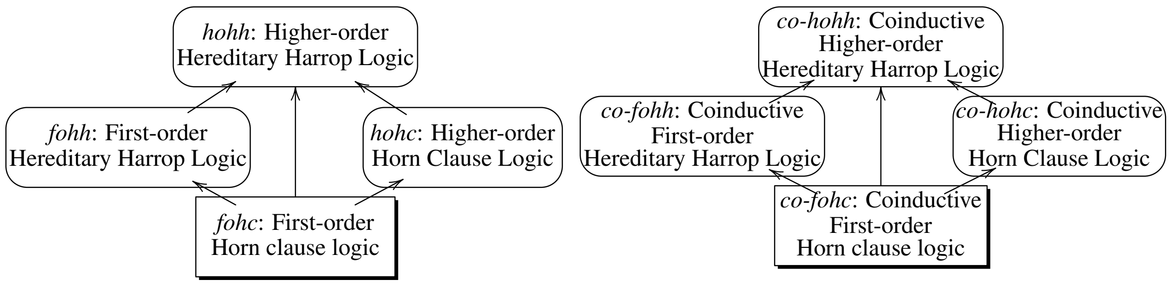

Figure 1: **Left: uniform proof diamond by Miller et al [5]. Right: coinductive uniform proof diamond proposed in this paper.** The arrows show syntactic extensions from first-order to higher-order, from Horn to hereditary Harrop clauses.

2 Meta-theory: Coinductive Uniform Proofs

Our term system extends simply typed lambda terms (typically M,N) by allowing constructs of the form $\mathit{fix}\ \lambda x\,.\,M$, which shall satisfy standard guarding conditions to denote infinite objects. We use $=_{\mathit{fix}\beta}$ for equivalence of two infinite objects (formal details omitted). For example, for a regular fixed point term of Example 5:

$$\mathit{fix}\ \lambda x\,.\,f\ x =_{\mathit{fix}\beta} f(\mathit{fix}\ \lambda x\,.\,f\ x) =_{\mathit{fix}\beta} f(f(\mathit{fix}\ \lambda x\,.\,f\ x)) =_{\mathit{fix}\beta} \cdots$$

The infinite stream $[0, (s\ 0), s(s\ 0), s(s(s\ 0)), \ldots]$ is defined by the higher-order term

$$((\mathit{fix}\ \lambda f n\ .\ \mathit{scons}\ (n, (f\ \ (s\ n))))\ 0)$$

for which we write $\mathit{fr\_str}\ 0$ as a short hand, and which satisfies the following relations

$$\mathit{fr\_str}\ 0 =_{\mathit{fix}\beta} \mathit{scons}(0, (\mathit{fr\_str}\,(s\ 0))) =_{\mathit{fix}\beta}\ \ \mathit{scons}\left(0, \left(\mathit{scons}\left((s\ 0), \left(\mathit{fr\_str}\left(s^2\ 0\right)\right)\right)\right)\right) =_{\mathit{fix}\beta} \cdots$$

The rest of syntax specifications follow the uniform proof theory. We use simple types involving the formula type o, and terms are built using constants from a signature Σ and variables from the countably infinite set *Var*. An atomic formula $B : o$ has the form $(h\,N_1\ \ldots\ N_n)$ where h is either a constant different from $\wedge, \vee, \forall_\tau, \exists_\tau$ and $\supset$, or a variable; B is *rigid* (respectively, *flexible*) if h is a constant (respectively, variable). A term is *closed* if it does not have free variables. We use $\equiv$ for syntactical identity modulo $\alpha-$equivalence, $=_\beta$ for $\beta-$equivalence. We define $\mathscr{U}_1^\Sigma$ as the set of all terms over Σ that do *not* contain $\forall_\tau$ and $\supset$, and $\mathscr{U}_2^\Sigma$ as the set of all terms over Σ that do *not* contain $\supset$. Table 2 defines, for each of the four languages, the set D of *program clauses* and the set G of *goals*. Given a signature Σ, a *program* P is a finite set of *closed* D-formulae over Σ.

We have two kinds of sequents. One kind of sequents are in the form $\Sigma; P \longrightarrow G$, encoding the proposition that the closed goal formula G is provable in intuitionistic logic from the program P on Σ. We use Miller et al's uniform proof rules (with slight extension to support the $=_{\mathit{fix}\beta}$ relation, see Figure 2) to prove sequents of this kind. We are interested in proving the other kind of sequents, which are in the form $\Sigma; P \looparrowright G$, encoding that the closed goal formula G is *coinductively* provable from the program P on Σ.

Proving sequents on $\looparrowright$ is closely related to proving sequents on $\longrightarrow$, and for this point we give both formal and informal explanations. Informally, consider the scenario where we begin with proving

$\Sigma; P \looparrowright G$, which amounts to prove $\Sigma; P, G \longrightarrow G$ next, but the way we can apply inference rules to prove $\Sigma; P, G \longrightarrow G$ is more *restricted*, compared to a related but different scenario in which we begin with proving $\Sigma; P, G \longrightarrow G$. The motivation for such restriction is to ensure consistency, i.e. to avoid erroneously making arbitrary formulae coinductively provable. Formally, we use the CO-FIX rule (Figure 3) for sequents on $\looparrowright$, and we introduce the notation $\langle\rangle$ in the CO-FIX rule, so that a formula marked with $\langle\rangle$ is *guarded*[1] and a sequent with guarded formulae shall be reduced using rules in Figure 4, which encodes the restriction we mentioned in the earlier informal account.

A (coinductive uniform) *proof* is a finite tree such that the root is labeled with $\Sigma; P \looparrowright M$, and leaves are labeled with *initial sequents* which are sequents that can occur as a lower sequent in the rules INITIAL or INITIAL$\langle\rangle$. A proof is constructed in *co-fohc* if all formulae in the proof satisfy the language syntax of *co-fohc*. Proofs constructed in *co-fohh*, *co-hohc*, or *co-hohh* are defined similarly.

	Program Clauses	Goals
co-fohc	$D ::= A^1 \mid G \supset D \mid D \wedge D \mid \forall Var\ D$	$G ::= \top \mid A^1 \mid G \wedge G \mid G \vee G \mid \exists Var\ G$
co-hohc	$D ::= A_r \mid G \supset D \mid D \wedge D \mid \forall Var\ D$	$G ::= \top \mid A \mid G \wedge G \mid G \vee G \mid \exists Var\ G$
co-fohh	$D ::= A^1 \mid G \supset D \mid D \wedge D \mid \forall Var\ D$	$G ::= \top \mid A^1 \mid G \wedge G \mid G \vee G \mid \exists Var\ G \mid D \supset G \mid \forall Var\ G$
co-hohh	$D ::= A_r \mid G \supset D \mid D \wedge D \mid \forall Var\ D$	$G ::= \top \mid A \mid G \wedge G \mid G \vee G \mid \exists Var\ G \mid D \supset G \mid \forall Var\ G$

Table 2: **D- and G-formulae**. A and A_r denote atoms and rigid atoms, respectively. A^1 denote first-order atoms. In the setting of *co-hohc*, A and A_r are from $\mathscr{U}_1^\Sigma$; in the setting of *co-hohh*, A and A_r are from $\mathscr{U}_2^\Sigma$.

$$\frac{\Sigma; P, D \longrightarrow G}{\Sigma; P \longrightarrow D \supset G}\ \supset R \qquad \frac{c : \tau, \Sigma; P \longrightarrow G[x := c]}{\Sigma; P \longrightarrow \forall_\tau x\ G}\ \forall R \qquad \frac{\Sigma; P \longrightarrow G[x := N]}{\Sigma; P \longrightarrow \exists_\tau x\ G}\ \exists R$$

$$\frac{\Sigma; P \longrightarrow G_1}{\Sigma; P \longrightarrow G_1 \vee G_2}\ \vee R \qquad \frac{\Sigma; P \longrightarrow G_2}{\Sigma; P \longrightarrow G_1 \vee G_2}\ \vee R \qquad \frac{\Sigma; P \longrightarrow G_1 \quad \Sigma; P \longrightarrow G_2}{\Sigma; P \longrightarrow G_1 \wedge G_2}\ \wedge R$$

$$\frac{\Sigma; P \xrightarrow{D} A \quad \Sigma; P \longrightarrow G}{\Sigma; P \xrightarrow{G \supset D} A}\ \supset L \qquad \frac{\Sigma; P \xrightarrow{D_1} A}{\Sigma; P \xrightarrow{D_1 \wedge D_2} A}\ \wedge L \qquad \frac{\Sigma; P \xrightarrow{D_2} A}{\Sigma; P \xrightarrow{D_1 \wedge D_2} A}\ \wedge L \qquad \frac{\Sigma; P \xrightarrow{D[x:=N]} A}{\Sigma; P \xrightarrow{\forall_\tau x\ D} A}\ \forall L$$

$$\frac{\Sigma; P \xrightarrow{D} A}{\Sigma; P \longrightarrow A}\ \text{DECIDE} \qquad \frac{}{\Sigma; P \xrightarrow{A'} A}\ \text{INITIAL} \qquad \frac{}{\Sigma; P \longrightarrow \top}\ \top R$$

Figure 2: **Uniform proof rules**. *Rule restrictions:* in $\exists R$ and $\forall L$, $N : \tau$ is a closed term on Σ. Moreover, if used in *co-fohc* or *co-fohh*, then N is first order; if used in *co-hohc*, then $N \in \mathscr{U}_1^\Sigma$; if used in *co-hohh*, then $N \in \mathscr{U}_2^\Sigma$. In $\forall R$, $c : \tau \notin \Sigma$ (c is also known as an *eigenvariable*). In DECIDE, $D \in P$. In the rule INITIAL, $A =_{fix\beta} A'$.

3 Discussion

Using coinductive uniform proofs, we can categorize infinite SLD-derivations, and we can uniformly and proof-theoretically formalize the coinductive reasoning performed by the two algorithms mentioned

[1] There are two distinct notions of *guard* in coinductive uniform proof: one is for the syntax of fixed-point terms, to ensure that they model infinite objects; the other is for formulae in certain sequents, to ensure consistency.

$$\frac{\Sigma;P,\langle M\rangle \longrightarrow \langle M\rangle}{\Sigma;P \looparrowright M}\ \text{CO-FIX} \qquad \begin{array}{llll} \textit{co-fohc} & M := A^1 \mid M \wedge M & \textit{co-fohh} & M := A^1 \mid M \wedge M \mid M \supset M \mid \forall Var\ M \\ \textit{co-hohc} & M := A_r \mid M \wedge M & \textit{co-hohh} & M := A_r \mid M \wedge M \mid M \supset M \mid \forall Var\ M \end{array}$$

Figure 3: **The coinductive fixed-point rule and syntax for core formulae.** *Note:* In the upper sequent of CO-FIX rule, the left occurrence of M is called a *coinductive invariant*, and the right occurrence of M is called a *coinductive goal*. The formula M occurs on *both* sides of the upper sequent in the CO-FIX rule, therefore M must satisfy the syntax of both program clauses and goals. Formulae with such syntactic character as M are called *core formulae* [5].

$$\frac{\Sigma;P,\langle M_1\rangle \longrightarrow \langle M_2\rangle}{\Sigma;P \longrightarrow \langle M_1 \supset M_2\rangle}\ {\supset}R\langle\rangle \qquad \frac{c:\tau,\Sigma;P \longrightarrow \langle M\,[x:=c]\rangle}{\Sigma;P \longrightarrow \langle \forall_\tau x\, M\rangle}\ \forall R\langle\rangle \qquad \frac{\Sigma;P \longrightarrow \langle M_1\rangle \quad \Sigma;P \longrightarrow \langle M_2\rangle}{\Sigma;P \longrightarrow \langle M_1 \wedge M_2\rangle}\ \wedge R\langle\rangle$$

$$\frac{\Sigma;P^* \xrightarrow{D} A \quad \Sigma;P^* \longrightarrow G}{\Sigma;P \xrightarrow{G\supset D} \langle A\rangle}\ {\supset}L\langle\rangle \qquad \frac{\Sigma;P \xrightarrow{D_1} \langle A\rangle}{\Sigma;P \xrightarrow{D_1\wedge D_2} \langle A\rangle}\ \wedge L\langle\rangle \qquad \frac{\Sigma;P \xrightarrow{D_2} \langle A\rangle}{\Sigma;P \xrightarrow{D_1\wedge D_2} \langle A\rangle}\ \wedge L\langle\rangle \qquad \frac{\Sigma;P \xrightarrow{D[x:=N]} \langle A\rangle}{\Sigma;P \xrightarrow{\forall x\, D} \langle A\rangle}\ \forall L\langle\rangle$$

$$\frac{\Sigma;P \xrightarrow{D^*} \langle A\rangle}{\Sigma;P \longrightarrow \langle A\rangle}\ \text{DECIDE}\langle\rangle \qquad \frac{}{\Sigma;P \xrightarrow{A'} \langle A\rangle}\ \text{INITIAL}\langle\rangle$$

Figure 4: **Rules for guarded coinductive goals.** *Rule restrictions*: In DECIDE$\langle\rangle$, D^* must be a formula without $\langle\rangle$ mark. In ${\supset}L\langle\rangle$, P^* results from erasing all $\langle\rangle$ marks in P. The restrictions for INITIAL$\langle\rangle$, $\forall L\langle\rangle$ and $\forall R\langle\rangle$ are the same as for INITIAL, $\forall L$ and $\forall R$ respectively. *Note:* Formulae added to the left-hand side by CO-FIX and ${\supset}R\langle\rangle$ are guarded, so that they are not selected by the DECIDE$\langle\rangle$ rule for back-chaining with guarded atomic goals. The ${\supset}L\langle\rangle$ rule frees all formulae from being guarded for each upper sequent, then rules in Figure 2 become applicable in further sequent reductions.

earlier. For instance, to handle Example 6, we need *co-hohh* extended with fixed point terms to express and prove the coinductive invariant $\forall x\, \mathit{from}\ x\ (\mathit{fr\_str}\ x)$, with the root sequent $\Sigma_5;\Gamma_5 \looparrowright \forall x\, \mathit{from}\ x\ (\mathit{fr\_str}\ x)$.

We give the *co-hohh* proof[2] for the sequent $\Sigma_5;\Gamma_5 \looparrowright \forall x(\mathit{from}\ x\ (\mathit{fr\_str}\ x))$. Note that fr_str is defined in Section 2, CH abbreviates the coinductive hypothesis $\forall x(\mathit{from}\ x\ (\mathit{fr\_str}\ x))$, Z is an arbitrary eigenvariable, and the step marked by ✓ indicates involvement of the relation

$$\mathit{from}\ Z\ (scons(Z,(fr\_str\ (s\ Z)))) =_{\mathit{fix}\beta} \mathit{from}\ Z\ (\mathit{fr\_str}\ Z)$$

The two $\forall L\langle\rangle$ steps involve the substitutions $x := Z, y := (fr\_str\ (s\ Z))$. The $\forall L$ step involves the substitution $x := s\ Z$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{Z,\Sigma;\Gamma,CH \xrightarrow{\mathit{from}\ Z\ (scons\ Z\ (fr\_str\ (s\ Z)))} \mathit{from}\ Z\ (\mathit{fr\_str}\ Z)}\ \text{INITIAL}✓ \qquad \dfrac{\dfrac{\dfrac{}{Z,\Sigma;\Gamma,CH \xrightarrow{\mathit{from}\ (s\ Z)\ (\mathit{fr\_str}\ (s\ Z))} \mathit{from}\ (s\ Z)\ (\mathit{fr\_str}\ (s\ Z))}\ \text{INITIAL}}{Z,\Sigma;\Gamma,CH \xrightarrow{CH} \mathit{from}\ (s\ Z)\ (\mathit{fr\_str}\ (s\ Z))}\ \forall L}{Z,\Sigma;\Gamma,CH \longrightarrow \mathit{from}\ (s\ Z)\ (\mathit{fr\_str}\ (s\ Z))}\ \text{DECIDE}}{Z,\Sigma;\Gamma,\langle CH\rangle \xrightarrow{\mathit{from}\ (s\ Z)(\mathit{fr\_str}\ (s\ Z))\ \supset \mathit{from}\ Z\ (scons\ Z\ (\mathit{fr\_str}\ (s\ Z)))} \langle \mathit{from}\ Z\ (\mathit{fr\_str}\ Z)\rangle}\ {\supset}L\langle\rangle}{Z,\Sigma;\Gamma,\langle CH\rangle \xrightarrow{\forall xy\ \mathit{from}\ (s\ x)\ y \supset \mathit{from}\ x\ (scons\ x\ y)} \langle \mathit{from}\ Z\ (\mathit{fr\_str}\ Z)\rangle}\ \forall L\langle\rangle\ (2\text{ times})}{\dfrac{Z,\Sigma;\Gamma,\langle CH\rangle \longrightarrow \langle \mathit{from}\ Z\ (\mathit{fr\_str}\ Z)\rangle}{\dfrac{\Sigma;\Gamma,\langle CH\rangle \longrightarrow \langle \forall x(\mathit{from}\ x\ (\mathit{fr\_str}\ x))\rangle}{\Sigma;\Gamma \looparrowright \forall x(\mathit{from}\ x\ (\mathit{fr\_str}\ x))}\ \text{CO-FIX}}\ \forall R\langle\rangle}\ \text{DECIDE}\langle\rangle$$

[2]We omit the subscript 5 for Σ,Γ in the proof.

Given this proof, we can obtain the proof for *from 0* (*fr_str 0*) as a corollary. This is exactly the goal we were not able to acheive in Example 6 by loop detection.

The fact that the CO-FIX rule can only be applied once and as the first step in a proof, is a simplification that helps to highlight the basic coinductive argument performed by the coinductive uniform proofs. The absence of nested coinduction in the meta-theory can be mitigated by allowing using the already proven coinductive invariants as lemmas to prove further coinductive conclusions.

4 Future Work

We omit technical details of the proof of soundness of coinductive uniform proofs w.r.t greatest complete Herbrand models. Intuitively, the proof proceeds by defining a scheme by which we can reconstruct a corresponding non-terminating derivation, and then showing that the proofs are sound w.r.t greatest complete Herbrand models. However, in contrast with CoLP, the reconstruction is generally more complicated and involves

- a construction of a function that generates countably many different substitution instances for the derivation scheme, and
- showing that these instances can be composed in a certain way in order to restore the full infinite derivation.

The proof is constructive, and in addition uses a coinductive proof principle when showing correspondence of the derivation schemes to greatest complete Herbrand model construction.

Now that we have a sound framework for automated coinductive proof construction, the practical problem is to formulate heuristics that can find suitable coinductive invariants to prove. It can be shown that CoLP method in fact finds coinductive invariants expressed and proven in *co-fohc* (with and without fixed point terms). The method presented in [1] formulates coinductive invariants in *co-fohh* (without fixed point terms). The current work is on the way to generalise these methods to other logics.

References

[1] Peng Fu, Ekaterina Komendantskaya, Tom Schrijvers & Andrew Pond (2016): *Proof Relevant Corecursive Resolution*. In: *Functional and Logic Programming - 13th International Symposium, FLOPS 2016, Kochi, Japan, March 4-6, 2016, Proceedings*, *Lecture Notes in Computer Science* 9613, Springer, pp. 126–143, doi:10.1007/978-3-319-29604-3_9.

[2] Gopal Gupta, Ajay Bansal, Richard Min, Luke Simon & Ajay Mallya (2007): *Coinductive Logic Programming and Its Applications*. In Véronica Dahl & Ilkka Niemelä, editors: *Logic Programming*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 27–44, doi:10.1007/978-3-540-74610-2_4.

[3] Ekaterina Komendantskaya & Yue Li (2017): *Productive corecursion in logic programming*. *TPLP* 17(5-6), pp. 906–923, doi:10.1017/S147106841700028X.

[4] John W. Lloyd (1987): *Foundations of Logic Programming, 2nd Edition*. Springer, doi:10.1007/978-3-642-83189-8.

[5] Dale Miller & Gopalan Nadathur (2012): *Programming with Higher-Order Logic*. Cambridge University Press, doi:10.1017/CBO9781139021326.

[6] Luke Simon, Ajay Mallya, Ajay Bansal & Gopal Gupta (2006): *Coinductive Logic Programming*. In: *Logic Programming, 22nd International Conference, ICLP 2006, Seattle, WA, USA, August 17-20, 2006, Proceedings*, *Lecture Notes in Computer Science* 4079, Springer, pp. 330–345, doi:10.1007/11799573_25.