HoCHC: A Refutationally Complete and Semantically Invariant System of Higher-order Logic Modulo Theories

C.-H. Luke Ong
University of Oxford

Dominik Wagner
University of Oxford

Abstract—**We present a simple resolution proof system for *higher-order constrained Horn clauses* (HoCHC)—a system of higher-order logic modulo theories—and prove its soundness and refutational completeness w.r.t. both standard and Henkin semantics. As corollaries, we obtain the compactness theorem and semi-decidability of HoCHC for semi-decidable background theories, and we prove that HoCHC satisfies a canonical model property. Moreover a variant of the well-known translation from higher-order to 1st-order logic is shown to be sound and complete for HoCHC in both semantics. We illustrate how to transfer decidability results for (fragments of) 1st-order logic modulo theories to our higher-order setting, using as example the Bernays-Schönfinkel-Ramsey fragment of HoCHC modulo a restricted form of Linear Integer Arithmetic.**

I. Introduction

Cathcart Burn et al. [1] recently advocated an automatic, programming-language independent approach to verify safety properties of higher-order programs by framing them as solvability problems for systems of higher-order constraints. These systems consist of Horn clauses of higher-order logic, containing constraints expressed in some suitable background theory. Consider the functional program:

```
let $add\,x\,y = x + y$
letrec $iter\,f\,s\,n =$ if $n \le 0$ then $s$ else $f\,n\,(iter\,f\,s\,(n-1))$
    in $\lambda n.$assert $\big(n \ge 1 \;\to\; (iter\,add\,n\,n > n + n)\big)$
```

Thus $(iter\,add\,n\,n)$ computes the value $n + \sum_{i=1}^{n} i$.

To verify that the program is *safe* (i.e. the assertion is never violated), it suffices to find overapproximations of the input-output-graph (i.e. *invariants*) of the functions that imply the required property. The idea then is to express the problem of finding such a program invariant, *logically*, as a satisfiability problem for the following higher-order constrained system:

Example 1 (Invariant as system of higher-order constraints)**.**

$$\begin{aligned}
&\forall x, y, z \,.\, \big(z = x + y \to \mathrm{Add}\ x\,y\,z\big)\\
&\forall f, s, n, x \,.\, \big(n \le 0 \land s = x \to \mathrm{Iter}\ f\,s\,n\,x\big)\\
&\forall f, s, n, x \,.\, \big(n > 0 \land \exists y \,.\, (\mathrm{Iter}\ f\,s\,(n-1)\,y \land f\,n\,y\,x)\\
&\qquad\qquad \to \mathrm{Iter}\ f\,s\,n\,x\big)\\
&\forall n, x \,.\, \big(n \ge 1 \land \mathrm{Iter}\ \mathrm{Add}\ n\,n\,x \;\to\; x > n + n\big)
\end{aligned}$$

The above are Horn clauses of higher-order logic, obtained by transformation from the preceding program; $\mathrm{Add} : \iota \to \iota \to \iota \to o$ and $\mathrm{Iter} : (\iota \to \iota \to \iota \to o) \to \iota \to \iota \to \iota \to o$ are higher-order relations, and the binary predicates $(\le, >, \cdots)$ are formulas of the background theory, Linear Integer Arithmetic (LIA).

Since the the assertion in the program is violated for $n = 1$, the clauses are unsatisfiable.

Is higher-order logic modulo theories a sensible algorithmic approach to verification? Is it well-founded?

To set the scene, recall that 1st-order logic is semi-decidable: 1st-order validities[1] are recursively enumerable; moreover if a formula is unsatisfiable then it is provable by resolution [2], [3]. By contrast, higher-order logic in standard semantics is wildly undecidable. E.g. the set[1] $\mathbf{V}^2(=)$ of valid sentences of the 2nd-order language of equality is not even analytical [4].

This does not necessarily spell doom for the higher-order logic approach. One could consider higher-order logic in *Henkin semantics* [5], which is, after all, "nothing but many-sorted 1st-order logic with comprehension axioms" [4] (see also [6], [7]). However, because the standard semantics is natural and comparatively simple, it seems to be the semantics of choice in program verification (e.g. monadic 2nd-order logic in model checking, and HOL theorem prover [8], [9] in automated deduction) and in program specification.

In this paper, we study the algorithmic, model-theoretic and semantical properties of higher-order Horn clauses with a 1st-order background theory.

a) A Complete Resolution Proof System for HoCHC: The main technical contribution of this paper is the design of a simple resolution proof system for *higher-order constrained Horn clauses* (HoCHC) where the background theory has a unique model [1], and its refutational completeness proof with respect to the standard semantics (Sec. IV). The proof system and its refutational completeness proof are generalised in Sec. VI to arbitrary *compact* background theories, which may have more than one model.

The completeness proof hinges on a novel model-theoretic insight: we prove that the immediate consequence operator is

[1] Define $\mathbf{V}^n(P)$ to be the set of valid sentences of nth-order logic with 2-place predicate P. Then $\mathbf{V}^1(=)$ is recursively enumerable

978-1-7281-3608-0/19/$31.00 ©2019 IEEE

quasi-continuous, although it is not continuous in the standard Scott sense. Thus, the immediate consequence operator gives rise to a syntactic explanation for unsatisfiability. Moreover, we adapt the proof of the standardisation theorem of the λ-calculus in [10] to argue that this explanation can be captured by the rules of the resolution proof system.

b) Canonical Model Property: As shown in [1], a disadvantage of the standard semantics is failure of the least model property (w.r.t. the pointwise ordering). However, we prove in Sec. III that the immediate consequence operator is "sufficiently" monotone and hence (by an extension of the Knaster-Tarski theorem) gives rise to a model of all satisfiable instances.

c) Compactness Theorem and Semi-decidability of HoCHC: A well-known feature of higher-order logic in standard semantics is failure of the compactness theorem. As a consequence of HoCHC's refutational completeness, it follows that the compactness theorem *does* hold for HoCHC (in standard semantics): for every unsatisfiable set Γ of HoCHCs, there is a finite subset $\Gamma' \subseteq \Gamma$ which is unsatisfiable.

Moreover, if the consistency of conjunctions of atoms in the background theory is semi-decidable, so is HoCHC unsatisfiability. Crucially, this underpins the *practicality* of the HoCHC-based approach to program verification.

d) Semantic Invariance: The soundness and completeness of our resolution proof system has another pleasing corollary: satisfiability of HoCHC does *not* depend on the choice of semantics[2] (Sec. V). In particular, this constitutes an alternative proof of the equivalence of standard, monotone and continuous semantics for HoCHCs, without exhibiting explicit translations between semantics. Moreover, this demonstrates that, in contrast to (full) higher-order logic, satisfiability of HoCHCs with respect to standard semantics on the one hand, and to Henkin semantics on the other, coincide.

Semantic invariance is an important advantage for program verification. It follows that one can use (the simpler and more intuitive) standard semantics for specification, but use Henkin semantics for the development of refined proof systems that are complete [11-14], and use continuous semantics (which enjoys a richer structure) to construct solution methods and in static analysis.

e) Complete 1st-order Translation: As suggested by the equivalence of standard and Henkin semantics, we show that there is a variant of the standard translation of higher-order logic into 1st-order logic which is sound and complete also for standard semantics, when restricted to HoCHC (Sec. VII).

f) Decidable Fragments of HoCHC: Satisfiability of finite sets of HoCHCs is trivially decidable for background theories with finite domains. In Sec. VIII, we identify a fragmentof HoCHC (the Bernays-Schönfinkel-Ramsey fragment of HoCHC modulo a restricted form of Linear Integer Arithmetic) with a decidable satisfiability problem by showing equi-satisfiability to clauses w.r.t. a finite number of such background theories.

Outline: We begin with some key definitions in Sec. II. Then we show that even standard semantics satisfies a canonical model property (Sec. III). In Sec. IV, we present the resolution proof system for HoCHC and prove its completeness. In Sec. V we show HoCHC's semantic invariance and in Sec. VI we generalise the refutational completeness proof to compact background theories, which may have more than one model. In Sec. VII we present a 1st-order translation of higher-order logic and prove it complete when restricted to HoCHC. In Sec. VIII we exhibit decidable fragments of HoCHC. Finally, we discuss related work in Sec. IX, and conclude in Sec. X.

For the extended version of the paper refer to [15].

II. Technical Preliminaries

This section introduces the syntax and semantics of a restricted form of higher-order logic (Sec. II-A), higher-order constrained Horn clauses (Sec. II-B) and programs (Sec. II-C).

A. Relational Higher-order Logic

1) Syntax: For a fixed set $\mathfrak{I}$ (intuitively the types of individuals), the set of *argument types*, *relational types*, *1st-order types* and *types* (generated by $\mathfrak{I}$) are mutual recursively defined by

$$\begin{array}{llll} \textit{Argument type} & \tau & ::= & \iota \mid \rho \\ \textit{Relational type} & \rho & ::= & o \mid \tau \to \rho \\ \textit{1st-order type} & \sigma_{\mathrm{FO}} & ::= & \iota \mid \iota \to o \mid \iota \to \sigma_{\mathrm{FO}} \\ \textit{Type} & \sigma & ::= & \rho \mid \sigma_{\mathrm{FO}}, \end{array}$$

where $\iota \in \mathfrak{I}$. We sometimes abbreviate the (1st-order) type $\underbrace{\iota \to \cdots \to \iota}_{n} \to \iota$ to $\iota^n \to \iota$ (similarly for $\iota^n \to o$). For types $\tau_1 \to \cdots \to \tau_n \to \sigma$ we also write $\overline{\tau} \to \sigma$. Intuitively, o is the type of the truth values (or Booleans). Besides, σ_{FO} contains all (1st-order) types of the form $\iota^n \to \iota$ or $\iota^n \to o$, i.e. all arguments are of type ι. Moreover, each relational type has the form $\overline{\tau} \to o$.

A *type environment* (typically Δ) is a function mapping variables (typically denoted by x, y, z etc.) to argument types; for $x \in \mathrm{dom}(\Delta)$, we write $x : \tau \in \Delta$ to mean $\Delta(x) = \tau$. A *signature* is a set of distinct typed *symbols* $c : \sigma$, where $c \notin \mathrm{dom}(\Delta)$ and c is not one of the *logical symbols* $\neg$, $\wedge$, $\vee$ and $\exists_\tau$ (for argument types τ, which we omit frequently). It is *1st-order* if for each $c : \sigma \in \Sigma$, σ is 1st-order. We often write $c \in \Sigma$ if $c : \sigma \in \Sigma$ for some σ.

The set of Σ-*pre-terms* is given by

$$M ::= x \mid c \mid \neg \mid \wedge \mid \vee \mid \exists_\tau \mid MM \mid \lambda x.\, M$$

where $c \in \Sigma$. Following the usual conventions we assume that application associates to the left and the scope of abstractions extend as far to the right as possible. We also write $M\,\overline{N}$ and $\lambda\overline{x}.\, M'$ for $M\,N_1 \cdots N_n$ and $\lambda x_1. \cdots \lambda x_n.\, M'$, respectively, assuming implicitly that M is not an application. Besides,

[2] within the reasonable bounds formalised by *(complete) frames*

$$\frac{x \in \mathrm{dom}(\Delta)}{\Delta \vdash x : \Delta(x)}\ (\mathrm{Var}) \qquad \frac{c : \sigma \in \Sigma}{\Delta \vdash c : \sigma}\ (\mathrm{Cst}) \qquad \frac{\Delta \vdash M_1 : \sigma_1 \to \sigma_2 \quad \Delta \vdash M_2 : \sigma_1}{\Delta \vdash M_1 M_2 : \sigma_2}\ (\mathrm{App}) \qquad \frac{\Delta \vdash M : \rho}{\Delta \vdash \lambda x.\, M : \Delta(x) \to \rho}\ (\mathrm{Abs})$$

$$\frac{\circ \in \{\wedge, \vee\}}{\Delta \vdash \circ : o \to o \to o}\ (\mathrm{And/Or}) \qquad \frac{\Delta \vdash M : o}{\Delta \vdash \neg M : o}\ (\mathrm{Neg}) \qquad \frac{}{\Delta \vdash \exists_\tau : (\tau \to o) \to o}\ (\mathrm{Ex})$$

Figure 1. Typing judgements

we abbreviate $\exists_\tau(\lambda x.\, M)$ as $\exists x.\, M$. Moreover, we identify terms up to α-equivalence and adopt Barendregt's *variable convention* [16].

The typing judgement $\Delta \vdash M : \sigma$ is defined in Fig. 1. We say that M is Σ*-term* if $\Delta \vdash M : \sigma$ for some σ and it is a Σ*-formula* if $\sigma = o$. A Σ-formula is a *1st-order* Σ*-formula* if its construction is restricted to symbols $c : \sigma_{\mathrm{FO}} \in \Sigma$ and variables $x : \iota \in \Delta$, and uses no λ-abstraction. Finally, for a Σ-term M, $\mathrm{fv}(M)$ is the set of free variables, and M is a *closed* Σ-term if $\mathrm{fv}(M) = \emptyset$.

Remark 2. It follows from the definitions that (i) each term $\Delta \vdash M : \iota^n \to \iota$ can only contain variables of type ι and constants of non-relational 1st-order type, and contains neither λ-abstractions nor logical symbols (a similar approach is adopted in [17]); (ii) $\neg$ can only occur in a term if applied to a formula (and not in pre-terms of the form $R\,\neg$).

The following kind of terms is particularly significant:

Definition 3. A Σ-term is *positive existential* if the logical constant "$\neg$" is not a subterm.

For Σ-terms $M, N_1, \ldots, N_n$ and variables $x_1, \ldots, x_n$ satisfying $\Delta \vdash N_i : \Delta(x_i)$, the *(simultaneous) substitution* $M[N_1/x_1, \ldots, N_n/x_n]$ is defined in the standard way.

2) Semantics: There are two classic semantics for higher-order logic: *standard* and *Henkin semantics* [5]. Whereas in standard semantics the interpretation of higher types is uniquely determined by the domains of individuals (quantifiers range over *all* set-theoretic functions of the appropriate type), it can be *stipulated* quite liberally in Henkin semantics.

Formally, a *pre-frame* $\mathcal{F}$ assigns to each type σ a non-empty set $\mathcal{F}[\![\sigma]\!]$ such that

(i) $\mathcal{F}[\![o]\!] := \mathbb{B} := \{0, 1\}$ and for each type $\sigma_1 \to \sigma_2$, $\mathcal{F}[\![\sigma_1 \to \sigma_2]\!] \subseteq [\mathcal{F}[\![\sigma_1]\!] \to \mathcal{F}[\![\sigma_2]\!]]$
(ii) $\mathrm{and}, \mathrm{or} \in \mathcal{F}[\![o \to o \to o]\!]$
(iii) $\exists_\tau \in \mathcal{F}[\![(\tau \to o) \to o]\!]$ for each argument type τ

where $[\mathcal{F}[\![\sigma_1]\!] \to \mathcal{F}[\![\sigma_2]\!]]$ is the set of functions $\mathcal{F}[\![\sigma_1]\!] \to \mathcal{F}[\![\sigma_2]\!]$ and

$$\mathrm{and}(b_1)(b_2) := \min\{b_1, b_2\} \qquad \mathrm{or}(b_1)(b_2) := \max\{b_1, b_2\}$$
$$\mathrm{exists}_\tau(r) := \max\{r(s) \mid s \in \mathcal{F}[\![\tau]\!]\}$$

Example 4 (Pre-frames)**.** For every $\iota \in \mathfrak{I}$, we fix an arbitrary non-empty set D_ι. We define $\mathcal{S}$, $\mathcal{M}$ and $\mathcal{C}$, which we call the *standard*, *monotone* and *continuous frame*, respectively, recursively by $\mathcal{S}[\![o]\!] := \mathcal{M}[\![o]\!] := \mathcal{C}[\![o]\!] := \mathbb{B}$; $\mathcal{S}[\![\iota]\!] := \mathcal{M}[\![\iota]\!] := \mathcal{C}[\![\iota]\!] := D_\iota$ for $\iota \in \mathfrak{I}$; and

$$\mathcal{S}[\![\tau \to \sigma]\!] := [\mathcal{S}[\![\tau]\!] \to \mathcal{S}[\![\sigma]\!]]$$
$$\mathcal{M}[\![\tau \to \sigma]\!] := [\mathcal{M}[\![\tau]\!] \xrightarrow{m} \mathcal{M}[\![\sigma]\!]]$$
$$\mathcal{C}[\![\tau \to \sigma]\!] := [\mathcal{C}[\![\tau]\!] \xrightarrow{c} \mathcal{C}[\![\sigma]\!]],$$

where $[P \xrightarrow{m} P']$ ($[P \xrightarrow{c} P']$) is the set of monotone (continuous) functions from the posets P to P' (cf. [18]).

Let Σ be a signature and $\mathcal{F}$ be a pre-frame. A $(\Sigma, \mathcal{F})$-*structure* $\mathcal{A}$ assigns to each $c : \sigma \in \Sigma$ an element $c^{\mathcal{A}} \in \mathcal{F}[\![\sigma]\!]$ and we set $\mathcal{A}[\![\sigma]\!] := \mathcal{F}[\![\sigma]\!]$ for types σ. A $(\Delta, \mathcal{F})$-*valuation* α is a function such that for every $x : \tau \in \Delta$, $\alpha(x) \in \mathcal{F}[\![\tau]\!]$. For a $(\Delta, \mathcal{F})$-valuation α, variable x and $r \in \mathcal{F}[\![\Delta(x)]\!]$, $\alpha[x \mapsto r]$ is defined in the usual way.

The *denotation* $\mathcal{A}[\![M]\!](\alpha)$ of a term M with respect to $\mathcal{A}$ and α is defined recursively by

$$\mathcal{A}[\![x]\!](\alpha) := \alpha(x) \qquad \mathcal{A}[\![c]\!](\alpha) := c^{\mathcal{A}}$$
$$\mathcal{A}[\![\wedge]\!](\alpha) := \mathrm{and} \qquad \mathcal{A}[\![\vee]\!](\alpha) := \mathrm{or}$$
$$\mathcal{A}[\![\exists_\tau]\!](\alpha) := \mathrm{exists}_\tau \qquad \mathcal{A}[\![\neg M]\!](\alpha) := 1 - \mathcal{A}[\![M]\!](\alpha)$$
$$\mathcal{A}[\![M_1\, M_2]\!](\alpha) := \mathcal{A}[\![M_1]\!](\alpha)(\mathcal{A}[\![M_2]\!](\alpha))$$
$$\mathcal{A}[\![\lambda x.\, M]\!](\alpha) := [\lambda r \in \mathcal{A}[\![\Delta(x)]\!].\, \mathcal{A}[\![M]\!](\alpha[x \mapsto r])]_{\Delta(x) \to \rho}$$

(assuming $\Delta \vdash M : \rho$ in the last case), where $[r]_\sigma = r$ if $r \in \mathcal{A}[\![\sigma]\!]$ and otherwise $[r]_\sigma \in \mathcal{A}[\![\sigma]\!]$ is arbitrary. Thus, for each term $\Delta \vdash M : \sigma$, $\mathcal{A}[\![M]\!](\alpha) \in \mathcal{A}[\![\sigma]\!]$.

Being independent of valuations, the denotation of closed terms M is abbreviated as $\mathcal{A}[\![M]\!]$. Besides, for Σ-formulas F, we write $\mathcal{A}, \alpha \models F$ if $\mathcal{A}[\![F]\!](\alpha) = 1$, and $\mathcal{A} \models F$ if for all α', $\mathcal{A}, \alpha' \models F$. We extend $\models$ in the usual way to sets of formulas.

A *frame* is a pre-frame $\mathcal{F}$ that satisfies the

> *Comprehension Axiom*: for each signature Σ, type environment Δ, $(\Sigma, \mathcal{F})$-structure $\mathcal{A}$, $(\Delta, \mathcal{F})$-valuation α, positive existential Σ-term $\lambda x.\, M$, and $r \in \mathcal{A}[\![\Delta(x)]\!]$, $\mathcal{A}[\![\lambda x.\, M]\!](\alpha)(r) = \mathcal{A}[\![M]\!](\alpha[x \mapsto r])$.

Our comprehension axiom ensures that positive existential terms are interpreted in the expected way; it is non-standard in that it is restricted to positive existential formulas.

As a consequence, if $\mathcal{F}$ is a frame then for every relational type $\overline{\tau} \to o$, $\top_{\overline{\tau} \to o} \in \mathcal{F}[\![\overline{\tau} \to o]\!]$, where $1 =: \top_{\overline{\tau} \to o}(\overline{r}) = \mathcal{A}[\![\lambda \overline{x}.\, y]\!](\alpha[y \mapsto 1])(\overline{r})$.

a) Complete Frames: For types σ, let $\sqsubseteq_\sigma \subseteq \mathcal{F}[\![\sigma]\!] \times \mathcal{F}[\![\sigma]\!]$ be the usual partial order defined pointwise for higher types, which is the discrete order on $\mathcal{F}[\![\iota]\!]$ and the “less than or equal” relation on $\mathcal{F}[\![o]\!]$.

For relational types ρ and $\mathfrak{R} \subseteq \mathcal{F}[\![\rho]\!]$, the least upper bound $\bigsqcup_{\rho} \mathfrak{R}$ is defined pointwise, by recursion on ρ. In particular, $\bigsqcup_{\overline{\tau}\to o} \emptyset = \bot_{\overline{\tau}\to o}$, where $\bot_{\overline{\tau}\to o}(\overline{r}) := 0$. For a singleton set $\{f\} \subseteq \mathcal{F}[\![\iota^n \to \iota]\!]$ we define $\bigsqcup_{\iota^n\to\iota}\{f\} := f$. Throughout the paper, we omit type subscripts to reduce clutter because they can be inferred.

A (pre-)frame $\mathcal{F}$ is *complete* if for every relational ρ and $\mathfrak{R} \subseteq \mathcal{F}[\![\rho]\!]$, $\bigsqcup \mathfrak{R} \in \mathcal{F}[\![\rho]\!]$, i.e. each $\mathcal{F}[\![\rho]\!]$ is a complete lattice ordered by $\sqsubseteq_{\rho}$ with least upper bounds $\bigsqcup_{\rho}$.

Example 5 (complete frames)**.** $\mathcal{S}$ is trivially a complete frame. It is not difficult to prove that $\mathcal{M}$ and $\mathcal{C}$ are also complete frames [15].

b) 1st-order Structures: Let Σ be a 1st-order signature. A *1st-order* Σ*-structure* is a $(\Sigma, \mathcal{S})$-structure. Note that by taking standard frames this coincides with the standard definition in a purely 1st-order setting (cf. e.g. [19]).

Example 6. In the examples we will primarily be concerned with the signature of *Linear Integer Arithmetic*[3] $\Sigma_{\mathrm{LIA}} := \{0, 1, +, -, <, \leq, =, \neq, \geq, >\}$ and its standard model $\mathcal{A}_{\mathrm{LIA}}$.

B. Higher-order Constrained Horn Clauses

Assumption. *Henceforth, we fix a 1st-order signature Σ over a single type of individuals ι and a 1st-order Σ-structure $\mathcal{A}$.*

Moreover, we fix a signature Σ' extending Σ with (only) symbols of relational type, and a type environment Δ such that $\Delta^{-1}(\tau)$ is infinite for each argument type τ.

Intuitively, Σ and $\mathcal{A}$ correspond to the language and interpretation of the background theory, e.g. Σ_{LIA} together with its standard model $\mathcal{A}_{\mathrm{LIA}}$. In particular, we first focus on background theories with a single model. In Sec. VI we extend our results to a more general setting.

We are interested in whether 1st-order structures can be expanded to larger (higher-order) signatures. This is made precise by the following:

Definition 7. (i) A frame $\mathcal{F}$ *expands* $\mathcal{A}$ if $\mathcal{F}[\![\iota]\!] = \mathcal{A}[\![\iota]\!]$ and $c^{\mathcal{A}} \in \mathcal{F}[\![\sigma]\!]$ for all $c : \sigma \in \Sigma$.
(ii) Suppose $\mathcal{F}$ expands $\mathcal{A}$. Then a $(\Sigma', \mathcal{F})$-structure $\mathcal{B}$ is a $(\Sigma', \mathcal{F})$*-expansion of* $\mathcal{A}$ if $c^{\mathcal{A}} = c^{\mathcal{B}}$ for all $c \in \Sigma$.

Remark 8. (i) By Remark 2 the denotation of terms $\Delta \vdash M : \iota^n \to \iota$ is the same for all $(\Sigma', \mathcal{F})$-expansions of $\mathcal{A}$ and $(\Delta, \mathcal{F})$-valuations agreeing on $\Delta^{-1}(\tau)$.
(ii) In case $\mathcal{F}$ is complete, the $(\Sigma', \mathcal{F})$-expansions of $\mathcal{A}$ ordered by $\sqsubseteq$ constitute a complete lattice with least upper bounds $\bigsqcup$, where $\sqsubseteq$ and $\bigsqcup$ are lifted in a pointwise fashion to $(\Sigma', \mathcal{F})$-expansions of $\mathcal{A}$.[4]

Next, we introduce higher-order constrained Horn clauses and their satisfiability problem.

Definition 9. (i) An *atom* is a Σ'-formula that does not contain a logical symbol.
(ii) An atom is a *background atom* if it is also a 1st-order Σ-formula. Otherwise it is a *foreground atom*.

Note that a foreground atom has one of the following forms: (i) $R\,\overline{M}$ where $R \in (\Sigma' \setminus \Sigma)$, (ii) $x\,\overline{M}$, or (iii) $(\lambda y.\,N)\overline{M}$.

We use φ and A (and variants thereof) to refer to background atoms and general atoms, respectively.

Definition 10 (HoCHC)**.** (i) A *goal clause* is a disjunction $\neg A_1 \vee \cdots \vee \neg A_n$, where each A_i is an atom. We write $\bot$ to mean the empty (goal) clause.
(ii) If G is a goal clause, $R \in (\Sigma' \setminus \Sigma)$ and the variables in $\overline{x}$ are distinct, then $G \vee R\,\overline{x}$ is a *definite clause*.
(iii) A *(higher-order) constrained Horn clause (HoCHC)* is a goal or definite clause.

In the following we transform the higher-order sentences in Ex. 1 into HoCHCs (by first converting to prenex normal form and then omitting the universal quantifiers).

Example 11 (A system of HoCHCs)**.** Let $\Sigma' = \Sigma_{\mathrm{LIA}} \cup \{\mathrm{Add} : \iota \to \iota \to \iota \to o, \mathrm{Iter} : (\iota \to \iota \to \iota \to o) \to \iota \to \iota \to \iota \to o\}$ and let Δ be a type environment satisfying $\Delta(x) = \Delta(y) = \Delta(z) = \Delta(n) = \Delta(s) = \iota$ and $\Delta(f) = \iota \to \iota \to \iota \to o$.

$$\begin{aligned}
\neg(z = x + y) \vee \mathrm{Add}\; x\, y\, z \\
\neg(n \leq 0) \vee \neg(s = x) \vee \mathrm{Iter}\; f\, s\, n\, x \\
\neg(n > 0) \vee \neg\, \mathrm{Iter}\; f\, s\, (n-1)\, y \vee \neg(f\, n\, y\, x) \vee \mathrm{Iter}\; f\, s\, n\, x \\
\neg(n \geq 1) \vee \neg\, \mathrm{Iter}\; \mathrm{Add}\; n\, n\, x \vee \neg(x \leq n + n)
\end{aligned}$$

We refer to the first three (definite) HoCHCs as D_1 to D_3 and to the last (goal) HoCHC as G.

Definition 12. Let Γ be a set of HoCHCs, and suppose $\mathcal{F}$ is a frame expanding $\mathcal{A}$.

(i) Γ is $(\mathcal{A}, \mathcal{F})$*-satisfiable* if there exists a $(\Sigma', \mathcal{F})$-expansion $\mathcal{B}$ of $\mathcal{A}$ satisfying $\mathcal{B} \models \Gamma$.
(ii) Γ is $\mathcal{A}$*-Henkin-satisfiable* if it is $(\mathcal{A}, \mathcal{F})$-satisfiable for some frame $\mathcal{F}$ expanding $\mathcal{A}$.
(iii) Γ is $\mathcal{A}$*-standard-satisfiable* if it is $(\mathcal{A}, \mathcal{S})$-satisfiable.
(iv) Γ is $\mathcal{A}$*-monotone-satisfiable* if it is $(\mathcal{A}, \mathcal{M})$-satisfiable.
(v) Γ is $\mathcal{A}$*-continuous-satisfiable* if it is $(\mathcal{A}, \mathcal{C})$-satisfiable.

Observe that $\mathcal{A}$-Henkin satisfiability is trivially implied by all notions of satisfiability in Def. 12.

[3] with the usual types $0, 1 : \iota$; $+, - : \iota \to \iota \to \iota$ and $\lhd : \iota \to \iota \to o$ for $\lhd \in \{<, \leq, =, \neq, \geq, >\}$; and we use the common abbreviation n for $\underbrace{1 + \cdots + 1}_{n}$, where $1 \leq n \in \mathbb{N}$

[4] This is possible because $(\Sigma', \mathcal{F})$-expansions of $\mathcal{A}$ agree on symbols of type $\iota^n \to \iota$.

C. Programs

Whilst HoCHCs have a simple syntax (thus yielding a simple proof system), our completeness proof relies on programs, which are syntactically slightly more complex.

Definition 13. A *program* (usually denoted by Π) is a set of Σ'-formulas $\{\neg F_R \vee R\,\overline{x}_R \mid R \in (\Sigma' \setminus \Sigma)\}$ such that for each $R \in \Sigma' \setminus \Sigma$, F_R is positive existential, the variables in $\overline{x}_R$ are distinct, and $\mathrm{fv}(F_R) \subseteq \mathrm{fv}(R\,\overline{x}_R)$.

For each goal clause G there is a closed positive existential formula[5] $\mathrm{posex}(G)$ such that for each frame $\mathcal{F}$ and $(\Sigma', \mathcal{F})$-structure $\mathcal{B}$, $\mathcal{B} \not\models G$ iff $\mathcal{B} \models \mathrm{posex}(G)$. Similarly, for each finite set of HoCHCs Γ, there exists a program[5] Π_Γ such that for each frame $\mathcal{F}$ and $(\Sigma', \mathcal{F})$-structure $\mathcal{B}$, $\mathcal{B} \models \{D \in \Gamma \mid D \text{ definite}\}$ iff $\mathcal{B} \models \Pi_\Gamma$.

Example 14 (Program)**.** The following program corresponds to the set of (definite) HoCHCs of Ex. 11 (modulo renaming of variables):

$$\begin{aligned}&\neg(z = x + y) \vee \mathrm{Add}\; x\, y\, z\\ &\neg\big((n \leq 0 \;\wedge\; s = x) \vee\\ &\qquad(\exists y.\, n > 0 \wedge \mathrm{Iter}\; f\, s\, (n-1)\, y \wedge f\, n\, y\, x)\big) \vee \mathrm{Iter}\; f\, s\, n\, x.\end{aligned}$$

III. Canonical Model Property

The introduction of monotone semantics for HoCHC in [1] was partly motivated by the observation that the least model property (w.r.t. the pointwise ordering $\sqsubseteq$) fails for standard semantics (but holds for monotone semantics):

Example 15. Consider the program Π

$$\neg x_R\, U \vee R\, x_R \qquad\qquad \neg x_U \neq x_U \vee U\, x_U$$

with signature $\Sigma' = \Sigma_{\mathrm{LIA}} \cup \{R : ((\iota \to o) \to o) \to o, U : \iota \to o\}$, a type environment Δ satisfying $\Delta(x_R) = (\iota \to o) \to o$ and $\Delta(x_U) = \iota$ taken from [1]. Let $\mathcal{F} = \mathcal{S}$ be the standard frame and let $\mathrm{neg} \in \mathcal{S}[\![(\iota \to o) \to o]\!]$ be such that $\mathrm{neg}(s) = 1$ iff $s = \bot_{\iota\to o}$.

There are (at least) two expansions $\mathcal{B}_1$ and $\mathcal{B}_2$ defined by $U^{\mathcal{B}_1} = \bot_{\iota\to o}$ and $R^{\mathcal{B}_1}(s) = 1$ iff $s(\bot_{\iota\to o}) = 1$, and $U^{\mathcal{B}_2} = \top_{\iota\to o}$ and $R^{\mathcal{B}_2}(s) = 1$ iff $s(\top_{\iota\to o}) = 1$, respectively.

Note that $\mathcal{B}_1 \models \Pi$, $\mathcal{B}_2 \models \Pi$ and there are no models smaller than any of these with respect to the pointwise ordering $\sqsubseteq$. Furthermore, neither $\mathcal{B}_1 \sqsubseteq \mathcal{B}_2$ nor $\mathcal{B}_2 \sqsubseteq \mathcal{B}_1$ holds because $R^{\mathcal{B}_1}(\mathrm{neg}) = 1 > 0 = R^{\mathcal{B}_2}(\mathrm{neg})$ and for any $n \in \mathcal{S}[\![\iota]\!]$, $U^{\mathcal{B}_2}(n) = 1 > 0 = U^{\mathcal{B}_1}(n)$.

In this section, we sharpen and extend the result: HoCHC *does* enjoy a *canonical* (though not least w.r.t. $\sqsubseteq$) model property. More precisely, the structure obtained by iterating the *immediate consequence operator* (see e.g. [17]) is a model of all satisfiable HoCHCs.

Assumption. *For Sec. III and IV we fix a complete frame $\mathcal{F}$ expanding $\mathcal{A}$. Furthermore, let Γ be a finite set of HoCHCs and let $\Pi = \Pi_\Gamma$ (the program corresponding to Γ).*

If no confusion arises, we refrain from mentioning Σ', Δ and $\mathcal{F}$ explicitly.

Given an expansion $\mathcal{B}$ of $\mathcal{A}$, the *immediate consequence operator* T_Π returns the expansion $T_\Pi(\mathcal{B})$ of $\mathcal{A}$ defined by $R^{T_\Pi(\mathcal{B})} := \mathcal{B}[\![\lambda \overline{x}_R.\, F_R]\!]$, for relational symbols $R \in \Sigma' \setminus \Sigma$. (Recall that F_R is the unique positive existential formula such that $\neg F_R \vee R\,\overline{x}_R \in \Pi$.) Observe that the prefixed points of T_Π (i.e. structures $\mathcal{B}$ such that $T_\Pi(\mathcal{B}) \sqsubseteq \mathcal{B}$) are precisely the models of Π.

Unfortunately, the immediate consequence operator is not monotone w.r.t. $\sqsubseteq$. Hence, we cannot apply the Knaster-Tarski theorem. Therefore, we introduce the notion of *quasi-monotonicity* and a slightly stronger version of that theorem. This is a warm-up for Sec. IV-A, where we propose *quasi-continuity* and a version of Kleene's fixed point theorem.

A. Quasi-monotonicity

Assumption. *Let L be a complete lattice ordered by $\leq$ with least upper bounds $\bigvee$ and least element $\bot$. Furthermore, let $F : L \to L$ be an (endo-)function.*

We define

$$\begin{aligned} a_{\beta+1} &:= F(a_\beta) && (\beta \in \mathbf{On})\\ a_\gamma &:= \bigvee_{\beta<\gamma} a_\beta && (\gamma \in \mathbf{Lim})\\ a_F &:= \bigvee_{\beta\in\mathbf{On}} a_\beta \end{aligned}$$

In particular, $a_0 = \bot$. Clearly, $a_F, a_\beta \in L$ for all ordinals β.

Definition 16. Let $\precsim \subseteq L \times L$ be a relation.

(i) $\precsim$ is *compatible* with $\leq$ if
 (C1) for all $a, b, c \in L$, if $a \precsim b$ and $b \leq c$ then $a \precsim c$,
 (C2) for all $a \in L$ and $A \subseteq \{b \in L \mid b \precsim a\}$, $\bigvee A \precsim a$.

(ii) F is *quasi-monotone* if for all $a, b \in L$, $a \precsim b$ implies $F(a) \precsim F(b)$.

In particular, $\leq$ is compatible to itself and $\bot \precsim a$ for $a \in L$.

Proposition 17. *(i) $F(a_F) \leq a_F$ and (ii) if $\precsim$ is compatible with $\leq$, F is quasi-monotone and $b \in L$ satisfies $F(b) \leq b$ then $a_F \precsim b$.*

The proof idea is the same as for the standard Knaster-Tarski theorem, which can be recovered from the above by using $\leq$ for $\precsim$.

B. Application to the Immediate Consequence Operator

The idea now is to instantiate L with the complete lattice of expansions of $\mathcal{A}$ (see Remark 8(ii)), and F with the immediate

[5] see [15] for details

consequence operator T_Π. We denote the structure at stage β by $\mathcal{A}_\beta$ and the limit structure by $\mathcal{A}_\Pi$.

Intuitively, we start from the $\sqsubseteq$-minimal structure assigning $\bot_\rho$ to every $R : \rho \in \Sigma' \setminus \Sigma$ and we incrementally extend the structure to satisfy more of the program. $\mathcal{A}_\Pi$ is a prefixed point of T_Π (Prop. 17). Therefore,

Corollary 18. *$\mathcal{A}_\Pi \models \Pi$ and $\mathcal{A}_\Pi \models \{D \in \Gamma \mid D$ definite$\}$.*

Next, suppose there are relations $\precsim_\sigma \subseteq \mathcal{F}[\![\sigma]\!] \times \mathcal{F}[\![\sigma]\!]$ (for types σ) compatible with $\sqsubseteq_\sigma$, and

(i) if $\mathcal{B} \precsim \mathcal{B}'$ and $\alpha \precsim \alpha'$ then $\mathcal{B}[\![M]\!](\alpha) \precsim \mathcal{B}'[\![M]\!](\alpha')$, and
(ii) $b \precsim b'$ iff $b \leq b'$ for $b, b' \in \mathcal{B}[\![o]\!] = \mathbb{B}$,

where we omit type subscripts and lift $\precsim$ in the usual pointwise manner to structures and valuations. Then T_Π is quasi-monotone. Besides, if $\mathcal{B}$ is an expansion of $\mathcal{A}$ satisfying $\mathcal{B} \models \Pi$ then by Prop. 17, for closed positive existential formulas F, $\mathcal{A}_\Pi[\![F]\!] \leq \mathcal{B}[\![F]\!]$. Consequently, $\mathcal{A}_\Pi \models \Gamma$ if $\mathcal{B} \models \Gamma$.

The main obstacle (and where $\sqsubseteq$ fails) is to ensure that $\precsim$ is compatible with applications, i.e. if $r \precsim_{\tau\to\rho} r'$ and $s \precsim_\tau s'$ then $r(s) \precsim_\rho r'(s')$. Therefore, we simply *define* it that way:

Definition 19. We define a relation $\precsim_\sigma \subseteq \mathcal{F}[\![\sigma]\!] \times \mathcal{F}[\![\sigma]\!]$ as follows by recursion on the type σ:

$$\begin{array}{lll}
n \precsim_\iota n' & := \; n = n' & (n, n' \in \mathcal{F}[\![\iota]\!]) \\
b \precsim_o b' & := \; b \leq b' & (b, b' \in \mathcal{F}[\![o]\!]) \\
r \precsim_{\tau\to\sigma} r' & := \; \forall s, s' \in \mathcal{F}[\![\tau]\!].\, s \precsim_\tau s' \to & \\
 & \qquad\qquad r(s) \precsim_\sigma r'(s') & (r, r' \in \mathcal{F}[\![\tau \to \sigma]\!])
\end{array}$$

$\precsim$ is transitive but neither reflexive (Ex. 20(iii)) nor antisymmetric, in general, and coincides with the pointwise ordering $\sqsubseteq$ on the monotone frame $\mathcal{M}$ [15].

Example 20. (i) For all relational types ρ and $s \in \mathcal{F}[\![\rho]\!]$, $\bot_\rho \precsim s \precsim \top_\rho$.
(ii) or $\precsim$ or, and $\precsim$ and and for argument types τ, $\mathrm{exists}_\tau \precsim \mathrm{exists}_\tau$.
(iii) Let $\mathcal{F} = \mathcal{S}$ be the standard frame and let $\mathrm{neg} \in \mathcal{S}[\![(\iota \to o) \to o]\!]$ as in Ex. 15. Recall that $\bot_{\iota\to o} \precsim \top_{\iota\to o}$. However, $\mathrm{neg}(\bot_{\iota\to o}) = 1 > 0 = \mathrm{neg}(\top_{\iota\to o})$. This shows that $\precsim$ is not reflexive, in general.

Example 21. For the structures $\mathcal{B}_1$ and $\mathcal{B}_2$ of Ex. 15 it holds that $\mathcal{B}_1 = \mathcal{A}_\Pi$ and $\mathcal{B}_1 \precsim \mathcal{B}_2$ because due to $\bot_{\iota\to o} \precsim \top_{\iota\to o}$, for any $s \precsim s'$, $s(\bot_{\iota\to o}) \leq s'(\top_{\iota\to o})$ and therefore $R^{\mathcal{B}_1}(s) \leq R^{\mathcal{B}_2}(s')$. In particular, the fact that $R^{\mathcal{B}_1}(\mathrm{neg}) > R^{\mathcal{B}_2}(\mathrm{neg})$ is not a concern because $\mathrm{neg} \precsim \mathrm{neg}$ does *not* hold.

A simple induction on the type σ shows that $\precsim_\sigma$ is compatible with $\sqsubseteq_\sigma$. Furthermore,

Lemma 22. *Let $\mathcal{B} \precsim \mathcal{B}'$ be expansions of $\mathcal{A}$, $\alpha \precsim \alpha'$ be valuations and let M be a positive existential term. Then $\mathcal{B}[\![M]\!](\alpha) \precsim \mathcal{B}'[\![M]\!](\alpha')$.*

Consequently, the immediate consequence operator is quasi-monotone and we conclude:

Theorem 23. *If Γ is $(\mathcal{A}, \mathcal{F})$-satisfiable then $\mathcal{A}_\Pi \models \Gamma$.*

IV. Resolution Proof System

Our resolution proof system is remarkably simple, consisting of only three rules: (i) a higher-order version of the usual resolution rule [3] between a pair of goal and definite clauses (thus yielding a goal clause), (ii) a rule for β-reductions on leftmost (outermost) positions of atoms in goal clauses and (iii) a rule to refute certain goal clauses which are not satisfied by the model of the background theory (similar to [20]).

Resolution $$\frac{\neg R\,\overline{M} \vee G \qquad G' \vee R\,\overline{x}}{G \vee \big(G'[\overline{M}/\overline{x}]\big)}$$

β-Reduction $$\frac{\neg(\lambda x.\, L) M\, \overline{N} \vee G}{\neg L[M/x]\, \overline{N} \vee G}$$

Constraint refutation $$\frac{G \vee \neg\varphi_1 \vee \cdots \vee \neg\varphi_n}{\bot}$$

provided that each atom in G has the form[6] $x\,\overline{M}$, each φ_i is a background atom and there exists a valuation α such that $\mathcal{A}, \alpha \models \varphi_1 \wedge \cdots \wedge \varphi_n$.

Example 24 (Refutation proof)**.** A refutation of the set of HoCHCs from Ex. 11 is given in Fig. 2. The last inference is admissible because for any valuation satisfying $\alpha(n) = \alpha(y) = 1$ and $\alpha(x) = 2$,

$$\begin{aligned}\mathcal{A}_{\mathrm{LIA}}, \alpha \models{}& (n \geq 1) \wedge (n > 0) \wedge (n - 1 \leq 0) \wedge \\ & (n = y) \wedge (x = n + y) \wedge (x \leq n + n).\end{aligned}$$

Since variables are implicitly universally quantified, the rules have to be applied modulo the renaming of (free) variables; we write $\Gamma' \vdash_{\mathcal{A}} \Gamma' \cup \{G\}$ if G can be thus derived from the clauses in Γ' using the above rules and $\vdash_{\mathcal{A}}^*$ for the reflexive, transitive closure of $\vdash_{\mathcal{A}}$.

Proposition 25 (Soundness)**.** *Let Γ be a set of HoCHCs.*

If $\Gamma \vdash_{\mathcal{A}}^* \Gamma' \cup \{\bot\}$ (for some Γ') then Γ is $(\mathcal{A}, \mathcal{F})$-unsatisfiable, and this holds even if $\mathcal{F}$ is not complete.

Proof sketch. The most interesting case occurs when the constraint refutation rule is applied to $G := \bigvee_{i=1}^m \neg x_i\, \overline{M}_i \vee \bigvee_{j=1}^n \neg\varphi_j$. Being of relational type, each variable x_i cannot occur in any φ_j. Thus, modifying witnesses α of $\mathcal{A}, \alpha \models \varphi_1 \wedge \cdots \wedge \varphi_n$ to satisfy $\alpha'(x) = \top_\rho$ for $x : \rho \in \Delta$, we conclude $\mathcal{B}, \alpha' \not\models G$ for all expansions $\mathcal{B}$ of $\mathcal{A}$. □

Observe that the argument makes use of $\top_\rho \in \mathcal{F}[\![\rho]\!]$, which is a consequence of the comprehension axiom.

The following completeness theorem is significantly more difficult. In fact, we will not prove it until Sec. IV-D.

Theorem 26 (Completeness)**.** *If Γ is $(\mathcal{A}, \mathcal{F})$-unsatisfiable then $\Gamma \vdash_{\mathcal{A}}^* \{\bot\} \cup \Gamma'$ for some Γ'.*

[6] where x is a variable

$$
\begin{array}{rcl}
 & \overbrace{\neg(n \geq 1) \vee \colorbox{lightgray}{$\neg\,\mathrm{Iter\ Add}\,n\,n\,x$} \vee \neg(x \leq n+n)}^{G} & D_3 \\ \hline
\text{Resolution} & \neg(n \geq 1) \vee \underline{\neg(n > 0)} \vee \underline{\neg\,\mathrm{Iter\ Add}\,n\,(n-1)\,y} \vee \colorbox{lightgray}{$\underline{\neg\,\mathrm{Add}\,n\,y\,x}$} \vee \neg(x \leq n+n) & D_1 \\ \hline
\text{Resolution} & \neg(n \geq 1) \vee \neg(n > 0) \vee \colorbox{lightgray}{$\neg\,\mathrm{Iter\ Add}\,n\,(n-1)\,y$} \vee \underline{\neg(x = n+y)} \vee \neg(x \leq n+n) & D_2 \\ \hline
\text{Resolution} & \neg(n \geq 1) \vee \neg(n > 0) \vee \underline{\neg(n-1 \leq 0)} \vee \underline{\neg(n = y)} \vee \neg(x = n+y) \vee \neg(x \leq n+n) & \\ \hline
\text{Constraint refutation} & \bot &
\end{array}
$$

Figure 2. Refutation of the set of HoCHCs from Ex. 11. Atoms involved in resolution steps are shaded; atoms that are added are wavy-underlined.

Consequently, the resolution proof system gives rise to a semi-decision procedure for the $(\mathcal{A},\mathcal{F})$-unsatisfiability problem provided it is (semi-)decidable whether a goal clause of background atoms is not satisfied by the background theory[7].

Outline of the Completeness Proof:

- **(S1)** First, we prove that some goal clause is not satisfied by the canonical structure already after a *finite* number of iterations if Γ is $(\mathcal{A},\mathcal{F})$-unsatisfiable (Sec. IV-A).
- **(S2)** Consequently, there is a *syntactic* reason for Γ's $(\mathcal{A},\mathcal{F})$-unsatisfiability (by "unfolding definitions") (Sec. IV-B).
- **(S3)** Finally, we prove that the "unfolding" actually only needs to take place at the leftmost (outermost) positions of atoms (Sec. IV-C), which can be captured by the resolution proof system (Sec. IV-D).

Observe that Proof Step **(S1)** is model theoretic / semantic, whilst Proof Steps **(S2)** and **(S3)** are proof theoretic / syntactic.

A. *Quasi-Continuity*

Whilst in Sec. III we have shown that $\mathcal{A}_\Pi$ is a model of the definite clauses, we now examine the consequences of $\mathcal{A}_\Pi \not\models G$ for some goal clause $G \in \Gamma$. Unlike the 1st-order case [21], stage ω is not a fixed point of T_Π in general, as the following example illustrates:

Example 27. Consider the following program:

$$\neg(x_R = 0 \vee R\,(x_R - 1)) \vee R\,x_R \qquad\qquad \neg(x_U\,R) \vee U\,x_U$$

where $\Sigma' = \Sigma_{\mathrm{LIA}} \cup \{R : \iota \to o, U : ((\iota \to o) \to o) \to o\}$, $\Delta(x_R) = \iota$ and $\Delta(x_U) = (\iota \to o) \to o$. Let $\mathcal{A}$ be the standard model of Linear Integer Arithmetic $\mathcal{A}_{\mathrm{LIA}}$ and let $\mathcal{F} = \mathcal{S}$ be the standard frame. For ease of notation, we introduce functions $r_\alpha : \mathcal{S}[\![\iota]\!] \to \mathbb{B}$ such that $r_\alpha(n) = 1$ iff $0 \leq n < \alpha$, and $\delta_\alpha : \mathcal{S}[\![\iota \to o]\!] \to \mathbb{B}$ such that $\delta_\alpha(r) = 1$ iff $r = r_\alpha$, where $\alpha \in \omega \cup \{\omega\}$. Then it holds $R^{\mathcal{A}_n} = r_n$, $U^{\mathcal{A}_0} = \bot_{(\iota\to o)\to o}$ and $U^{\mathcal{A}_n}(s) = s(r_{n-1})$ for $n > 0$. Therefore $R^{\mathcal{A}_\omega} = r_\omega$ and $U^{\mathcal{A}_\omega}(s) = 1$ iff there exists $n < \omega$ satisfying $s(r_n) = 1$. In particular, $U^{\mathcal{A}_\omega}(\delta_\omega) = 0$. On the other hand, $U^{\mathcal{A}_{\omega+1}}(\delta_\omega) = \mathcal{A}_\omega[\![\lambda x_U.\, x_U\,R]\!](\delta_\omega) = 1$. Consequently, $\mathcal{A}_\omega \neq \mathcal{A}_{\omega+1}$.

Nonetheless, there still exists a (finite) $n \in \omega$ satisfying $\mathcal{A}_n \not\models G$ if $\mathcal{A}_\Pi \not\models G$ (Thm. 32). We make use of a similar strategy to establishing the canonical model property: we introduce the notion of *quasi-continuity*, state a version of Kleene's fixed point theorem and prove the immediate consequence operator to be quasi-continuous.

Definition 28. Let $\precsim \subseteq L \times L$ be a relation.

(i) $\precsim$ is *$\precsim$-directed* if for every $a, b \in L$, $a \precsim a$ and there exists $c \in L$ satisfying $a, b \precsim c$.
For $a \in L$ we write $\mathrm{dir}_{\precsim}(a)$ for the set of $\precsim$-directed subsets D of L satisfying $a \precsim \bigvee D$.

(ii) F is *quasi-continuous* if for all $a \in L$ and $D \in \mathrm{dir}_{\precsim}(a)$, $F(a) \precsim \bigvee_{b \in D} F(b)$.

Thus, every quasi-continuous function is in particular quasi-monotone if $\precsim$ is reflexive.

Proposition 29. *If $\precsim$ is compatible with $\leq$ and F is quasi-continuous then (i) for all ordinals $\beta \leq \beta'$, $a_\beta \precsim a_{\beta'}$ and (ii) $a_F \precsim a_\omega$.*

Combined with Prop. 17 this yields Kleene's fixed point theorem in the case of $\precsim := \leq$.

Similarly as in Sec. III, we need relations $\precsim_\sigma \subseteq \mathcal{F}[\![\sigma]\!] \times \mathcal{F}[\![\sigma]\!]$ which behave well with applications in order for the immediate consequence operator to be quasi-continuous. Therefore, we stipulate (overloading the notation of Sec. III):

Definition 30. We define $\precsim_\sigma \subseteq \mathcal{F}[\![\sigma]\!] \times \mathcal{F}[\![\sigma]\!]$ by recursion on the type σ:

$$
\begin{array}{lll}
b \precsim_o b' & := \; b \leq b' & (b, b' \in \mathcal{F}[\![o]\!]) \\
n \precsim_\iota n' & := \; n = n' & (n, n' \in \mathcal{F}[\![\iota]\!]) \\
r \precsim_{\tau\to\sigma} r' & := \; \forall s \in \mathcal{F}[\![\tau]\!], \mathfrak{S}' \in \mathrm{dir}_{\precsim_\tau}(s). & \\
 & \qquad r(s) \precsim_\sigma \bigsqcup_{s' \in \mathfrak{S}'} r'(s') & (r, r' \in \mathcal{F}[\![\tau \to \sigma]\!])
\end{array}
$$

There is an elementary inductive argument that each $\precsim_\sigma$ is compatible with $\sqsubseteq_\sigma$. We lift $\precsim$ to structures and valuations in a pointwise way, and abbreviate $\mathrm{dir}_{\precsim}$ as dir.

Lemma 31. *Let M be a positive existential term, $\mathcal{B}$ be an expansion of $\mathcal{A}$, $\mathfrak{B}' \in \mathrm{dir}(\mathcal{B})$, α be a valuation and let $\mathfrak{a}' \in \mathrm{dir}(\alpha)$. Then*[8]

$$\mathcal{B}[\![M]\!](\alpha) \precsim \bigsqcup_{\mathcal{B}' \in \mathfrak{B}', \alpha' \in \mathfrak{a}'} \mathcal{B}'[\![M]\!](\alpha').$$

[7] i.e. whether there exists a valuation α such that $\mathcal{A}, \alpha \models \varphi_1 \wedge \cdots \wedge \varphi_n$

[8] By Remark 8(i) the right-hand side is well-defined.

Consequently, the immediate consequence operator is quasi-continuous. Moreover, by Prop. 29, for closed positive existential formulas F, $\mathcal{A}_\Pi[\![F]\!] \leq \max_{n\in\omega} \mathcal{A}_n[\![F]\!]$. Therefore, we get the following result, which is key for the refutational completeness of the proof system.

Theorem 32. *Let G be a goal clause. If $\mathcal{A}_\Pi \not\models G$ then there exists $n \in \omega$ such that $\mathcal{A}_n \not\models G$.*

B. Syntactic Unfolding

Having established Proof Step **(S1)**, we study a functional relation $\rightarrow_{\|}$ on positive existential terms, which is a syntactic counterpart of the immediate consequence operator. Essentially[9], it holds $M \rightarrow_{\|} N$ if N is obtained from M by replacing all occurrences of symbols $R \in \Sigma'\backslash\Sigma$ with $\lambda\overline{x}_R.\, F_R$, which is reminiscent of the definition of $R^{T_\Pi(\mathcal{B})}$. Therefore:

Proposition 33. *Let $\mathcal{B}$ be an expansion of $\mathcal{A}$ and let M and N be positive existential terms satisfying $M \rightarrow_{\|} N$. Then for all valuations α, $T_\Pi(\mathcal{B})[\![M]\!](\alpha) = \mathcal{B}[\![N]\!](\alpha)$.*

A similar idea is exploited in [17].

Next, let $\upsilon := \{(R, \lambda\overline{x}_R.\, F_R) \mid R \in \Sigma'\backslash\Sigma\}$ and $\beta\upsilon := \beta\cup\upsilon$. Besides, let $\rightarrow_{\beta\upsilon}$ be the compatible closure [16, p. 51] of $\beta\upsilon$. It is easy to see that $\rightarrow_{\|} \subseteq \twoheadrightarrow_{\beta\upsilon}$, where $\twoheadrightarrow_{\beta\upsilon}$ is the reflexive, transitive closure of $\rightarrow_{\beta\upsilon}$.

C. Leftmost (Outermost) Reduction

There is an important mismatch between the relation $\rightarrow_{\beta\upsilon}$ and the rules of the proof system: in contrast to the former, the latter only take leftmost (outermost) positions of atoms into account. Fortunately, arbitrary sequences of $\beta\upsilon$-reductions can be mimicked by sequences which are standard in the sense that purely leftmost reductions are followed by purely non-leftmost ones (Cor. 35).

Fig. 3 defines $\xrightarrow[\ell]{\cdot}$ and $\xrightarrow[s]{}$, which formalise leftmost (outermost) and standard reductions, respectively. We write $M \underset{\ell}{\twoheadrightarrow} N$ if $M \xrightarrow[\ell]{m} N$ for some m, where m corresponds to the number of leftmost $\beta\upsilon$-reductions having been performed. The idea is that $L \xrightarrow[s]{} N$ if for some M, $L \underset{\ell}{\twoheadrightarrow} M$ and we can obtain N from M by performing standard $\beta\upsilon$-reductions only on non-leftmost positions.

Lemma 34. *If $K \xrightarrow[s]{} M \rightarrow_{\beta\upsilon} N$ then $K \xrightarrow[s]{} N$.*

The proof of this proposition is very similar to the proof of the standardisation theorem in the λ-calculus as presented in [10] and relies on the insight that if all of $\overline{K} \xrightarrow[s]{} \overline{M}$, $K' \xrightarrow[s]{} M'$ and $O \xrightarrow[s]{} Q$ hold then $K'[O/x]\,\overline{K} \xrightarrow[s]{} M'[Q/x]\,\overline{M}$.

Corollary 35. *Let M and N be positive existential terms such that $M \twoheadrightarrow_{\beta\upsilon} N$. Then $M \xrightarrow[s]{} N$.*

Next, we consider the relation $\triangleright$ on positive existential formulas and valuations inductively defined by:

$$\frac{}{\alpha \triangleright x\,\overline{M}} \qquad \frac{\mathcal{A}, \alpha \models \varphi}{\alpha \triangleright \varphi} \qquad \frac{r \in \mathcal{F}[\![\Delta(x)]\!] \quad \alpha[x \mapsto r] \triangleright M}{\alpha \triangleright \exists x.\, M}$$

$$\frac{i \in \{1,2\} \quad \alpha \triangleright M_i}{\alpha \triangleright M_1 \vee M_2} \qquad \frac{\alpha \triangleright M_1 \quad \alpha \triangleright M_2}{\alpha \triangleright M_1 \wedge M_2}$$

Intuitively, $\alpha \triangleright F$ if for some α' (agreeing with α on $\Delta^{-1}(\iota)$), $\mathcal{A}_0, \alpha' \models F$ and there are no λ-abstractions in relevant leftmost positions.

Remark 36. If G is a goal clause and $\alpha \triangleright \mathrm{posex}(G)$ (for some α) then G has the form $\bigvee_{i=1}^m \neg x_i\,\overline{M}_i \vee \bigvee_{j=1}^n \neg\varphi_j$ and G can be refuted by the constraint refutation rule in one step.

Lemma 37. *Let G be a goal clause, F be a β-normal positive existential formula and α be a valuation such that $\mathcal{A}_0, \alpha \models F$ and $\mathrm{posex}(G) \xrightarrow[s]{} F$. Then there exists a positive existential formula F' satisfying $\mathrm{posex}(G) \underset{\ell}{\twoheadrightarrow} F'$ and $\alpha \triangleright F'$.*

D. Concluding Refutational Completeness

Finally, we establish a connection between the (abstract) relation $\underset{\ell}{\twoheadrightarrow}$ on positive existential terms and the resolution proof system on clauses. We define a function μ assigning natural numbers or ω to positive existential formulas E by

$$\mu(E) := \min\Big(\{\omega\} \cup \{m \mid E \xrightarrow[\ell]{m} F \text{ and } \alpha \triangleright F \text{ for some } \alpha\}\Big)$$

which is extended to non-empty sets Γ' of HoCHCs by $\mu(\Gamma') := \min\{\mu(\mathrm{posex}(G)) \mid G \in \Gamma\}$.

We can use the resolution proof system to derive a set of HoCHCs Γ'' with a strictly smaller measure by simulating a $\xrightarrow[\ell]{1}$-reduction step:

Proposition 38. *Let $\Gamma' \supseteq \Gamma$ be a set of HoCHCs satisfying $0 < \mu(\Gamma') < \omega$. Then there exists $\Gamma'' \supseteq \Gamma$ satisfying $\Gamma' \vdash_{\mathcal{A}} \Gamma''$ and $\mu(\Gamma'') < \mu(\Gamma')$.*

Example 39. Consider the HoCHCs $\Gamma = \{\neg(x_R \geq 5) \vee R\,x_R, \neg R\,(x_R - 5) \vee R\,x_R, \neg R\,5\}$. It holds that $R\,5 \xrightarrow[\ell]{1} (\lambda x_R.\, x_R \geq 5 \vee R\,(x_R - 5))\,5 \xrightarrow[\ell]{1} 5 \geq 5 \vee R\,(5 - 5)$ and $\mu(\Gamma) = 2$. Furthermore, $\Gamma \vdash_{\mathcal{A}} \Gamma \cup \{\neg 5 \geq 5\}$ and $\mu(\Gamma \cup \{\neg 5 \geq 5\}) = 0$.

Combining everything, we finally obtain:

Theorem 26 (Completeness)**.** *If Γ is $(\mathcal{A}, \mathcal{F})$-unsatisfiable then $\Gamma \vdash_{\mathcal{A}}^* \{\bot\} \cup \Gamma'$ for some Γ'.*

Proof. By Cor. 18, $\mathcal{A}_\Pi \models D$ for all definite clauses $D \in \Gamma$. Since Γ is $(\mathcal{A}, \mathcal{F})$-unsatisfiable there exists a goal clause $G \in \Gamma$ satisfying $\mathcal{A}_\Pi \not\models G$. By Thm. 32 there exists $n \in \omega$ such that $\mathcal{A}_n \not\models G$. Let F_n be such that $\mathrm{posex}(G) \rightarrow_{\|}^n F_n$ (where $\rightarrow_{\|}^n$ is the n-fold composition of $\rightarrow_{\|}$). By Prop. 33, $\mathcal{A}_0, \alpha \models F_n$ (for any α as F_n is closed). Let F_n' be the β-normal form of F_n. By Cor. 35 and Lemma 37 there exists F' such that

[9] For a formal definition refer to [15].

$$\frac{}{R\,\overline{M} \xrightarrow[\ell]{1} (\lambda \overline{x}_R.\, F_R)\overline{M}}\ R \in \Sigma' \setminus \Sigma \qquad \frac{}{(\lambda x.\, L)M\,\overline{N} \xrightarrow[\ell]{1} L[M/x]\overline{N}} \qquad \frac{M_1 \xrightarrow[\ell]{m_1} N_1 \qquad M_2 \xrightarrow[\ell]{m_2} N_2}{M_1 \circ M_2 \xrightarrow[\ell]{m_1+m_2} N_1 \circ N_2}\ \circ \in \{\wedge, \vee\}$$

$$\frac{M \xrightarrow[\ell]{m} N}{\exists x.\, M \xrightarrow[\ell]{m} \exists x.\, N} \qquad \frac{}{M \xrightarrow[\ell]{0} M} \qquad \frac{L \xrightarrow[\ell]{m_1} M \quad M \xrightarrow[\ell]{m_2} N}{L \xrightarrow[\ell]{m_1+m_2} N}$$

(a) Definition of leftmost (outermost) reductions.

$$\frac{\overline{M} \underset{s}{\rightarrow} \overline{N}}{L \underset{s}{\rightarrow} c\,\overline{N}}\ L \underset{\ell}{\twoheadrightarrow} c\,\overline{M}, c \in \Sigma' \cup \{\wedge, \vee, \exists_\tau\} \qquad \frac{\overline{M} \underset{s}{\rightarrow} \overline{N}}{L \underset{s}{\rightarrow} x\,\overline{N}}\ L \underset{\ell}{\twoheadrightarrow} x\,\overline{M} \qquad \frac{M' \underset{s}{\rightarrow} N' \quad \overline{M} \underset{s}{\rightarrow} \overline{N}}{L \underset{s}{\rightarrow} (\lambda x.\, N')\overline{N}}\ L \underset{\ell}{\twoheadrightarrow} (\lambda x.\, M')\overline{M} \quad .$$

(b) Definition of standard reductions (by $\overline{M} \underset{s}{\rightarrow} \overline{N}$ we mean $M_j \underset{s}{\rightarrow} N_j$ for each $1 \leq j \leq n$, assuming $\overline{M}$ is $M_1, \ldots, M_n$ and $\overline{N}$ is $N_1, \ldots, N_n$).

Figure 3. Leftmost outermost and standard reductions.

$\mathrm{posex}(G) \underset{\ell}{\twoheadrightarrow} F'$ and $\alpha \rhd F'$. Consequently, $\mu(\Gamma) < \omega$. By Prop. 38 there exists $\Gamma' \supseteq \Gamma$ satisfying $\Gamma \vdash^*_{\mathcal{A}} \Gamma'$ and $\mu(\Gamma') = 0$.

Hence, there exists $G \in \Gamma'$ and α such that $\mathrm{posex}(G) \xrightarrow[\ell]{0} F'$ and $\alpha \rhd F'$. Clearly, this implies $F' = \mathrm{posex}(G)$, and by Remark 36, $\Gamma \vdash^*_{\mathcal{A}} \Gamma' \vdash_{\mathcal{A}} \{\bot\} \cup \Gamma'$. □

E. Compactness of HoCHC

The reason why we restrict Γ to be finite is to achieve correspondence with programs (Def. 13), which are finite expressions. If we simply extend programs with infinitary disjunctions (but keep HoCHCs finitary) we can carry out exactly the same reasoning to derive that also every *infinite*, $(\mathcal{A}, \mathcal{F})$-unsatisfiable set of HoCHCs can be refuted in the proof system. Consequently:

Theorem 40 (Compactness)**.** *For every $(\mathcal{A}, \mathcal{F})$-unsatisfiable set Γ of HoCHCs there exists a finite subset $\Gamma' \subseteq \Gamma$ which is $(\mathcal{A}, \mathcal{F})$-unsatisfiable.*

V. Semantic Invariance

[1] details an explicit translation between standard and monotone models of HoCHCs, thus yielding the equivalence of $\mathcal{A}$-standard- and $\mathcal{A}$-monotone-satisfiability.

As a consequence of the Completeness Thm. 26 for the proof system, $(\mathcal{A}, \mathcal{F})$-unsatisfiability for *any* complete frame $\mathcal{F}$ implies the existence of a refutation, which in turn entails $(\mathcal{A}, \mathcal{F}')$-unsatisfiability for *any* frame $\mathcal{F}'$ by the Soundness Prop. 25.

Therefore, exploiting Ex. 5, we obtain an equivalence result encompassing a much wider class of semantics:

Theorem 41 (Semantic Invariance)**.** *Let Γ be a set of HoCHCs. Then the following are equivalent:*

(i) Γ *is $\mathcal{A}$-standard-satisfiable,*
(ii) Γ *is $\mathcal{A}$-Henkin-satisfiable,*
(iii) Γ *is $\mathcal{A}$-monotone-satisfiable,*
(iv) Γ *is $\mathcal{A}$-continuous-satisfiable,*
(v) Γ *is $(\mathcal{A}, \mathcal{F})$-satisfiable, where $\mathcal{F}$ is a complete frame expanding $\mathcal{A}$.*

Thus, we call a set of Γ of HoCHCs *$\mathcal{A}$-satisfiable* if it satisfies any of the equivalent conditions of Thm. 41.

VI. Compact Theories

In this section, we extend our results to background theories (over Σ) with a *set* $\mathfrak{A}$ of models (i.e. Σ-structures), calling a set of HoCHCs *$\mathfrak{A}$-satisfiable* if it is $\mathcal{A}$-satisfiable for some $\mathcal{A} \in \mathfrak{A}$. Otherwise it is *$\mathfrak{A}$-unsatisfiable*.

Observe that the Completeness Thm. 26 critically relies on the observation that $\mathcal{A}$-unsatisfiability can be traced back to the failure of *a single* goal clause of background atoms (manifested in the constraint refutation rule). Therefore, it is natural to generalise constraint refutation to a rule refuting *sets* of $\mathfrak{A}$-unsatisfiable[10] goal clauses of background atoms:

Comp. const. refutation $\dfrac{G_1 \vee \bigvee_{i=1}^{m_1} \neg\varphi_{1,i} \quad \ldots \quad G_n \vee \bigvee_{i=1}^{m_n} \neg\varphi_{n,i}}{\bot}$

provided that each atom in each G_i has the form $x\,\overline{M}$, each $\varphi_{i,j}$ is a background atom and $\{\neg\varphi_{j,1} \vee \cdots \vee \neg\varphi_{j,m_j} \mid 1 \leq j \leq n\}$ is $\mathfrak{A}$-unsatisfiable.

and let $\vdash_{\mathfrak{A}}$ be defined accordingly. However, to match the rule's finitary nature, $\mathfrak{A}$ needs to be restricted a little:

Definition 42. A set $\mathfrak{A}$ of 1st-order Σ-structures is *compact* if for all $\mathfrak{A}$-unsatisfiable sets Γ of goal clauses of background atoms there exists a finite $\Gamma' \subseteq \Gamma$ which is $\mathfrak{A}$-unsatisfiable.

In particular, every finite $\mathfrak{A}$ is compact. Then we obtain:

Theorem 43 (Soundness and Completeness)**.** *Let $\mathfrak{A}$ be a compact set of Σ-structures and Γ be a set of HoCHCs. Then Γ is $\mathfrak{A}$-unsatisfiable iff $\Gamma \vdash^*_{\mathfrak{A}} \Gamma' \cup \{\bot\}$ for some Γ'.*

As an interesting special case, this shows that the proof system is also sound and complete in the unconstrained setting: by the compactness theorem for 1st-order logic the set of 1st-order Σ-structures (possibly interpreting (in-)equality symbols as (non-)identity) is compact. Consequently, there

[10] Note that for a set Γ of goal clauses of background atoms, $\mathcal{A}$-satisfiability is in essence not about the existence of $\mathcal{A} \in \mathfrak{A}$ and an *expansion* $\mathcal{B}$ of $\mathcal{A}$ such that $\mathcal{B} \models \Gamma$ but only about the existence of $\mathcal{A} \in \mathfrak{A}$ such that $\mathcal{A} \models \Gamma$.

does not exist a Σ'-structure $\mathcal{B}$ (interpreting (in-)equality as (non-)identity) satisfying $\mathcal{B} \models \Gamma$ iff Γ is refutable.

VII. 1st-order Translation

It is folklore that there is a 1st-order translation of higher-order logic which is sound and complete for Henkin semantics (see e.g. [22-24]). The essence of the technique is to replace all symbols by constants (of a base type) and encode application using dedicated binary function symbols.

For the reasons discussed in the introduction this translation is however not in general complete for standard semantics. In this section, we present a particularly simple 1st-order translation of HoCHC which is sound and complete even for standard semantics. Fortunately, the target fragment is still semi-decidable.

We do not need to consider HoCHCs containing λ-abstractions because they can be eliminated by a logical counterpart of λ-lifting [25] (i.e. introducing new relational symbols and adding appropriate "definitions" for them [15]. This constitutes a considerable generalisation of the "polarity-dependent renaming" for 1st-order logic [26], [27].

Therefore, the following is without loss of generality:

Assumption. *Henceforth, we fix a finite set Γ of HoCHCs which does not contain λ-abstractions and a set $\mathfrak{A}$ of 1st-order Σ-structures.*

Let $\mathfrak{I} = \{\iota\} \cup \{\lfloor\rho\rfloor \mid \rho \text{ relational}\}$ (and we set $\lfloor \iota^n \to \iota \rfloor := \iota^n \to \iota$). Clearly, we can regard Σ and each $\mathcal{A} \in \mathfrak{A}$ as a 1st-order signature and structure, respectively, over the extended set of types of individuals.

We assume a type environment $\lfloor\Delta\rfloor$ such that for $x : \tau \in \Delta$, $\lfloor\Delta\rfloor(x) = \lfloor\tau\rfloor$ and define $\lfloor\Sigma'\rfloor$ to be the following 1st-order extension of Σ:

$$\begin{aligned}
&\Sigma \cup \{c_R : \lfloor\rho\rfloor \mid R : \rho \in \Sigma' \setminus \Sigma\}\\
&\quad \cup \{c_\rho : \lfloor\rho\rfloor \mid \rho \text{ relational}\}\\
&\quad \cup \{@_{\tau,\rho} : \lfloor\tau \to \rho\rfloor \to \lfloor\tau\rfloor \to \lfloor\rho\rfloor \mid \tau \to \rho \text{ relational}\}\\
&\quad \cup \{H : \lfloor o\rfloor \to o\}
\end{aligned}$$

To reduce clutter, we often omit the subscripts from @.

Intuitively, @ encodes application, relational symbols $R \in \Sigma' \setminus \Sigma$ become constants c_R, H maps the "bogus booleans" $\lfloor o\rfloor$ to o and the following *comprehension axiom* Comp_ρ (for relational $\rho = \tau_1 \to \cdots \to \tau_n \to o$) asserts the existence of an element (the interpretation of c_ρ) corresponding to $\top_\rho$:

$$\mathrm{Comp}_\rho := H\,(@\,(\cdots(@\,(@\ c_\rho\, x_1)\, x_2)\cdots)\, x_n)$$

where the x_i are distinct variables of type $\lfloor\tau_i\rfloor$.

For a Σ'-term M containing neither logical symbols nor λ-abstractions, we define $\lfloor M\rfloor'$ by structural recursion:

$$\begin{aligned}
\lfloor x\rfloor' &:= x\\
\lfloor R\rfloor' &:= c_R && \text{if } R \in \Sigma' \setminus \Sigma\\
\lfloor c\,\overline{N}\rfloor' &:= c\,\overline{N} && \text{if } c \in \Sigma\\
\lfloor M\,\overline{N}\,N'\rfloor' &:= @\,\lfloor M\,\overline{N}\rfloor'\,\lfloor N'\rfloor' && \text{if } M \notin \Sigma
\end{aligned}$$

Thus, terms of the background theory are unchanged by $\lfloor\cdot\rfloor'$ and by Remark 2, for each Σ'-term $\Delta \vdash M : \sigma$ which is not a background atom, $\lfloor\Delta\rfloor \vdash \lfloor M\rfloor' : \lfloor\sigma\rfloor$. The following operator $\lfloor\cdot\rfloor$ ensures that also foreground atoms have type o (not $\lfloor o\rfloor$)

$$\lfloor A\rfloor := \begin{cases} A & \text{if } A = c\,\overline{N} \text{ with } c \in \Sigma\\ H\,\lfloor A\rfloor' & \text{otherwise } (A \text{ is a foreground atom}).\end{cases}$$

and we lift $\lfloor\cdot\rfloor$ to HoCHCs by

$$\lfloor(\neg)A_1 \vee \cdots \vee (\neg)A_n\rfloor := (\neg)\lfloor A_1\rfloor \vee \cdots \vee (\neg)\lfloor A_n\rfloor$$

Finally, for Γ we set

$$\lfloor\Gamma\rfloor := \{\lfloor C\rfloor \mid C \in \Gamma\} \cup \{\mathrm{Comp}_\rho \mid x : \rho \in \Delta \text{ occurs in } \Gamma\}.$$

Note that $\lfloor\Gamma\rfloor$ is a finite set of 1st-order Horn clauses[11] of the (1st-order) language of $\lfloor\Sigma'\rfloor$.

Example 44 (1st-order translation $\lfloor\cdot\rfloor$)**.** Consider again the set Γ of HoCHCs from Ex. 11. Applying the translation $\lfloor\cdot\rfloor$ to Γ we get the 1st-order clauses in Fig. 4.

For $\mathcal{A} \in \mathfrak{A}$ and a Σ'-expansion $\mathcal{B}$ of $\mathcal{A}$, let $\lfloor\mathcal{B}\rfloor$ be the 1st-order $\lfloor\Sigma\rfloor$-expansion of $\mathcal{A}$ defined by

$$\begin{gathered}
\lfloor\mathcal{B}\rfloor[\![\lfloor\rho\rfloor]\!] := \mathcal{B}[\![\rho]\!] \qquad c_R^{\lfloor\mathcal{B}\rfloor} := R^{\mathcal{B}} \qquad c_\rho^{\lfloor\mathcal{B}\rfloor} := \top_\rho\\
@_{\tau,\rho'}^{\lfloor\mathcal{B}\rfloor}(r)(s) := r(s) \qquad H^{\lfloor\mathcal{B}\rfloor}(b) := b
\end{gathered}$$

for relational ρ and $\tau \to \rho'$, $R \in \Sigma' \setminus \Sigma$, $r \in \lfloor\mathcal{B}\rfloor[\![\lfloor\tau \to \rho'\rfloor]\!]$, $s \in \lfloor\mathcal{B}\rfloor[\![\lfloor\tau\rfloor]\!]$ and $b \in \lfloor\mathcal{B}\rfloor[\![\lfloor o\rfloor]\!] = \mathbb{B}$. It is easy to see that $\mathcal{B} \models \Gamma$ implies $\lfloor\mathcal{B}\rfloor \models \lfloor\Gamma\rfloor$. Consequently:

Proposition 45. *If Γ is $\mathfrak{A}$-satisfiable then $\lfloor\Gamma\rfloor$ is $\mathfrak{A}$-satisfiable.*

Conversely, applications of the (higher-order) resolution rule can be matched by 1st-order resolution inferences between the corresponding translated clauses. Besides, the 1st-order translation contains comprehension axioms Comp_ρ, which complements the instantiation of relational variables with $\top_\rho$ in the proof of the Soundness Prop. 25. Therefore, we obtain:

Lemma 46. *Let Γ' be a set of HoCHCs not containing λ-abstractions and suppose $\Gamma' \vdash_{\mathfrak{A}} \Gamma' \cup \{G\}$. Then*

(i) *G does not contain λ-abstractions*
(ii) *if $G \neq \bot$ then $\lfloor\Gamma'\rfloor \models \lfloor\Gamma' \cup \{G\}\rfloor$*
(iii) *if $G = \bot$ then $\lfloor\Gamma'\rfloor$ is $\mathfrak{A}$-unsatisfiable.*

By the Completeness Thm. 43 we conclude:

[11] in the standard sense

$$\begin{aligned}
\lfloor D_1 \rfloor &= \neg(z = x+y) \lor H\,(@\,(@\,(@\ \mathrm{Add}\ x)\,y)\,z) \\
\lfloor D_2 \rfloor &= \neg(n \leq 0) \lor \neg(s = x) \lor H\,(@\,(@\,(@\,(@\ \mathrm{Iter}\ f)\,s)\,n)\,x) \\
\lfloor D_3 \rfloor &= \neg(n > 0) \lor \neg H\,(@\,(@\,(@\,(@\ \mathrm{Iter}\ f)\,s)\,(n-1))\,y) \lor \neg H\,(@\,(@\,(@\ f\,n)\,y)\,x) \lor H\,(@\,(@\,(@\,(@\ \mathrm{Iter}\ f)\,s)\,n)\,x) \\
\lfloor G \rfloor &= \neg(n \geq 1) \lor \neg H\,(@\,(@\,(@\,(@\ \mathrm{Iter}\ \mathrm{Add})\,n)\,n)\,x) \lor \neg(x \leq n+n) \\
\mathrm{Comp}_{\iota^3 \to o} &= H\,(@\,(@\,(@\,c_{\iota^3 \to o}\,x_1)\,x_2)\,x_3)
\end{aligned}$$

Figure 4. 1st-order translation of the set of HoCHCs of Ex. 11.

Corollary 47. *If $\mathfrak{A}$ is compact and Γ is $\mathfrak{A}$-unsatisfiable then $\lfloor \Gamma \rfloor$ is $\mathfrak{A}$-unsatisfiable.*

Theorem 48. *Assuming that $\mathfrak{A}$ is compact, Γ is $\mathfrak{A}$-satisfiable iff $\lfloor \Gamma \rfloor$ is $\mathfrak{A}$-satisfiable.*

It is remarkable that our translation does not require extensionality axioms and only a very restricted form of comprehension axioms (cf. [6]).

Finally, if $\mathfrak{A}$ is compact, *definable*[12] and $\mathfrak{A}$-unsatisfiability of goal clauses of background atoms is semi-decidable, then $\mathfrak{A}$-unsatisfiability of $\lfloor \Gamma \rfloor$ is also semi-decidable [20, Thm. 24].

VIII. Decidable Fragments

Satisfiability of HoCHC is undecidable in general because already its 1st-order fragments are undecidable for Linear Integer Arithmetic [28], [29] or the unconstrained setting[13] [30].

Remark 49. Despite these negative results, $\mathfrak{A}$-satisfiability of finite Γ is decidable if $\mathfrak{A}$ is a finite set of Σ-structures such that for each $\mathcal{A} \in \mathfrak{A}$ and type σ, $\mathcal{A}[\![\sigma]\!]$ is finite. This is a consequence of Thm. 23 and the fact that we can compute each $\mathcal{A}_{\Pi_\Gamma}$ explicitly and check whether $\mathcal{A}_{\Pi_\Gamma} \models \Gamma$ holds.

Thanks to this insight, we have identified two decidable fragments of HoCHC, one of which is presented as follows; we leave the other (higher-order Datalog) to [15].

A. Combining the Bernays-Schönfinkel-Ramsey Fragment of HoCHC with Simple Linear Integer Arithmetic

Some authors [31], [32] have studied 1st-order clauses without function symbols (the so-called *Bernays-Schönfinkel-Ramsey class*[14]) extended with a restricted form of Linear Integer Arithmetic. The fragment enjoys the attractive property that every clause set is equi-satisfiable with a finite set of its ground instances, which implies decidability [31], [32].

In this section, we transfer this result to our higher-order Horn setting.

Assumption. *Let Σ be a (1st-order) signature extending Σ_{LIA} with constant symbols $c : \iota$, and let $\Sigma' \supseteq \Sigma$ be a relational extension of Σ.*

Definition 50. (i) A Σ-atom is *simple* if it has the form $x \leq M$, $M \leq x$ or $x \leq y$, where M is closed[15].

(ii) A HoCHC is a *higher-order simple linear arithmetic Bernays-Schönfinkel-Ramsey Horn clause (HoBHC(SLA))* if it has the form $\neg\varphi_1 \lor \cdots \lor \neg\varphi_n \lor C$, where each φ_i is a simple (linear arithmetic) background atom and C is $\bot$ or it does not contain symbols from Σ.

Note that we could also have allowed background atoms of the form $M \lhd N$, $x \lhd M$ and $x \trianglelefteq y$, where M, N are closed, $\lhd \in \{<, \leq, =, \neq, \geq, >\}$ and $\trianglelefteq \in \{\leq, =, \geq\}$ [32].

Example 51. Let $\Sigma = \Sigma_{\mathrm{LIA}} \cup \{c, d : \iota\}$, $\Sigma' = \Sigma \cup \{R : \iota \to o, U : (\iota \to o) \to \iota \to o\}$, $\Delta(x) = \Delta(y) = \Delta(z) = \iota$ and $\Delta(f) = \iota \to o$. The following is a set of HoBHC(SLA):

$$\begin{aligned}
&\neg(x \leq c + d - 5) \lor R\,x \\
&\neg f\,x \lor \neg(y \leq x) \lor \neg(x \leq d) \lor U\,f\,y \\
&\neg(c \leq x) \lor \neg(x \leq -1) \\
&\neg(x \leq d - 5) \lor \neg(d - 5 \leq x) \lor \neg(y \leq c - 10) \lor \\
&\qquad \neg(c - 10 \leq y) \lor \neg U\,(\lambda z.\, R\,x)\,y.
\end{aligned}$$

Assumption. *Let $\mathfrak{A}$ be the set of Σ-expansions of $\mathcal{A}_{\mathrm{LIA}}$ and let Γ be a finite set of HoBHC(SLA).*

As in the 1st-order case, only the relations between ground terms are relevant. Therefore, we replace ground terms M with (fresh) constant symbols c_M and consider only structures in which "$\leq$" is interpreted consistently (with the meaning of the constants).

Formally, let $\mathrm{gt}_\iota(\Gamma)$ be the set of closed terms of type ι occurring in Γ and we define

$$\begin{aligned}
\Sigma^\flat &:= \{\leq : \iota \to \iota \to o\} \cup \{c_M : \iota \mid M \in \mathrm{gt}_\iota(\Gamma)\} \\
(\Sigma')^\flat &:= \Sigma^\flat \cup (\Sigma' \setminus \Sigma)
\end{aligned}$$

For $\mathcal{A} \in \mathfrak{A}$, let $\mathcal{A}^\flat$ be the 1-order $\Sigma^\flat$-structure defined by

$$\mathcal{A}^\flat[\![\iota]\!] := \mathrm{gt}_\iota(\Gamma) \quad \leq^{\mathcal{A}^\flat}(M)(N) := \mathcal{A}[\![M \leq N]\!] \quad c_M^{\mathcal{A}^\flat} := M$$

for $M, N \in \mathrm{gt}_\iota(\Gamma)$, and let $\mathfrak{A}^\flat := \{\mathcal{A}^\flat \mid \mathcal{A} \in \mathfrak{A}\}$.

[12] or *term-generated* [20], i.e. for every $\mathcal{A} \in \mathfrak{A}$ and $a \in \mathcal{A}[\![\iota]\!]$ there exists a closed Σ-term M such that $\mathcal{A}[\![M]\!] = a$

[13] i.e. the background theory imposes no restriction at all

[14] Precisely the set of sentences that, when written in prenex normal form, have a $\exists^*\forall^*$-quantifier prefix and contain no function symbols.

[15] or *ground* because atoms do not contain (existential) quantifiers by definition

Furthermore, for simple atoms $x \leq M$ and $M \leq x$, we set $(x \leq M)^\flat := x \leq c_M$ and $(M \leq x)^\flat := c_M \leq x$. For all other atoms A (i.e. $x \leq y$ or foreground atoms) we set $A^\flat := A$; we lift $\cdot^\flat$ in the obvious way to clauses[16] and define $\Gamma^\flat := \{C^\flat \mid C \in \Gamma\}$. Note that $\Gamma^\flat$ is a set of HoCHCs for $\Sigma^\flat$ and $(\Sigma')^\flat$, and that $\mathfrak{A}^\flat$ is finite.

Clearly, there is an inverse $\cdot^\sharp$ of $\cdot^\flat$ on formulas, e.g. satisfying $(x \leq c_M)^\sharp = (x \leq M)$.

Now, suppose $\mathcal{A} \in \mathfrak{A}$. Then valuations α over (a frame induced by) $\mathcal{A}^\flat[\![\iota]\!]$ naturally correspond to valuations $\alpha^\sharp$ over $\mathcal{A}[\![\iota]\!]$ by evaluating ground terms[17] and it holds $\mathcal{A}[\![\varphi]\!](\alpha^\sharp) = \mathcal{A}^\flat[\![\varphi^\flat]\!](\alpha)$ for simple background atoms φ.

Conversely, for valuations α and $\alpha^\flat$ (over $\mathcal{A}[\![\iota]\!]$ and $\mathcal{A}^\flat[\![\iota]\!]$, respectively) satisfying

$$\alpha^\flat(x) = \begin{cases} \arg\max_{M \in \mathrm{gt}_\iota(\Gamma)} \mathcal{A}[\![M]\!] \\ \qquad\qquad \text{if } \{M \in \mathrm{gt}_\iota(\Gamma) \mid \mathcal{A}[\![M]\!] \geq \alpha(x)\} = \emptyset \\ \arg\min_{M \in \mathrm{gt}_\iota(\Gamma) \wedge \mathcal{A}[\![M]\!] \geq \alpha(x)} \mathcal{A}[\![M]\!] \qquad \text{otherwise} \end{cases}$$

for $x : \iota \in \Delta$, it holds $\mathcal{A}[\![\varphi]\!](\alpha) \leq \mathcal{A}^\flat[\![\varphi^\flat]\!](\alpha^\flat)$ if $\mathrm{gt}_\iota(\varphi) \subseteq \mathrm{gt}_\iota(\Gamma)$. Therefore:

Lemma 52. *Let* Γ' *be a set of goal clauses of simple background atoms satisfying* $\mathrm{gt}_\iota(\Gamma') \subseteq \mathrm{gt}_\iota(\Gamma)$.

Then Γ' *is* $\mathfrak{A}$*-satisfiable iff* $(\Gamma')^\flat$ *is* $\mathfrak{A}^\flat$*-satisfiable.*

Lemma 53. *Let* Γ' *be a set of HoBHC(SLA) satisfying* $\mathrm{gt}_\iota(\Gamma') \subseteq \mathrm{gt}_\iota(\Gamma)$. *Then*

(i) $\Gamma' \vdash_{\mathfrak{A}} \Gamma' \cup \{G\}$ *implies* $(\Gamma')^\flat \vdash_{\mathfrak{A}^\flat} (\Gamma')^\flat \cup \{G^\flat\}$
(ii) $(\Gamma')^\flat \vdash_{\mathfrak{A}^\flat} (\Gamma')^\flat \cup \{G\}$ *implies* $\Gamma' \vdash_{\mathfrak{A}} \Gamma' \cup \{G^\sharp\}$.

The proof of the Completeness Thm. 43 can be strengthened [15] to yield:

Proposition 54. *If* Γ *is* $\mathfrak{A}$*-unsatisfiable then* $\Gamma \vdash^*_{\mathfrak{A}} \Gamma' \cup \{\bot\}$ *for some* Γ'.

Consequently, if Γ is $\mathfrak{A}$-unsatisfiable then $\Gamma \vdash^*_{\mathfrak{A}} \Gamma' \cup \{\bot\}$ for some Γ'. It is easy to see that sets Γ' of HoBHC(SLA) satisfying $\mathrm{gt}_\iota(\Gamma') \subseteq \mathrm{gt}_\iota(\Gamma)$ are closed under the rules of the proof system. Hence, by Lemma 53(i), $\Gamma^\flat \vdash^*_{\mathfrak{A}^\flat} (\Gamma')^\flat \cup \{\bot\}$ and therefore by soundness (Prop. 25), $\Gamma^\flat$ is $\mathfrak{A}^\flat$-unsatisfiable.

The converse can be derived similarly and we conclude:

Proposition 55. Γ *is* $\mathfrak{A}$*-satisfiable iff* $\Gamma^\flat$ *is* $\mathfrak{A}^\flat$*-satisfiable.*

Finally, $\mathfrak{A}^\flat$, which is finite, can be effectively obtained as a result of the decidability of Linear Integer Arithmetic (or *Presburger arithmetic*) [33]. Moreover, for every $\mathcal{A}^\flat \in \mathfrak{A}^\flat$ and type σ, $\mathcal{A}^\flat[\![\sigma]\!]$ is finite. Hence, by Remark 49, we obtain:

Theorem 56. *Let* Γ *be a finite set of HoBHC(SLA). It is decidable if there is a* Σ'*-expansion* $\mathcal{B}$ *of* $\mathcal{A}_{\mathrm{LIA}}$ *satisfying* $\mathcal{B} \models \Gamma$.

IX. Related Work

a) Higher-order Automated Theorem Proving: There is a long history of resolution-based procedures for higher-order logic *without* background theories which are refutationally complete for Henkin semantics e.g. [11-14]. Furthermore, a tableau-style proof system has been proposed [34]. Their completeness proofs construct *countable* Henkin models out of terms in case the proof system is unable to refute a problem. Hence, these proofs do not seem to be extendable to provide standard models when restricted to HoCHCs.

Furthermore, there are efforts to extend SMT solvers to higher-order logic [35], [36] but the techniques seem to be incomplete even for Henkin semantics.

b) Theorem Proving for 1st-order Logic Modulo Theories: In the 1990s, superposition [37]—the basis of most state-of-the-art theorem provers [38], [39]—was extended to a setting with background theories [20], [40]. The proof system is sound and complete, assuming a compact background theory and some technical conditions. Abstractly, their proof system is very similar to ours: there is a clear separation between logical / foreground reasoning and reasoning in the background theory. Moreover, the search is directed purely by the former whilst the latter is only used in a final step to check satisfiability of a conjunction of theory atoms.

c) Defunctionalisation: Our translation to 1st-order logic (Sec. VII) resembles Reynolds' *defunctionalisation* [41]. A whole-program transformation, defunctionalisation reduces higher-order functional programs to 1st-order ones. It eliminates higher-order features, such as partial applications and λ-abstractions, by storing arguments in data types and recovering them in an application function, which performs a matching on the data type.

Recently, the approach was adapted to the satisfiability problem for HoCHC [42] and implemented in the tool *DefMono*[18]: given a set of HoCHCs, it generates an equi-satisfiable set of 1st-order Horn clauses over the original background theory and additionally the theory of data types. By contrast, our translation is purely logical, directly yielding 1st-order Horn clauses, without recourse to inductive data types.

d) Extensional Higher-order Logic Programming: The aim of higher-order logic programming is not only to establish satisfiability of a set of Horn clauses without background theories but also to find (representatives of) "answers to queries", i.e. witnesses that goal clauses are falsified in every model of the definite clauses. Thus, [17] proposes a rather complicated domain-theoretic semantics (equivalent to the continuous semantics [17, Prop. 5.14]). They design a resolution-based proof system that supports a strong notion of completeness ([17, Thm. 7.38]) with respect to this semantics.

Their proof system is more complicated because it operates on more general formulas (which are nonetheless translatable to clauses). Moreover it requires the instantiation of variables

[16] i.e. $(\neg A_1 \vee \cdots \vee \neg A_n \vee (\neg)A)^\flat := \neg A_1^\flat \vee \cdots \vee \neg A_n^\flat \vee (\neg)A^\flat$

[17] precisely, $\alpha^\sharp(x) = \mathcal{A}[\![\alpha(x)]\!]$ for $x : \iota \in \Delta$

[18] see http://mjolnir.cs.ox.ac.uk/dfhochc/

with certain terms, which we avoid by implicitly instantiating all remaining relational variables with $\top_\rho$ in the constraint refutation rule.

e) Refinement Type Assignments: [1] also introduces a refinement type system, the aim of which is to automate the search for models. In this respect, the approach is orthogonal to our resolution proof system, which can be used to refute all unsatisfiable problems (but might fail on satisfiable instances). However, for satisfiable clause sets the method by [1], which is implemented in the tool *Horus*[19], may also be unable to generate models.

X. Conclusion and Future Directions

In sum, HoCHC lies at a "sweet spot" in higher-order logic, semantically robust and useful for algorithmic verification.

Future work: We expect that our proof system's robustness on *satisfiable* instances can be improved by tightening the rules (cf. Sec. IX) or combining it with a search for models [1], [43], [44]. Crucially, soundness and completeness even for standard semantics can be retained thanks to HoCHC's semantic invariance. To facilitate comparison of approaches, it would also be important to obtain an implementation of our techniques and conduct an empirical evaluation.

On the more theoretical side it would be interesting to identify extensions of HoCHC sharing its excellent properties.

Acknowledgments: We gratefully acknowledge support of EPSRC grants EP/N509711/1 and EP/M023974/1.

References

[1] T. Cathcart Burn, C.-H. L. Ong, and S. J. Ramsay, "Higher-order constrained Horn clauses for verification," *PACMPL*, vol. 2, no. POPL, pp. 11:1–11:28, 2018.

[2] M. Davis and H. Putnam, "A computing procedure for quantification theory," *J. ACM*, vol. 7, no. 3, pp. 201–215, 1960.

[3] J. A. Robinson, "A machine-oriented logic based on the resolution principle," *J. ACM*, vol. 12, no. 1, pp. 23–41, 1965.

[4] H. B. Enderton, *A Mathematical Introduction to Logic*, 2nd ed. Academic Press, 2001.

[5] L. Henkin, "Completeness in the theory of types," *J. Symb. Log.*, vol. 15, no. 2, pp. 81–91, 1950.

[6] J. V. Benthem and K. Doets, "Higher-order logic," in *Handbook of Philosophical Logic*, ser. Synthese Library (Studies in Epistemology, Logic, Methodology, and Philosophy of Science), D. M. Gabbay and F. Guenthner, Eds. Springer, Dordrecht, 1983, vol. 164.

[7] D. Leivant, "Higher order logic," in *Handbook of Logic in Artificial Intelligence and Logic Programming*, D. M. Gabbay, C. J. Hogger, and J. A. Robinson, Eds. New York, NY, USA: Oxford University Press, Inc., 1994, pp. 229–321.

[8] M. J. C. Gordon and T. F. Melham, *Introduction to HOL: A theorem proving environment for higher order logic*. Cambridge University Press, 1993.

[9] M. J. C. Gordon and A. M. Pitts, "The HOL logic and system," in *Towards Verified Systems*, J. Bowen, Ed. Elsevier, 1994, pp. 49–70.

[10] R. Kashima, "A proof of the standardization theorem in lambda-calculus," Tokyo Institute of Technology, Research Reports on Mathematical and Computing Sciences C-145, 2000.

[11] P. B. Andrews, "Resolution in type theory," *Journal of Symbolic Logic*, vol. 36, no. 3, pp. 414–432, 1971.

[12] G. P. Huet, "Constrained resolution: A complete method for higher-order logic." Ph.D. dissertation, Case Western Reserve University, Cleveland, OH, USA, 1972.

[13] C. Benzmüller and M. Kohlhase, "Extensional higher-order resolution," in *Automated Deduction - CADE-15, 15th International Conference on Automated Deduction, Lindau, Germany, July 5-10, 1998, Proceedings*, 1998, pp. 56–71.

[14] A. Bentkamp, J. C. Blanchette, S. Cruanes, and U. Waldmann, "Superposition for lambda-free higher-order logic," in *Automated Reasoning - 9th International Joint Conference, IJCAR 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings*, 2018, pp. 28–46.

[15] C. L. Ong and D. Wagner, "HoCHC: A refutationally complete and semantically invariant system of higher-order logic modulo theories," *CoRR*, vol. abs/1902.10396, 2019.

[16] H. P. Barendregt, *The lambda calculus, its syntax and semantics*, ser. Studies in Logic (London). College Publications, London, 2012, vol. 40, [Reprint of the 1984 revised edition, MR0774952], With addenda for the 6th imprinting, Mathematical Logic and Foundations.

[17] A. Charalambidis, K. Handjopoulos, P. Rondogiannis, and W. W. Wadge, "Extensional higher-order logic programming," *ACM Trans. Comput. Log.*, vol. 14, no. 3, pp. 21:1–21:40, 2013.

[18] S. Abramsky and A. Jung, "Domain theory," in *Handbook of Logic in Computer Science*, S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, Eds. New York, NY, USA: Oxford University Press, Inc., 1994, vol. 3, pp. 1–168.

[19] C. Chang and H. Keisler, *Model Theory*, 3rd ed., ser. Dover Books on Mathematics. New York, NY, USA: Dover Publications Inc., 2013.

[20] L. Bachmair, H. Ganzinger, and U. Waldmann, "Refutational theorem proving for hierarchic first-order theories," *Appl. Algebra Eng. Commun. Comput.*, vol. 5, pp. 193–212, 1994.

[21] N. Bjørner, A. Gurfinkel, K. L. McMillan, and A. Rybalchenko, "Horn clause solvers for program verification," in *Fields of Logic and Computation II - Essays Dedicated to Yuri Gurevich on the Occasion of His 75th Birthday*, 2015, pp. 24–51.

[22] J. Van Benthem and K. Doets, *Higher-Order Logic*. Dordrecht: Springer Netherlands, 1983, pp. 275–329.

[23] M. Kerber, "How to prove higher order theorems in first order logic," in *Proceedings of the 12th International Joint Conference on Artificial Intelligence. Sydney, Australia, August 24-30, 1991*, 1991, pp. 137–142.

[24] J. C. Blanchette, C. Kaliszyk, L. C. Paulson, and J. Urban, "Hammering towards QED," *J. Formalized Reasoning*, vol. 9, no. 1, pp. 101–148, 2016.

[25] T. Johnsson, "Lambda lifting: Transforming programs to recursive equations," in *Functional Programming Languages and Computer Architecture, FPCA 1985, Nancy, France, September 16-19, 1985, Proceedings*, 1985, pp. 190–203.

[26] D. A. Plaisted and S. Greenbaum, "A structure-preserving clause form translation," *J. Symb. Comput.*, vol. 2, no. 3, pp. 293–304, 1986.

[27] A. Nonnengart and C. Weidenbach, "Computing small clause normal forms," in *Handbook of Automated Reasoning (in 2 volumes)*, 2001, pp. 335–367.

[28] P. J. Downey, "Undecidability of presburger arithmetic with a single monadic predicate letter," Center for Research in Computer Technology, Harvard University, Technical Report TR-18-72, 1972.

[29] M. Horbach, M. Voigt, and C. Weidenbach, "The universal fragment of Presburger arithmetic with unary uninterpreted predicates is undecidable," *CoRR*, vol. abs/1703.01212, 2017.

[30] Z. Manna, *Mathematical Theory of Computation*. New York, NY, USA: Dover Publications Inc., 2003.

[31] Y. Ge and L. M. de Moura, "Complete instantiation for quantified formulas in satisfiabiliby modulo theories," in *Computer Aided Verification, 21st International Conference, CAV 2009, Grenoble, France, June 26 - July 2, 2009. Proceedings*, 2009, pp. 306–320.

[32] M. Horbach, M. Voigt, and C. Weidenbach, "On the combination of the Bernays-Schönfinkel-Ramsey fragment with simple linear integer arithmetic," in *Automated Deduction - CADE 26 - 26th International Conference on Automated Deduction, Gothenburg, Sweden, August 6-11, 2017, Proceedings*, 2017, pp. 77–94.

[33] M. Presburger, "über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt," in *Comptes Rendus du I Congrès de Mathématiciens des Pays Slaves*, 1929, p. 92–101.

[34] C. E. Brown, "Reducing higher-order theorem proving to a sequence of SAT problems," in *Automated Deduction - CADE-23 - 23rd International Conference on Automated Deduction, Wroclaw, Poland, July 31 - August 5, 2011. Proceedings*, 2011, pp. 147–161.

[19] see http://mjolnir.cs.ox.ac.uk/horus/

[35] H. Barbosa, J. C. Blanchette, S. Cruanes, D. E. Ouraoui, and P. Fontaine, "Language and proofs for higher-order SMT (work in progress)," in *Proceedings of the Fifth Workshop on Proof eXchange for Theorem Proving, PxTP 2017, Brasília, Brazil, 23-24 September 2017.*, 2017, pp. 15–22.

[36] H. Barbosa, A. Reynolds, D. E. Ouraoui, C. Tinelli, and C. W. Barrett, "Extending SMT solvers to higher-order logic," Technical Report, 2019.

[37] L. Bachmair and H. Ganzinger, "On restrictions of ordered paramodulation with simplification," in *10th International Conference on Automated Deduction, Kaiserslautern, FRG, July 24-27, 1990, Proceedings*, 1990, pp. 427–441.

[38] C. Weidenbach, D. Dimova, A. Fietzke, R. Kumar, M. Suda, and P. Wischnewski, "SPASS version 3.5," in *Automated Deduction - CADE-22, 22nd International Conference on Automated Deduction, Montreal, Canada, August 2-7, 2009. Proceedings*, 2009, pp. 140–145.

[39] L. Kovács and A. Voronkov, "First-order theorem proving and vampire," in *Computer Aided Verification - 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings*, 2013, pp. 1–35.

[40] E. Althaus, E. Kruglov, and C. Weidenbach, "Superposition modulo linear arithmetic SUP(LA)," in *Frontiers of Combining Systems, 7th International Symposium, FroCoS 2009, Trento, Italy, September 16-18, 2009. Proceedings*, 2009, pp. 84–99.

[41] J. C. Reynolds, "Definitional interpreters for higher-order programming languages," in *Proceedings of the ACM annual conference-Volume 2*. ACM, 1972, pp. 717–740.

[42] L. Pham, S. J. Ramsay, and C.-H. L. Ong, "Defunctionalization of higher-order constrained Horn clauses," *CoRR*, vol. abs/1810.03598, 2018.

[43] H. Unno, T. Terauchi, and N. Kobayashi, "Automating relatively complete verification of higher-order functional programs," in *The 40th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL '13, Rome, Italy - January 23 - 25, 2013*, 2013, pp. 75–86.

[44] N. Kobayashi, R. Sato, and H. Unno, "Predicate abstraction and CEGAR for higher-order model checking," in *Proceedings of the 32nd ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI 2011, San Jose, CA, USA, June 4-8, 2011*, 2011, pp. 222–233.