Deductive Systems for Logic Programs with Counting

Jorge Fandinno and Vladimir Lifschitz

May 7, 2024

Abstract

In answer set programming, two groups of rules are considered strongly equivalent if they have the same meaning in any context. Strong equivalence of two programs can be sometimes established by deriving rules of each program from rules of the other in an appropriate deductive system. This paper shows how to extend this method of proving strong equivalence to programs containing the counting aggregate.

1 Introduction

In answer set programming (ASP), two groups of rules are considered strongly equivalent if, informally speaking, they have the same meaning in any context [Lifschitz *et al.*, 2001]. If programs Π_1 and Π_2 are strongly equivalent then, for any program Π, programs $\Pi_1 \cup \Pi$ and $\Pi_2 \cup \Pi$ have the same stable models. Properties of this equivalence relation are important because they can help us simplify parts of an ASP program without examining its other parts. More generally, they can guide us in the process of developing correct and efficient code.

Strong equivalence of two programs can be sometimes established by deriving rules of each program from rules of the other in an appropriate deductive system. Deriving rules involves rewriting them in the syntax of first-order logic. The possibility of such proofs has been demonstrated for the ASP language mini-GRINGO [Lifschitz *et al.*, 2019, Lifschitz, 2021, Fandinno and Lifschitz, 2023a], and it was used in the design of a proof assistant for verifying strong equivalence [Heuer, 2020, Fandinno and Lifschitz, 2023b].

We are interested in extending this method of proving strong equivalence to ASP programs with aggregates, such as counting and summation [Gebser *et al.*, 2019, Section 3.1.12]. Procedures for representing rules with aggregates in the syntax of first-order logic have been proposed in several recent publications [Lifschitz, 2022, Fandinno *et al.*, 2022, Fandinno and Hansen, 2023]. The first of these papers describes a deductive system that can be used for proving strong equivalence of programs in the language called mini-GRINGO with counting (MGC). But that system is too weak for reasoning about MGC rules that contain variables in the right-hand side of an aggregate atom. For instance, let A be the pair of rules

$$\begin{array}{l} p(a), \\ q(Y) \leftarrow count\{X : p(X) \wedge X \neq a\} = Y, \end{array}$$

and let B stand for

$$\begin{array}{l} p(a), \\ q(Y-1) \leftarrow count\{X : p(X)\} = Y. \end{array}$$

These pairs of rules are strongly equivalent to each other, but the deductive system mentioned above would not allow us to justify this claim.

We propose here an alternative set of axioms for proving strong equivalence of programs with counting. After reviewing in Section 2 the language MGC and the translation τ^* that transforms MGC rules into first-order sentences, we define in Section 3 a deductive system of *here-and-there with counting (HTC)*. Any two MGC programs Π_1 and Π_2 such that $\tau^*\Pi_1$ and $\tau^*\Pi_2$ can be derived from each other in this deductive system are strongly equivalent. Furthermore, the sentences τ^*A and τ^*B, corresponding to the programs A and B above, are equivalent in HTC, as well as any two sentences that are equivalent in the deductive system from the previous publication on MGC (Sections 4 and 5).

The system HTC is not a first-order theory in the sense of classical logic, because some instances of the law of excluded middle $F \vee \neg F$ are not provable in it. This fact makes it difficult to automate reasoning in HTC, because existing work on automated reasoning deals for the most part with classical logic and its extensions. Lin [2002] showed how to modify the straightforward representation of propositional rules by formulas in such a way that strong equivalence will correspond to equivalence of formulas in classical logic. His method was used in the design of a system for verifying strong equivalence of propositional programs [Chen *et al.*, 2005]. It was also generalized to strong equivalence of propositional formulas [Pearce *et al.*, 2009], first-order formulas [Ferraris *et al.*, 2011], and mini-GRINGO programs [Fandinno and Lifschitz, 2023b], and it was used in the design of a system for verifying strong equivalence in mini-GRINGO [Heuer, 2020]. In Section 6 we show that this method is applicable to programs with counting as well. To this end, we define a classical first-order theory HTC' and an additional syntactic transformation γ such that two sentences F_1, F_2 are equivalent in HTC if and only if γF_1 is equivalent to γF_2 in HTC'. It follows that if the formula $\gamma\tau^*\Pi_1 \leftrightarrow \gamma\tau^*\Pi_2$ can be derived from the axioms of HTC' in classical first-order logic then Π_1 is strongly equivalent to Π_2.

Section 7 describes a modificaton HTC^ω of the deductive system HTC that is not only sound for proving strong equivalence, but also complete: any two MGC programs Π_1, Π_2 are strongly equivalent if and only if the formulas $\tau^*\Pi_1$ and $\tau^*\Pi_2$ are equivalent in the modified system. This is achieved by including rules with infinitely many premises, similar to the ω-rule in arithmetic investigated by Leon Henkin [1954]:

$$\frac{F(0) \quad F(1) \quad \ldots}{\forall n F(n)}.$$

Deductive systems of this kind are useful as theoretical tools. But derivations in such systems are infinite trees, and they cannot be represented in a finite computational device.

Proofs of theorems are presented in the sections that follow. Some of the proofs refer to the concept of an HT-interpretation, which is reviewed in Section 10. In Section 12, we define a class of *standard* HT-interpretations, for which the deductive system HTC^{ω} is sound and complete.

2 Background

2.1 Programs

The syntax of mini-GRINGO with counting is defined as follows.[1] We assume that three countably infinite sets of symbols are selected: *numerals*, *symbolic constants*, and *variables*. We assume that a 1-1 correspondence between numerals and integers is chosen; the numeral corresponding to an integer n is denoted by $\overline{n}$. (In examples of programs, we sometimes drop overlines in numerals.)

Precomputed terms are numerals and symbolic constants. We assume that a total order on the set of precomputed terms is selected such that numerals are contiguous (no symbolic constants between numerals) and are ordered in the standard way. MGC terms are formed from precomputed terms and variables using the unary operation symbol $|\,|$ and the binary operation symbols

$$+ \quad - \quad \times \quad / \quad \backslash \quad ..$$

An MGC *atom* is a symbolic constant optionally followed by a tuple of terms in parentheses. A *literal* is an MGC atom possibly preceded by one or two occurrences of *not*. A *comparison* is an expression of the form $t_1 \prec t_2$, where t_1, t_2 are mini-GRINGO terms, and $\prec$ is $=$ or one of the comparison symbols

$$\neq \quad < \quad > \quad \leq \quad \geq \tag{1}$$

An *aggregate element* is a pair $\mathbf{X} : \mathbf{L}$, where $\mathbf{X}$ is a tuple of distinct variables, and $\mathbf{L}$ is a conjunction of literals and comparisons such that every member of $\mathbf{X}$ occurs in $\mathbf{L}$. An *aggregate atom* is an expression of one of the forms

$$count\{E\} \geq t, \;\; count\{E\} \leq t, \tag{2}$$

where E is an aggregate element, and t is a term that does not contain the interval symbol (..). The conjunction of aggregate atoms (2) can be written as $count\{E\} = t$.

A *rule* is an expression of the form

$$Head \leftarrow Body, \tag{3}$$

where

[1] The description below differs slightly from the original publication [Lifschitz, 2022]: the absolute value symbol $|\,|$ is allowed in the definition of a term, and the symbols *inf* and *sup* are not included.

- *Body* is a conjunction (possibly empty) of literals, comparisons, and aggregate atoms, and
- *Head* is either an atom (then (3) is a *basic rule*), or an atom in braces (then (3) is a *choice rule*), or empty (then (3) is a *constraint*).

A variable that occurs in a rule R is *local* in R if each of its occurrences is within an aggregate element, and *global* otherwise. A rule is *pure* if, for every aggregate element $\mathbf{X} : \mathbf{L}$ in its body, all variables in the tuple $\mathbf{X}$ are local. For example, the rule

$$q(Y) \leftarrow count\{X : p(X)\} = Y \wedge X > 0$$

is not pure, because X is global.

In mini-GRINGO with counting, a *program* is a finite set of pure rules.

2.2 Stable models and strong equivalence

An atom $p(\mathbf{t})$ is *precomputed* if all members of the tuple $\mathbf{t}$ are precomputed terms. The semantics of MGC is based on an operator, called τ, which transforms pure rules into infinitary propositional formulas formed from precomputed atoms [Lifschitz, 2022, Section 5]. For example, the rule

$$q \leftarrow count\{X : p(X)\} \leq 5$$

is transformed by τ into the formula

$$\left(\bigwedge_{\Delta\,:\,|\Delta|>5} \neg \bigwedge_{x \in \Delta} p(x) \right) \rightarrow q,$$

where Δ ranges over finite sets of precomputed terms, and $|\Delta|$ stands for the cardinality of Δ. The result of applying τ to a program Π is defined as the conjunction of formulas τR for all rules R of Π.

Stable models of an MGC program Π are defined as stable models of $\tau\Pi$ in the sense of Truszczynski [2012]. Thus stable models of programs are sets of precomputed atoms.

About programs Π_1 and Π_2 we say that they are *strongly equivalent* to each other if $\tau\Pi_1$ is strongly equivalent to $\tau\Pi_2$; in other words, if for every set Φ of infinitary propositional formulas formed from precomputed atoms, $\{\tau\Pi_1\} \cup \Phi$ and $\{\tau\Pi_2\} \cup \Phi$ have the same stable models. It is clear that if Π_1 is strongly equivalent to Π_2 then, for any program Π, $\Pi_1 \cup \Pi$ has the same stable models as $\Pi_2 \cup \Pi$ (take Φ to be $\{\tau\Pi\}$).

2.3 Representing MGC terms and atoms by formulas

In first-order formulas used to represent programs, we distinguish between terms of two sorts: the sort *general* and its subsort *integer*. General variables are meant to range over arbitrary precomputed terms, and we assume them to be the same as variables used in MGC programs. Integer variables are meant to range over numerals (or, equivalently, integers). In this paper, integer variables are represented by letters from the middle of the alphabet $(I, \ldots, N)$.

The two-sorted signature σ_0 includes

- all numerals as object constants of the sort *integer*;
- all symbolic constants as object constants of the sort *general*;
- the symbol $|\,|$ as a unary function constant; its argument and value have the sort *integer*;
- the symbols $+$, $-$ and $\times$ as binary function constants; their arguments and values have the sort *integer*;
- pairs p/n, where p is a symbolic constant and n is a nonnegative integer, as n-ary predicate constants; their arguments have the sort *general*;
- symbols (1) as binary predicate constants; their arguments have the sort *general*.

Note that the definition of σ_0 does not allow terms that contain a general variable in the scope of an arithmetic operation. For example, the MGC term $Y - \overline{1}$ is not a term over σ_0.

A formula of the form $(p/n)(\mathbf{t})$, where $\mathbf{t}$ is a tuple of terms, can be written also as $p(\mathbf{t})$. Thus precomputed atoms can be viewed as sentences over σ_0.

The set of values of an MGC term[2] t can be described by a formula over the signature σ_0 that contains a variable Z that does not occur in t [Lifschitz *et al.*, 2019, Fandinno and Lifschitz, 2023a]. This formula, "Z is a value of t," is denoted by $val_t(Z)$. Its definition is recursive, and we reproduce here two of its clauses:

- if t is a precomputed term or a variable then $val_t(Z)$ is $Z = t$,
- if t is $t_1 \; op \; t_2$, where op is $+$, $-$, or $\times$ then $val_t(Z)$ is

$$\exists IJ(val_{t_1}(I) \wedge val_{t_2}(J) \wedge Z = I \; op \; J),$$

  where I and J are integer variables that do not occur in t.

For example, $val_{Y-\overline{1}}(Z)$ is

$$\exists IJ(I = Y \wedge J = \overline{1} \wedge Z = I - J).$$

The translation τ^B transforms MGC atoms, literals and comparisons into formulas over the signature σ_0. (The superscript B conveys the idea that

[2]We talk about a *set* of values because an MGC term may contain the interval symbol. For instance, the values of the MGC term $\overline{1}..\overline{3}$ are $\overline{1}$, $\overline{2}$, and $\overline{3}$. On the other hand, the set of values of the term $a - \overline{1}$, where a is a symbolic constant, is empty.

this translation expresses the meaning of expressions in *bodies* of rules.) For example, τ^B transforms $p(t)$ into the formula $\exists Z(val_t(Z) \wedge p(Z))$. The complete definition of τ^B can be found in earlier publications [Lifschitz *et al.*, 2019, Fandinno and Lifschitz, 2023a].

2.4 Representing aggregate expressions and rules

To represent aggregate expressions by first-order formulas, we need to extend the signature σ_0 [Lifschitz, 2022, Section 7]. The signature σ_1 is obtained from σ_0 by adding all predicate constants of the forms

$$Atleast_F^{\mathbf{X};\mathbf{V}} \text{ and } Atmost_F^{\mathbf{X};\mathbf{V}} \tag{4}$$

where $\mathbf{X}$ and $\mathbf{V}$ are disjoint lists of distinct general variables, and F is a formula over σ_0 such that each of its free variables belongs to $\mathbf{X}$ or to $\mathbf{V}$. The number of arguments of each of constants (4) is greater by 1 than the length of $\mathbf{V}$; all arguments are of the sort *general*. If n is a positive integer then the formula $Atleast_F^{\mathbf{X},\mathbf{V}}(\mathbf{V}, \overline{n})$ is meant to express that F holds for at least n values of $\mathbf{X}$. The intuitive meaning of $Atmost_F^{\mathbf{X},\mathbf{V}}(\mathbf{V}, \overline{n})$ is similar: F holds for at most n values of $\mathbf{X}$.

For an aggregate atom of the form $count\{\mathbf{X} : \mathbf{L}\} \geq t$ in the body of a rule, the corresponding formula over σ_1 is

$$\exists Z \left(val_t(Z) \wedge Atleast^{\mathbf{X};\mathbf{V}}_{\exists \mathbf{W} \tau^B(\mathbf{L})}(\mathbf{V}, Z) \right),$$

where

- $\mathbf{V}$ is the list of global variables that occur in $\mathbf{L}$, and
- $\mathbf{W}$ is the list of local variables that occur in $\mathbf{L}$ and are different from the members of $\mathbf{X}$.

For example, the aggregate atom $count\{X : p(X)\} \geq Y$ is represented by the formula

$$\exists Z \left(Z = Y \wedge Atleast^{X;}_{\exists Z(Z=X \wedge p(Z))}(Z) \right)$$

($\mathbf{V}$ and $\mathbf{W}$ are empty).

The formula representing $count\{\mathbf{X} : \mathbf{L}\} \leq t$ is formed in a similar way, with *Atmost* in place of *Atleast*.

Now we are ready to define the translation τ^*, which transforms pure rules into sentences over σ_1. It converts a basic rule

$$p(t) \leftarrow B_1 \wedge \cdots \wedge B_n$$

into the universal closure of the formula

$$B_1^* \wedge \cdots \wedge B_n^* \wedge val_t(Z) \rightarrow p(Z),$$

where B_i^* is

- $\tau^B(B_i)$, if B_i is a literal or comparison, and
- the formula representation of B_i formed as described above, if B_i is an aggregate atom.

The definition of τ^* for pure rules of other forms can be found in the previous paper on mini-GRINGO with counting [Lifschitz, 2022, Sections 6 and 8]. For any program Π, $\tau^*\Pi$ stands for the conjunction of the sentences τ^*R for all rules R of Π.

2.5 Logic of here-and-there and standard interpretations

We are interested in deductive systems S with the following property:

$$\begin{array}{c}\textit{for any programs } \Pi_1 \textit{ and } \Pi_2,\\ \textit{if } \tau^*\Pi_1 \textit{ and } \tau^*\Pi_2 \textit{ can be derived from each other in } S\\ \textit{then } \Pi_1 \textit{ is strongly equivalent to } \Pi_2.\end{array} \tag{5}$$

Systems with property (5) cannot possibly sanction unlimited use of classical propositional logic. Consider, for instance, the one-rule programs

$$p \leftarrow not\ q \quad \text{and} \quad q \leftarrow not\ p.$$

They have different stable models, although the corresponding formulas

$$\neg q \rightarrow p \quad \text{and} \quad \neg p \rightarrow q$$

have the same truth table.

This observation suggests that the study of subsystems of classical logic may be relevant. One such subsystem is first-order intuitionistic logic (with equality) adapted to the two-sorted signature σ_1. Intuitionistic logic does have property (5). Furthermore, this property is preserved if we extend it by the axiom schema

$$F \vee (F \rightarrow G) \vee \neg G, \tag{6}$$

introduced by Hosoi [1966] as part of his formalization of the propositional logic known as the logic of here-and-there.

The axiom schema

$$\exists X(F \rightarrow \forall X\, F) \tag{7}$$

(for a variable X of either sort) can be included without losing property (5) as well. It was introduced to extend the logic of here-and-there to a language with variables and quantifiers [Lifschitz *et al.*, 2007]. Both (6) and (7) are provable classically, but not intuitionistically.

The axioms and inference rules discussed so far are abstract, in the sense that they are not related to any properties of the domains of variables (except that one is a subset of the other). To describe more specific axioms, we need the following definition. An interpretation of the signature σ_0 is *standard* if

- its domain of the sort *general* is the set of precomputed terms;

- its domain of the sort *integer* is the set of numerals;
- every object constant represents itself;
- the absolute value symbol and the binary function constants are interpreted as usual in arithmetic;
- predicate constants (1) are interpreted in accordance with the total order on precomputed terms chosen in the definition of MGC (Section 2.1).

Two standard interpretations of σ_0 can differ only by how they interpret the predicate symbols p/n. If a sentence over σ_0 does not contain these symbols then it is either satisfied by all standard interpretations or is not satisfied by any of them.

Let *Std* be the set of sentences over σ_0 that do not contain predicate symbols of the form p/n and are satisfied by standard interpretations. Property (5) will be preserved if we add any members of *Std* to the set of axioms. The set *Std* includes, for instance, the law of excluded middle $F \vee \neg F$ for every formula F over σ_0 that does not contain symbols p/n. Other examples of formulas from *Std* are

$$\overline{2} \times \overline{2} = \overline{4}, \quad \forall N(N * N \geq \overline{0}), \quad t_1 \neq t_2,$$

where t_1, t_2 are distinct precomputed terms.

To reason about MGC programs, we need also axioms for *Atleast* and *Atmost*. A possible choice of such additional axioms is described in the next section.

3 Deductive system *HTC*

The deductive system *HTC* ("here-and-there with counting") operates with formulas of the signature σ_2, which is obtained from σ_1 (Section 2.4) by adding the predicate constants $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}$, where $\mathbf{X}$ and $\mathbf{V}$ are disjoint lists of distinct general variables, and F is a formula over σ_0 such that each of its free variables belongs to $\mathbf{X}$ or to $\mathbf{V}$. The number of arguments of each of these constants is the combined length of $\mathbf{X}$ and $\mathbf{V}$ plus 1. The last argument is of the sort *integer*, and the other arguments are of the sort *general*.

For any integer n, the formula $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, \overline{n})$ is meant to express that if $n > 0$ then there exists a lexicographically increasing sequence $\mathbf{X}_1, \dots \mathbf{X}_n$ of values satisfying F such that the first of them is $\mathbf{X}$.

3.1 Axioms of *HTC*

The axioms for *Start* define these predicates recursively:

$$\begin{aligned}
&\forall \mathbf{X}\mathbf{V}N(N \leq \overline{0} \rightarrow \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)),\\
&\forall \mathbf{X}\mathbf{V}(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, \overline{1}) \leftrightarrow F),\\
&\forall \mathbf{X}\mathbf{V}N(N > \overline{0} \rightarrow (\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + \overline{1}) \leftrightarrow\\
&\qquad\qquad F \wedge \exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{U}, \mathbf{V}, N)))).
\end{aligned}$$

Here N is an integer variable, and $\mathbf{U}$ is a list of distinct general variables of the same length as $\mathbf{X}$, which is disjoint from both $\mathbf{X}$ and $\mathbf{V}$. The symbol $<$ in the last line denotes lexicographic order: $(X_1, \ldots, X_m) < (U_1, \ldots, U_m)$ stands for

$$\bigvee_{l=1}^{m} \left((X_l < U_l) \wedge \bigwedge_{k=1}^{l-1} (X_k = U_k) \right).$$

This set of axioms for *Start* will be denoted by D_0.

The set of axioms for *Atleast* and *Atmost*, denoted by D_1, defines these predicates in terms of *Start*:

$$\forall \mathbf{V} Y (\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, Y) \leftrightarrow \exists \mathbf{X} N (\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \wedge N \geq Y)), \tag{8}$$

$$\forall \mathbf{V} Y (\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, Y) \leftrightarrow \forall \mathbf{X} N (\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow N \leq Y)). \tag{9}$$

In addition to the axioms listed above, we need the induction schema

$$F_{\overline{0}}^{N} \wedge \forall N \left(N \geq \overline{0} \wedge F \rightarrow F_{N+\overline{1}}^{N} \right) \rightarrow \forall N (N \geq \overline{0} \rightarrow F)$$

for all formulas F over σ_2. The set of the universal closures of its instances will be denoted by *Ind*.

The deductive system *HTC* is defined as first-order intuitionistic logic for the signature σ_2 extended by

- axiom schemas (6) and (7) for all formulas F, G, H over σ_2, and
- axioms *Std*, *Ind*, D_0 and D_1.

This deductive system has property (5):

Theorem 1. *For any programs Π_1 and Π_2, if the formula $\tau^*\Pi_1 \leftrightarrow \tau^*\Pi_2$ is provable in HTC then Π_1 and Π_2 are strongly equivalent.*

As discussed in Section 7, *HTC* is an extension of the system with property (5) introduced by Lifschitz [2022]. Furthermore, in Section 4 we show that *HTC* is sufficiently strong for proving the equivalence between τ^*A and τ^*B for the programs A and B from the introduction.

3.2 Some theorems of *HTC*

The characterization of *Atleast* and *Atmost* given by the axioms D_1 can be simplified, if we replace the variable Y by an integer variable:

Proposition 1. *The formulas*

$$\forall \mathbf{V} N (\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \leftrightarrow \exists \mathbf{X}\, \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)) \tag{10}$$

and

$$\forall \mathbf{V} N (\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \leftrightarrow \neg \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N + \overline{1})) \tag{11}$$

are provable in HTC.

A few other theorems of *HTC*:

Proposition 2. *The formulas*

$$\forall \mathbf{V} N(N \leq \overline{0} \rightarrow \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N)), \tag{12}$$

$$\forall \mathbf{V}(\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{1}) \leftrightarrow \exists \mathbf{X}\, F), \tag{13}$$

$$\forall \mathbf{X}(F \rightarrow G) \rightarrow \forall \mathbf{X} \mathbf{V} N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow \mathit{Start}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)), \tag{14}$$

$$\forall \mathbf{Z} \mathbf{V} N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{Z}, \mathbf{V}, N) \wedge N > \overline{0} \rightarrow F_{\mathbf{Z}}^{\mathbf{X}}) \tag{15}$$

are provable in HTC.

An expression of the form $\mathit{Exactly}_F^{\mathbf{X};\mathbf{V}}(\mathbf{t}, t)$ is shorthand for the conjunction

$$\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{t}, t) \wedge \mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{t}, t)$$

($\mathbf{t}$ is a tuple of terms, and t is a term). By (11), $\mathit{Exactly}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, N)$ is equivalent in *HTC* to

$$\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, N) \wedge \neg \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, N + \overline{1}).$$

Proposition 3. *The formulas*

$$\forall \mathbf{X} Y(\mathit{Exactly}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y) \rightarrow \exists N(Y = N \wedge N \geq \overline{0})) \tag{16}$$

and

$$\forall \mathbf{X}(F \leftrightarrow G) \rightarrow \forall \mathbf{X} Y(\mathit{Exactly}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y) \leftrightarrow \mathit{Exactly}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)) \tag{17}$$

are provable in HTC.

4 An example of reasoning about programs

In this section we show that $\tau^* A$ is equivalent to $\tau^* B$, for the programs A and B from the introduction, in the logic of here-and-there with counting, defined in the previous section. The proof consists of three parts.

4.1 Part 1: Simplification

The translation τ^* transforms program A into the conjunction of the formulas

$$\forall Z(Z = a \rightarrow p(Z)) \tag{18}$$

and

$$\begin{aligned} \forall Y Z(\exists Z_1(Z_1 = Y \wedge \mathit{Atleast}_F^{X;}(Z_1)) \wedge \\ \exists Z_2(Z_2 = Y \wedge \mathit{Atmost}_F^{X;}(Z_2)) \wedge Z = Y \\ \rightarrow q(Z)), \end{aligned} \tag{19}$$

where F stands for $\tau^B(p(a) \wedge X \neq a)$. Formula (18) is equivalent to $p(a)$, and (19) is equivalent to

$$\forall Y(\mathit{Atleast}_F^{X;}(Y) \wedge \mathit{Atmost}_F^{X;}(Y) \rightarrow q(Y)). \tag{20}$$

The antecedent of this implication can be written as $\mathit{Exactly}^{X;}_{F}(Y)$. By (16), it follows that the variable Y can be replaced by the integer variable N. Furthermore, by (17), formula (20) can be further rewritten as

$$\forall N(\mathit{Exactly}^{X;}_{p(a)\wedge X\neq a}(N) \to q(N)), \tag{21}$$

because F is equivalent to $p(a) \wedge X \neq a$.

The result of applying τ^* to B is the conjunction of (18) and

$$\begin{aligned}\forall YZ(\exists Z_1(Z_1 = Y \wedge \mathit{Atleast}^{X;}_{G}(Z_1)) \wedge \\ \exists Z_2(Z_2 = Y \wedge \mathit{Atmost}^{X;}_{G}(Z_2)) \wedge \\ \exists IJ(I = Y \wedge J = \overline{1} \wedge Z = I + J) \\ \to q(Z)),\end{aligned}$$

where G stands for $\tau^B(p(X))$. This formula can be equivalently rewritten as

$$\forall I(\mathit{Exactly}^{X;}_{G}(I + \overline{1}) \to q(I))$$

and further as

$$\forall I(\mathit{Exactly}^{X;}_{p(X)}(I + \overline{1}) \to q(I)), \tag{22}$$

because G is equivalent to $p(X)$.

Thus the claim that τ^*A is equivalent to τ^*B will be proved if we prove the formula

$$p(a) \to \forall N(\mathit{Exactly}^{X;}_{p(X)\wedge X\neq a}(N + \overline{1}) \leftrightarrow \mathit{Exactly}^{X;}_{p(X)}(N)).$$

It is clearly a consequence of the formula

$$p(a) \to \forall N(\mathit{Atleast}^{X;}_{p(X)\wedge X\neq a}(N + \overline{1}) \leftrightarrow \mathit{Atleast}^{X;}_{p(X)}(N)), \tag{23}$$

which is proved below.

4.2 Part 2: Three lemmas

Three lemmas will be proved in the next section:

$$\forall XN(N > \overline{0} \wedge X > a \wedge \mathit{Start}^{X;}_{p(X)}(X, N) \to \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N)), \tag{24}$$

$$\forall XN(N > \overline{0} \wedge X \neq a \wedge \mathit{Start}^{X;}_{p(X)}(X, N + \overline{1}) \to \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N)), \tag{25}$$

and

$$\begin{aligned}\forall XN(N > \overline{0} \wedge X < a \wedge p(a) \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N) \to \\ \mathit{Start}^{X;}_{p(X)}(X, N + \overline{1})).\end{aligned} \tag{26}$$

Using these lemmas, we will now prove (23). Assume $p(a)$; our goal is to show that

$$\mathit{Atleast}^{X;}_{p(X)\wedge X\neq a}(N + \overline{1}) \leftrightarrow \mathit{Atleast}^{X;}_{p(X)}(N).$$

We consider three cases, according to the axiom

$$\forall N(N < \overline{0} \vee N = \overline{0} \vee N > \overline{0})$$

from *Std* .

If $N < \overline{0}$ then both sides of the equivalence are true by (12). If $N = \overline{0}$ then the right-hand side is true by (12), and the left-hand side follows from $p(a)$ by (13). Assume that $N > \overline{0}$.

Left-to-right: assume $\mathit{Atleast}^{X;}_{p(X)}(N+\overline{1})$. By (10), there exists X such that

$$\mathit{Start}^{X;}_{p(X)}(X, N+\overline{1}). \tag{27}$$

Case 1: $X = a$, so that $\mathit{Start}^{X;}_{p(X)}(a, N+\overline{1})$. By D_0,

$$p(a) \wedge \exists U(a < U \wedge \mathit{Start}^{X;}_{p(X)}(U, N)).$$

Take U such that $a < U$ and $\mathit{Start}^{X;}_{p(X)}(U, N)$. By (24), it follows that

$$\mathit{Start}^{X;}_{p(X)\wedge X\neq a}(U, N).$$

Then $\mathit{Atleast}^{X;}_{p(X)\wedge X\neq a}(N)$ by (10). *Case 2:* $X \neq a$. By (27) and (25),

$$\mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N).$$

By (10), it follows that $\mathit{Atleast}^{X;}_{p(X)\wedge X\neq a}(N)$.

Right-to-left: assume $\mathit{Atleast}^{X;}_{p(X)\wedge X\neq a}(N)$. Then, for some X,

$$\mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N) \tag{28}$$

by (10), and consequently $\mathit{Start}^{X;}_{p(X)}(X, N)$ by (14). *Case 1:* $X > a$. Then

$$p(a) \wedge \exists U(a < U \wedge \mathit{Start}^{X;}_{p(X)}(U, N))$$

(take U to be X). By D_0, we can conclude that $\mathit{Start}^{X;}_{p(X)}(a, N+\overline{1})$. Then $\mathit{Atleast}^{X;}_{p(X)}(N+\overline{1})$ follows by (10). *Case 2:* $X \leq a$. From (28) and (15), $X \neq a$, so that $X < a$. From (28) and (26), $\mathit{Start}^{X;}_{p(X)}(X, N+\overline{1})$; $\mathit{Atleast}^{X;}_{p(X)}(N+\overline{1})$ follows by (10).

4.3 Part 3: Proofs of the lemmas

Proofs of all three lemmas use induction in the form

$$F^N_{\overline{1}} \wedge \forall N\left(N \geq \overline{1} \wedge F \rightarrow F^N_{N+\overline{1}}\right) \rightarrow \forall N(N \geq \overline{1} \rightarrow F), \tag{29}$$

which follows from *Ind* and *Std* .

Proof of (24). We need to show that for all positive N,

$$\forall X(X > a \wedge Start^{X;}_{p(X)}(X,N) \to Start^{X;}_{p(X)\wedge X\neq a}(X,N)). \tag{30}$$

If N is $\overline{1}$ then (30) is equivalent to

$$\forall X(X > a \wedge p(X) \to p(X) \wedge X \neq a)$$

by D_0; this formula follows from *Std*. Assume (30) for a positive N; we need to prove

$$\forall X(X > a \wedge Start^{X;}_{p(X)}(X,N+\overline{1}) \to Start^{X;}_{p(X)\wedge X\neq a}(X,N+\overline{1})).$$

Assume $X > a \wedge Start^{X;}_{p(X)}(X,N+\overline{1})$. From the second conjunctive term,

$$p(X) \wedge \exists U(X < U \wedge Start^{X;}_{p(X)}(U,N))$$

by D_0. Take U such that $X < U$ and $Start^{X;}_{p(X)}(U,N)$. Then $U > a$, so that by the induction hypothesis, $Start^{X;}_{p(X)\wedge X\neq a}(U,N)$. Since $p(X)$, $X \neq a$, and $X < U$,

$$Start^{X;}_{p(X)\wedge X\neq a}(X,N+\overline{1}))$$

follows by D_0.

Proof of (25). We need to show that for all positive N,

$$\forall X(X \neq a \wedge Start^{X;}_{p(X)}(X,N+\overline{1}) \to Start^{X;}_{p(X)\wedge X\neq a}(X,N)). \tag{31}$$

To prove this formula for N equal to $\overline{1}$, assume that $X \neq a \wedge Start^{X;}_{p(X)}(X,\overline{2})$. By (15), the second conjunctive term implies $p(X)$; $Start^{X;}_{p(X)\wedge X\neq a}(X,\overline{1})$ follows by D_0. Now assume (31) for a positive N; we need to prove

$$\forall X(X \neq a \wedge Start^{X;}_{p(X)}(X,N+\overline{2}) \to Start^{X;}_{p(X)\wedge X\neq a}(X,N+\overline{1})). \tag{32}$$

Assume $X \neq a \wedge Start^{X;}_{p(X)}(X,N+\overline{2})$. From the second conjunctive term we conclude by D_0 that $p(X)$ and, for some U,

$$U > X \wedge Start^{X;}_{p(X)}(U,N+\overline{1}). \tag{33}$$

Case 1: $U = a$, so that $Start^{X;}_{p(X)}(a,N+\overline{1})$. By D_0, it follows that for some V, $V > a \wedge Start^{X;}_{p(X)}(V,N)$. Then, by (24), $Start^{X;}_{p(X)\wedge X\neq a}(V,N)$. On the other hand, $p(X) \wedge X \neq a$ and $V > a = U > X$; the consequent of (32) follows by D_0.
Case 2: $U \neq a$. By the induction hypothesis, from the second conjunctive term of (33) we conclude that $Start^{X;}_{p(X)\wedge X\neq a}(U,N)$. Since $U > X$ and $p(X) \wedge X \neq a$, the consequent of (32) follows by D_0.

Proof of (26). We need to show that for all positive N,

$$\forall X(X < a \wedge p(a) \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N) \rightarrow \mathit{Start}^{X;}_{p(X)}(X, N + \overline{1})). \qquad (34)$$

To prove this formula for N equal to $\overline{1}$, assume that

$$X < a \wedge p(a) \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, \overline{1}).$$

By D_0, the second conjunctive term implies $\mathit{Start}^{X;}_{p(X)}(a, \overline{1})$, and the third term implies $p(X)$. Hence

$$p(X) \wedge \exists U(X < U \wedge \mathit{Start}^{X;}_{p(X)}(U, \overline{1}))$$

(take U to be a). By D_0, it follows that $\mathit{Start}^{X;}_{p(X)}(X, \overline{2})$. Now assume (34) for a positive N; we need to prove

$$\forall X(X < a \wedge p(a) \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N + \overline{1}) \rightarrow \mathit{Start}^{X;}_{p(X)}(X, N + \overline{2})).$$

Assume $X < a \wedge p(a) \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(X, N + \overline{1})$. From the last conjunctive term we conclude by D_0 that $p(X)$ and there exists U such that

$$X < U \wedge \mathit{Start}^{X;}_{p(X)\wedge X\neq a}(U, N). \qquad (35)$$

From the second conjunctive term of (35), by (15), $p(U)$ and $U \neq a$. *Case 1:* $U < a$. By the induction hypothesis, $\mathit{Start}^{X;}_{p(X)}(U, N + \overline{1})$. Since $p(X)$ and $X < U$, we can conclude by D_0 that $\mathit{Start}^{X;}_{p(X)}(X, N + \overline{2})$. *Case 2:* $U > a$. By (14), the second conjunctive term of (35) implies $\mathit{Start}^{X;}_{p(X)}(U, N)$. Since $p(a)$ and $a < U$, $\mathit{Start}^{X;}_{p(X)}(a, N + \overline{1})$ follows by D_0. Then, since $p(X)$ and $X < a$, $\mathit{Start}^{X;}_{p(X)}(X, N + \overline{2})$ follows in a similar way.

5 Comparison with the original formalization

The deductive system from the previous paper on MGC programs [Lifschitz, 2022] operates with formulas over the signature σ_1 (that is, σ_2 without *Start* predicates). Its definition uses the following notation. If r is a precomputed term, $\mathbf{X}$ is a tuple of distinct general variables, and F is a formula over σ_0, then the expression $\exists_{\geq r}\mathbf{X}F$ stands for

$$\begin{cases} \exists \mathbf{X}_1 \cdots \mathbf{X}_n \left(\bigwedge_{i=1}^{n} F^{\mathbf{X}}_{\mathbf{X}_i} \wedge \bigwedge_{i<j} \neg(\mathbf{X}_i = \mathbf{X}_j) \right) & \text{if } r = \overline{n} > \overline{0}, \\ \top, & \text{if } r \leq \overline{0}, \\ \bot, & \text{if } r > \overline{n} \text{ for all integers } n. \end{cases}$$

Here $\mathbf{X}_1, \ldots, \mathbf{X}_n$ are disjoint tuples of distinct general variables that do not occur in F. The symbols $\top$ and $\bot$ denote the logical constants *true*, *false*. The equality between tuples $X_1, X_2, \ldots$ and $Y_1, Y_2, \ldots$ is understood as the conjunction $X_1 = Y_1 \wedge X_2 = Y_2 \wedge \cdots$. The three cases in this definition cover all precomputed terms r, because the set of numerals is contiguous (Section 2.1). Similarly, $\exists_{\leq r}\mathbf{X}F$ stands for

$$
\begin{array}{ll}
\forall \mathbf{X}_1 \cdots \mathbf{X}_{n+1} \left(\bigwedge_{i=1}^{n+1} F^{\mathbf{X}}_{\mathbf{X}_i} \to \bigvee_{i<j} \mathbf{X}_i = \mathbf{X}_j \right) & \text{if } r = \overline{n} \geq \overline{0}, \\
\bot, & \text{if } r < \overline{0}, \\
\top, & \text{if } r > \overline{n} \text{ for all integers } n.
\end{array}
$$

By *Defs* we denote the set of all sentences of the forms

$$\forall \mathbf{V} \left(\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V}, r) \leftrightarrow \exists_{\geq r}\mathbf{X}F \right) \tag{36}$$

and

$$\forall \mathbf{V} \left(\mathit{Atmost}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V}, r) \leftrightarrow \exists_{\leq r}\mathbf{X}F \right). \tag{37}$$

These formulas are similar to the axioms D_1 of *HTC* (Section 3.1) in the sense that both *Defs* and D_1 can be viewed as definitions of *Atleast* and *Atmost*. But each formula in *Defs* refers to a specific value r of the last argument of *Atleast*, *Atmost*, whereas the last argument of *Atleast*, *Atmost* in D_1 is a variable. This difference explains why *HTC* may be a better tool for proving strong equivalence than deductive systems with the axioms *Defs*.

Theorem 2. *The formulas Defs are provable in HTC.*

The formulas *Defs* are the only axioms of the deductive system from the previous publication [Lifschitz, 2022] that are not included in *HTC*. So the theorem above shows that all formulas provable in that system are provable in *HTC* as well.

6 Deductive system *HTC'*

In this section we show that combining τ^* with an additional syntactic transformation γ allows us to replace *HTC* by a classical first-order theory.

The signature σ_2' is obtained from the signature σ_2 (Section 3.1) by adding, for every predicate symbol p other than comparison symbols (1), a new predicate symbol p' of the same arity. The formula $\forall \mathbf{X}(p(\mathbf{X}) \to p'(\mathbf{X}))$, where $\mathbf{X}$ is a tuple of distinct general variables, is denoted by $\mathcal{A}(p)$. The set of all formulas $\mathcal{A}(p)$ is denoted by $\mathcal{A}$.

For any formula F over the signature σ_2, by F' we denote the formula over σ_2' obtained from F by replacing every occurrence of every predicate symbol p other than comparison symbols by p'. The translation γ, which relates the logic of here-and-there to classical logic, maps formulas over σ_2 to formulas over σ_2'. It is defined recursively:

- $\gamma F = F$ if F is atomic,
- $\gamma(\neg F) = \neg F'$,

- $\gamma(F \wedge G) = \gamma F \wedge \gamma G$,
- $\gamma(F \vee G) = \gamma F \vee \gamma G$,
- $\gamma(F \to G) = (\gamma F \to \gamma G) \wedge (F' \to G')$,
- $\gamma(\forall X\, F) = \forall X\, \gamma F$,
- $\gamma(\exists X\, F) = \exists X\, \gamma F$.

To apply γ to a set of formulas means to apply γ to each of its members.

By HTC' we denote the classical first-order theory over the signature σ_2' with the axioms $\mathcal{A}$, $\gamma(Ind)$, Std, γD_0 and γD_1.

Theorem 3. *A sentence $F \leftrightarrow G$ over the signature σ_2 is provable in HTC iff $\gamma F \leftrightarrow \gamma G$ is provable in HTC'.*

Corollary 1. *A sentence F over the signature σ_2 is provable in HTC iff γF is provable in HTC'.*

Proof. In Theorem 3, take G to be $\top$. $\square$

From Theorems 1 and 3 we conclude that MGC programs Π_1 and Π_2 are strongly equivalent if the formula $\gamma\tau^*\Pi_1 \leftrightarrow \gamma\tau^*\Pi_2$ is provable in HTC'.

7 Deductive system HTC^ω

In case of the language mini-GRINGO, using inference rules with infinitely many premises allows us to define a deductive system that satisfies not only condition (5) but also its converse: programs Π_1, Π_2 are strongly equivalent *if and only if* $\tau^*\Pi_1$ and $\tau^*\Pi_2$ can be derived from each other [Fandinno and Lifschitz, 2023a, Theorem 6]. In this section we define a deductive system with the same property for the language MGC. This system, like the deductive system from the previous publication on MGC [Lifschitz, 2022], does not require extending the signature σ_1.

The system HTC^ω is an extension of first-order intuitionistic logic formalized as the natural deduction system *Int* [Fandinno and Lifschitz, 2023a, Section 5.1] for the signature σ_1. Its derivable objects are *sequents*—expressions $\Gamma \Rightarrow F$, where Γ is a finite set of formulas over σ_1 ("assumptions"), and F is a formula over σ_1. A sequent of the form $\Rightarrow F$ is identified with the formula F. The system HTC^ω is obtained from *Int* by adding

- axiom schemas (6) and (7) for all formulas F, G, H over σ_1,
- axioms *Std* and *Defs*, and
- the *ω-rules*

$$\frac{\Gamma \Rightarrow F^X_t \text{ for all precomputed terms } t}{\Gamma \Rightarrow \forall X\, F},$$

where X is a general variable, and

$$\frac{\Gamma \Rightarrow F_{n}^{N} \text{ for all integers } n}{\Gamma \Rightarrow \forall N\, F},$$

where N is an integer variable.

Induction axioms are not on this list, but the instances of the induction schema *Ind* for all formulas F over σ_1 are provable in HTC^ω. Indeed, we can prove in HTC^ω the sequents

$$F_{\overline{0}}^{N} \wedge \forall N \left(N \geq \overline{0} \wedge F \to F_{N+\overline{1}}^{N}\right) \Rightarrow \overline{n} \geq \overline{0} \to F$$

for all integers n; then *Ind* can be derived by the second ω-rule followed by implication introduction.

Theorem 4. *For any* MGC *programs* Π_1 *and* Π_2*, the formula* $\tau^*\Pi_1 \leftrightarrow \tau^*\Pi_2$ *is provable in* HTC^ω *iff* Π_1 *and* Π_2 *are strongly equivalent.*

The system HTC^ω is not an extension of HTC, because its axioms say nothing about the predicate symbols $Start_F^{\mathbf{X};\mathbf{V}}$. But all theorems of HTC that do not contain these symbols are provable in HTC^ω:

Theorem 5. *Every sentence over the signature* σ_1 *provable in HTC is provable in* HTC^ω.

8 Proofs of Propositions 1–3

8.1 A few more theorems of *HTC*

The symbols $\leq$ and $<$ between tuples refer to lexicographic order, as in Section 3.1.

Claim: If $\mathbf{X}$, $\mathbf{W}$ are disjoint tuples of distinct general variables of the same length, and the variables $\mathbf{W}$ are not free in F, then the formula

$$\forall \mathbf{X}\mathbf{W}\mathbf{V}N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N) \wedge \mathbf{W} \leq \mathbf{X} \wedge F_{\mathbf{W}}^{\mathbf{X}} \to Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{W},\mathbf{V},N)) \quad (38)$$

is provable in HTC.

Proof. By D_0, if $N \leq \overline{0}$ then $Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{W},\mathbf{V},N)$; also, if $N = \overline{1}$ then

$$F_{\mathbf{W}}^{\mathbf{X}} \to Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{W},\mathbf{V},N).$$

It remains to prove

$$N > \overline{0} \wedge Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1}) \wedge \mathbf{W} \leq \mathbf{X} \wedge F_{\mathbf{W}}^{\mathbf{X}} \to Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{W},\mathbf{V},N+\overline{1}). \quad (39)$$

(This assertion is justified by the formula

$$\forall N(N \leq \overline{0} \vee N = \overline{1} \vee \exists M(N = M + \overline{1} \wedge M > \overline{0})),$$

which belongs to *Std*.) Assume the antecedent of (39). From the first two conjunctive terms, by D_0, we can conclude that there exists $\mathbf{U}$ such that

$$\mathbf{X} < \mathbf{U} \wedge \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{U}, \mathbf{V}, N).$$

In combination with the last two conjunctive terms, we get

$$F_{\mathbf{W}}^{\mathbf{X}} \wedge \mathbf{W} < \mathbf{U} \wedge \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{U}, \mathbf{V}, N).$$

Now the consequent of (39) follows by D_0. □

Claim: The formula

$$\forall \mathbf{X}\mathbf{V}N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + \overline{1}) \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)) \tag{40}$$

is provable in *HTC*.

Proof. If $N \leq \overline{0}$ then the consequent of (40) follows from D_0. If $N > \overline{0}$ then, by D_0, the antecedent of (40) implies

$$F \wedge \exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{U}, \mathbf{V}, N)).$$

Thus there exists $\mathbf{U}$ such that $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{U}, \mathbf{V}, N) \wedge \mathbf{X} < \mathbf{U} \wedge F$. The consequent of (40) follows by (38). □

Claim: The formula

$$\forall \mathbf{X}\mathbf{V}MN(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, M) \wedge M \geq N \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)) \tag{41}$$

is provable in *HTC*.

Proof. Since $M \geq N$ is equivalent to $\exists K(K \geq \overline{0} \wedge M = N + K)$, formula (41) can be rewritten as

$$\forall K(K \geq \overline{0} \to \forall \mathbf{X}\mathbf{V}N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + K) \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N))).$$

The proof is by induction *Ind*. The basis

$$\forall \mathbf{X}\mathbf{V}N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + \overline{0}))$$

follows from the *Std* axiom $\forall N(N + \overline{0} = N)$. The induction hypothesis is

$$K \geq \overline{0} \wedge \forall \mathbf{X}\mathbf{V}N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + K) \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N));$$

we need to derive

$$\forall \mathbf{X}\mathbf{V}N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + K + \overline{1}) \to \mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)).$$

Assume $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + K + \overline{1})$. Then, by (40), $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + K)$, and $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N)$ follows by the induction hypothesis. □

8.2 Proof of Proposition 1

Proof of (10). By D_1, the left-hand side of (10) is equivalent to

$$\exists \mathbf{X} M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \wedge M \geq N).$$

From (41) we can conclude that

$$\exists M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \wedge M \geq N)$$

is equivalent to $Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N)$. □

Claim: The formula

$$\forall \mathbf{V} N(Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},N) \leftrightarrow \neg\exists \mathbf{X}\, Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1})) \tag{42}$$

is provable in HTC.

Proof. By D_1, the left-hand side of (42) is equivalent to

$$\forall \mathbf{X} M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \rightarrow M \leq N)$$

and consequently to

$$\neg\exists \mathbf{X} M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \wedge M > N).$$

The formula

$$\exists M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \wedge M > N)$$

is equivalent to

$$\exists M(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},M) \wedge M \geq N+\overline{1})$$

and, by (41), to $Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1})$. □

Formula (11) follows from (10) and (42).

8.3 Proof of Proposition 2

Formulas (12) and (13) follow from (10) and D_0.

Claim: Formula (14) is provable in HTC.

Proof. Assume $\forall \mathbf{X}(F \rightarrow G)$. If $N \leq 0$ then $Start_G^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N)$ by D_0. For positive N, the proof is by induction (29). The basis

$$Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},\overline{1}) \rightarrow Start_G^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},\overline{1})$$

is equivalent to $F \rightarrow G$ by D_0. Take a positive N and assume

$$\forall \mathbf{X}\mathbf{V}(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N) \rightarrow Start_G^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N));$$

we want to show that

$$\forall \mathbf{X}\mathbf{V}(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1}) \to Start_G^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1})). \qquad (43)$$

Assume $Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N+\overline{1})$. By D_0,

$$F \wedge \exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{U},\mathbf{V},N)).$$

Then G and, by the induction hypothesis,

$$\exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge Start_G^{\mathbf{X};\mathbf{V}}(\mathbf{U},\mathbf{V},N)).$$

The consequent of (43) follows by D_0. □

Claim: Formula (15) is provable in HTC.

Proof. Since

$$\forall N(N > \overline{0} \to N = \overline{1} \vee \exists M(N = M + \overline{1} \wedge M > \overline{0}))$$

(Group *Std* axiom), it is sufficient to show that

$$\forall \mathbf{Z}(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{Z},\mathbf{V},\overline{1}) \to F_{\mathbf{Z}}^{\mathbf{X}})$$

and

$$\forall \mathbf{Z}\mathbf{V}N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{Z},\mathbf{V},N+\overline{1}) \wedge N > \overline{0} \to F_{\mathbf{Z}}^{\mathbf{X}}).$$

Both formulas follow from axioms D_0. □

8.4 Proof of Proposition 3

Proof of (16). Assume $Exactly_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},Y)$. Then $Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},Y)$. From D_0 and D_1 we can conclude that

$$\forall \mathbf{V}Y(Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},Y) \to \overline{0} \leq Y).$$

Hence $\overline{0} \leq Y$. On the other hand, $Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},Y)$, so that $\exists N(N \geq Y)$ by D_1. Thus $\exists N(\overline{0} \leq Y \leq N)$. It remains to observe that the formula

$$\forall Y(\exists N(\overline{0} \leq Y \leq N) \to \exists N(Y = N \wedge N \geq \overline{0}))$$

is a group *Std* axiom, because the set of numerals is contiguous. □

Claim: The formula

$$\forall \mathbf{X}(F \to G) \to \forall \mathbf{X}Y(Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},Y) \to Atleast_G^{\mathbf{X};\mathbf{V}}(\mathbf{X},Y)) \qquad (44)$$

is provable in HTC.

Proof. Assume $\forall \mathbf{X}(F \to G)$ and $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)$. By D_1,

$$\exists \mathbf{X} N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \wedge N \geq Y)).$$

Then $\exists \mathbf{X} N(\mathit{Start}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \wedge N \geq Y))$ by (14), and $\mathit{Atleast}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)$ follows by D_1. □

Claim: The formula

$$\forall \mathbf{X}(F \to G) \to \forall \mathbf{X} Y(\mathit{Atmost}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y) \to \mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)) \tag{45}$$

is provable in *HTC*.

Proof. Assume $\forall \mathbf{X}(F \to G)$ and $\mathit{Atmost}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)$. By D_1,

$$\forall \mathbf{X} N(\mathit{Start}_G^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \to N \leq Y)).$$

Then $\forall \mathbf{X} N(\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \to N \leq Y))$ by (14), and $\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, Y)$ follows by D_1. □

Formula (17) follows from (44) and (45).

9 Proof of Theorem 2

9.1 A few more theorems of *HTC*, continued

Claim: Let F be a formula over σ_0, let $\mathbf{U}$, $\mathbf{W}$ are disjoint tuples of distinct general variables of the same length such that the variables $\mathbf{W}$ are not free in F, and let n is a positive integer. The formula

$$\exists_{\geq \overline{n+1}} \mathbf{U}\, F \leftrightarrow \exists \mathbf{U}(F \wedge \exists_{\geq \overline{n}} \mathbf{W}(\mathbf{U} < \mathbf{W} \wedge F_{\mathbf{W}}^{\mathbf{U}})) \tag{46}$$

is provable in *HTC*.

Proof. *Left-to-right:* take $\mathbf{U}_1, \ldots, \mathbf{U}_{n+1}$ such that

$$\bigwedge_{i=1}^{n+1} F_{\mathbf{U}_i}^{\mathbf{U}} \wedge \bigwedge_{i<j} \neg(\mathbf{U}_i = \mathbf{U}_j). \tag{47}$$

We reason by cases, using the axiom

$$\bigvee_{k=1}^{n+1} \bigwedge_{i=1}^{n+1} \mathbf{U}_k \leq \mathbf{U}_i$$

from *Std* ("for some k, $\mathbf{U}_k$ is lexicographically first among $\mathbf{U}_1, \ldots, \mathbf{U}_{n+1}$"). Consider the k-th case $\bigwedge_{i=1}^{n+1} \mathbf{U}_k \leq \mathbf{U}_i$. From (47),

$$\mathbf{U}_k < \mathbf{U}_i \text{ and } F_{\mathbf{U}_i}^{\mathbf{U}} \qquad (i = 1, \ldots, n+1;\ i \neq k)$$

and

$$\neg(\mathbf{U}_i = \mathbf{U}_j) \qquad (1 \le i < j \le n+1;\ i, j \ne k).$$

Hence $\exists_{\ge\overline{n}}\mathbf{W}(\mathbf{U}_k < \mathbf{W} \wedge F^{\mathbf{U}}_{\mathbf{W}})$. Since $F^{\mathbf{U}}_{\mathbf{U}_k}$, it follows that

$$\exists\mathbf{U}(F \wedge \exists_{\ge\overline{n}}\mathbf{W}(\mathbf{U} < \mathbf{W} \wedge F^{\mathbf{U}}_{\mathbf{W}})).$$

Right-to-left: assume

$$\exists\mathbf{U}\left(F \wedge \exists\mathbf{W}_1\cdots\mathbf{W}_n\left(\bigwedge_{i=1}^{n}(\mathbf{U} < \mathbf{W}_i \wedge F^{\mathbf{U}}_{\mathbf{W}_i}) \wedge \bigwedge_{i<j}\neg(\mathbf{W}_i = \mathbf{W}_j)\right)\right).$$

This formula is equivalent to

$$\exists\mathbf{U}\mathbf{W}_1\cdots\mathbf{W}_n\left(F \wedge \bigwedge_{i=1}^{n}(\mathbf{U} < \mathbf{W}_i \wedge F^{\mathbf{U}}_{\mathbf{W}_i}) \wedge \bigwedge_{i<j}\neg(\mathbf{W}_i = \mathbf{W}_j)\right)$$

and can be rewritten as

$$\exists\mathbf{W}_0\mathbf{W}_1\cdots\mathbf{W}_n\left(F^{\mathbf{U}}_{\mathbf{W}_0} \wedge \bigwedge_{i=1}^{n}(\mathbf{W}_0 < \mathbf{W}_i \wedge F^{\mathbf{U}}_{\mathbf{W}_i}) \wedge \bigwedge_{1\le i<j\le n}\neg(\mathbf{W}_i = \mathbf{W}_j)\right).$$

It implies

$$\exists\mathbf{W}_0\mathbf{W}_1\cdots\mathbf{W}_n\left(\bigwedge_{i=0}^{n}F^{\mathbf{U}}_{\mathbf{W}_i} \wedge \bigwedge_{0\le i<j\le n}\neg(\mathbf{W}_i = \mathbf{W}_j)\right),$$

which is equivalent to $\exists_{\ge\overline{n+1}}\mathbf{U}\,F$. $\square$

Claim: If $\mathbf{X}$, $\mathbf{U}$ are disjoint tuples of distinct general variables of the same length, the variables $\mathbf{U}$ are not free in F, and $n > 0$, then the sentence

$$\forall\mathbf{X}\mathbf{V}(\exists\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{U},\mathbf{V},\overline{n})) \leftrightarrow \exists_{\ge\overline{n}}\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}})) \tag{48}$$

is provable in HTC.

Proof. By induction on n. If $n = 1$ then $\mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{U},\mathbf{V},\overline{n})$ in the left-hand side of (48) is equivalent to $F^{\mathbf{X}}_{\mathbf{U}}$ by D_0, and the right-hand side of (48) is equivalent to $\exists\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}})$. Induction step: we will show that the formula

$$\forall\mathbf{X}\mathbf{V}(\exists\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{U},\mathbf{V},\overline{n+1})) \leftrightarrow \exists_{\ge\overline{n+1}}\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}})) \tag{49}$$

is derivable from (48) in HTC. By (46), the right-hand side of (49) is equivalent to

$$\exists\mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}} \wedge \exists_{\ge\overline{n}}\mathbf{W}(\mathbf{U} < \mathbf{W} \wedge \mathbf{X} < \mathbf{W} \wedge F^{\mathbf{X}}_{\mathbf{W}})). \tag{50}$$

In the presence of $\mathbf{X} < \mathbf{U}$, the subformula $\mathbf{U} < \mathbf{W} \wedge \mathbf{X} < \mathbf{W}$ is equivalent to $\mathbf{U} < \mathbf{W}$. Hence (50) is equivalent to

$$\exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}} \wedge \exists_{\geq \overline{n}} \mathbf{W}(\mathbf{U} < \mathbf{W} \wedge F^{\mathbf{X}}_{\mathbf{W}})). \tag{51}$$

On the other hand, (48) can be rewritten as

$$\forall \mathbf{X}\mathbf{V}(\exists \mathbf{W}(\mathbf{X} < \mathbf{W} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_{F}(\mathbf{W}, \mathbf{V}, \overline{n})) \leftrightarrow \exists_{\geq \overline{n}} \mathbf{W}(\mathbf{X} < \mathbf{W} \wedge F^{\mathbf{X}}_{\mathbf{W}})),$$

and it implies

$$\exists \mathbf{W}(\mathbf{U} < \mathbf{W} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_{F}(\mathbf{W}, \mathbf{V}, \overline{n})) \leftrightarrow \exists_{\geq \overline{n}} \mathbf{W}(\mathbf{U} < \mathbf{W} \wedge F^{\mathbf{X}}_{\mathbf{W}}).$$

It follows that (51) is equivalent to

$$\exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}} \wedge \exists \mathbf{W}(\mathbf{U} < \mathbf{W} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_{F}(\mathbf{W}, \mathbf{V}, \overline{n}))).$$

By D_0, this formula is equivalent to the left-hand side of (49). $\square$

Claim: If $\mathbf{U}$ is a tuple of distinct general variables of the same length as $\mathbf{X}$ such that its members do not belong to $\mathbf{X}$ and are not free in F then the sentence

$$\forall \mathbf{V}(\exists \mathbf{U} \mathit{Start}^{\mathbf{X},\mathbf{V}}_{F}(\mathbf{U}, \mathbf{V}, \overline{n}) \leftrightarrow \exists_{\geq \overline{n}} \mathbf{U}\, F^{\mathbf{X}}_{\mathbf{U}}) \tag{52}$$

is provable in *HTC*.

Proof. The following is one of the axioms of *Std*:

$$\overline{n} \leq \overline{0} \vee \overline{n} = \overline{1} \vee \overline{n} > \overline{1}.$$

Case 1: $\overline{n} \leq \overline{0}$. The right-hand side of (52) is $\top$ and its left-hand side follows from D_0. *Case 2:* $\overline{n} = \overline{1}$. The right-hand side of (52) is $\exists \mathbf{X}_1 F^{\mathbf{X}}_{\mathbf{X}_1}$ and (52) is immediate from D_0. *Case 3:* $\overline{n} > \overline{1}$. The left-hand side of (52) is equivalent to

$$\exists \mathbf{X}(F \wedge \exists \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge \mathit{Start}^{\mathbf{X};\mathbf{V}}_{F}(\mathbf{U}, \mathbf{V}, \overline{n-1})))$$

by D_0, and the right-hand side is equivalent to

$$\exists \mathbf{X}(F \wedge \exists_{\geq \overline{n-1}} \mathbf{U}(\mathbf{X} < \mathbf{U} \wedge F^{\mathbf{X}}_{\mathbf{U}}))$$

by (46). These two formulas are equivalent to each other by (48). $\square$

Claim: For any formula F over σ_0 and any nonnegative integer n, the formula

$$\exists_{\leq \overline{n}} \mathbf{X}\, F \leftrightarrow \neg \exists_{\geq \overline{n+1}} \mathbf{X}\, F. \tag{53}$$

is provable in *HTC*.

Proof.

$$\begin{array}{rcl}
\neg\exists_{\geq\overline{n+1}}\mathbf{X}\,F & = & \neg\exists\mathbf{X}_1\cdots\mathbf{X}_{n+1}\left(\bigwedge_{i=1}^{n+1}F^{\mathbf{X}}_{\mathbf{X}_i}\wedge\bigwedge_{i<j}\neg(\mathbf{X}_i=\mathbf{X}_j)\right)\\
 & \leftrightarrow & \forall\mathbf{X}_1\cdots\mathbf{X}_{n+1}\neg\left(\bigwedge_{i=1}^{n+1}F^{\mathbf{X}}_{\mathbf{X}_i}\wedge\bigwedge_{i<j}\neg(\mathbf{X}_i=\mathbf{X}_j)\right)\\
 & \leftrightarrow & \forall\mathbf{X}_1\cdots\mathbf{X}_{n+1}\left(\bigwedge_{i=1}^{n+1}F^{\mathbf{X}}_{\mathbf{X}_i}\rightarrow\neg\bigwedge_{i<j}\neg(\mathbf{X}_i=\mathbf{X}_j)\right)\\
 & \leftrightarrow & \forall\mathbf{X}_1\cdots\mathbf{X}_{n+1}\left(\bigwedge_{i=1}^{n+1}F^{\mathbf{X}}_{\mathbf{X}_i}\rightarrow\bigvee_{i<j}\mathbf{X}_i=\mathbf{X}_j\right)\\
 & = & \exists_{\leq\overline{n}}\mathbf{X}\,F.
\end{array}$$

□

9.2 Proof of Theorem 2, Part 1

We will show now that for every precomputed term r, sentence (36) is provable in HTC.

Case 1: $r\leq\overline{0}$; (36) is

$$\forall\mathbf{V}\left(\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)\leftrightarrow\top\right). \tag{54}$$

From D_1,

$$\forall\mathbf{V}\mathbf{X}(\mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{X},\mathbf{V},\overline{0})\wedge\overline{0}\geq r\rightarrow\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)).$$

The conjunctive term $\mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{X},\mathbf{V},\overline{0})$ follows from D_0, and second conjunctive term $\overline{0}\geq r$ is an axiom of *Std*. Consequently $\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)$, which is equivalent to (54).

Case 2: for all n, $r>\overline{n}$; (36) is

$$\forall\mathbf{V}\left(\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)\leftrightarrow\bot\right). \tag{55}$$

Assume $\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)$. From D_1,

$$\forall\mathbf{V}(\mathit{Atleast}^{\mathbf{X};\mathbf{V}}_F(\mathbf{V},r)\rightarrow\exists\mathbf{X}N(\mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{X},\mathbf{V},N)\wedge N\geq r)).$$

Consequently $\exists N(N\geq r)$, which contradicts the *Std* axiom $\forall N\neg(N\geq r)$.

Case 3: for some n, $\overline{0}<r\leq\overline{n}$. Since the set of numerals is contiguous, r is a numeral $\overline{m}$ ($m>0$). By (10), formula (36) can be rewritten as

$$\forall\mathbf{V}\left(\exists\mathbf{X}\,\mathit{Start}^{\mathbf{X};\mathbf{V}}_F(\mathbf{X},\mathbf{V},\overline{m})\leftrightarrow\exists_{\geq\overline{m}}\mathbf{X}F\right),$$

which is the universal closure of (52).

9.3 Proof of Theorem 2, Part 2

We will show now that for every precomputed term r, sentence (37) is provable in HTC.

Case 1: $r < \overline{0}$; (37) is

$$\forall \mathbf{V}\left(Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \leftrightarrow \bot\right). \tag{56}$$

From D_1,

$$Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \rightarrow \forall \mathbf{X}(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, \overline{0}) \rightarrow \overline{0} \leq r).$$

By D_0, $Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, \overline{0})$, so that

$$Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \rightarrow \overline{0} \leq r.$$

From the *Std* axiom $\neg(\overline{0} \leq r)$ we conclude that $\neg Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r)$, which is equivalent to (56).

Case 2: for all n, $r > \overline{n}$; (37) is

$$\forall \mathbf{V}\left(Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \leftrightarrow \top\right). \tag{57}$$

From D_1,

$$\forall \mathbf{V}(\forall \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow N \leq r) \rightarrow Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r)).$$

The antecedent $\forall \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow N \leq r)$ follows from the *Std* axiom $\forall N(N \leq r)$. Hence $Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r)$, which is equivalent to (57).

Case 3: for some n, $\overline{0} \leq r \leq \overline{n}$. Since the set of numerals is contiguous r is a numeral $\overline{m}$ ($m \geq 0$), so that (37) is

$$\forall \mathbf{V}\left(Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{m}) \leftrightarrow \exists_{\leq \overline{m}} \mathbf{X} F\right).$$

By (11) and (53), this formula is equivalent to

$$\forall \mathbf{V}\left(\neg Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{m+1}) \leftrightarrow \neg \exists_{\geq \overline{m+1}} \mathbf{X} F\right),$$

which follows from (36).

10 Review: HT-interpretations

A *propositional HT-interpretation* is a pair $\langle \mathcal{X}, \mathcal{Y} \rangle$, where $\mathcal{Y}$ is a set of propositional atoms, and $\mathcal{X}$ is a subset of $\mathcal{Y}$. In terms of Kripke models with two worlds, $\mathcal{X}$ is the here-world, and $\mathcal{Y}$ is the there-world. The recursive definition of the satisfaction relation between HT-interpretations and propositional formulas can be extended to infinitary propositional formulas [Truszczynski, 2012, Definition 2]. Equilibrium models of a set of formulas [Pearce, 1997, Pearce, 1999] are defined as its HT-models satisfying a certain minimality condition. A set $\mathcal{X}$ of atoms is a stable model of a set of infinitary propositional formulas iff $\langle \mathcal{X}, \mathcal{X} \rangle$ is an equlibrium model of that set [Truszczynski, 2012, Theorem 3]. Thus stable models of an MGC program Π can be characterized as sets $\mathcal{X}$ such that $\langle \mathcal{X}, \mathcal{X} \rangle$ is an equlibrium model of $\tau\Pi$.

The definition of a many-sorted HT-interpretation [Fandinno *et al.*, 2024, Appendices A and B] extends this construction to many-sorted first-order languages. In classical semantics of first-order formulas, the recursive definition of the satisfaction relation between an interpretation I of a signature σ and a sentence F over σ involves extending σ by new object constants d^*, which represent elements d of the domain of I [Lifschitz *et al.*, 2008, Section 1.2.2]. The extended signature is denoted by σ^I. In the definition of a many-sorted HT-interpretation, the predicate symbols of σ are assumed to be partitioned into extensional and intensional. For any interpretation I of such a signature σ, $I^{\downarrow}$ stands for the set of all atomic sentences over σ^I that have the form $p(\mathbf{d}^*)$, where p is intensional, $\mathbf{d}$ is a tuple of elements of appropriate domains of σ, and $I \models p(\mathbf{d}^*)$. An *HT-interpretation* of a many-sorted signature σ is a pair $\langle \mathcal{H}, I \rangle$, where I is an interpretation of σ, and $\mathcal{H}$ is a subset of $I^{\downarrow}$. In terms of Kripke models, I is the there-world, and $\mathcal{H}$ describes the intensional predicates in the here-world.

The satisfaction relation between HT-interpretations and sentences is denoted by $\models_{ht}$, to distinguish it from classical satisfaction. According to the persistence property of this relation, $\langle \mathcal{H}, I \rangle \models_{ht} F$ implies $I \models F$ for every sentence F over σ [Fandinno *et al.*, 2024, Proposition 3(a)].

The soundness and completeness theorem for the many-sorted logic of here-and-there [Fandinno and Lifschitz, 2023a, Theorem 2] can be stated as follows:

For any set Γ of sentences over a many-sorted signature σ and any sentence F over σ, the following two conditions are equivalent:

(i) every HT-interpretation of σ satisfying Γ satisfies F;

(ii) F can be derived from Γ in first-order intuitionistic logic extended by

- *axiom schemas (6) and (7) for all formulas F, G, H over σ;*
- *the axioms*
$$X = Y \vee X \neq Y, \tag{58}$$
where X and Y are variables of the same sort;
- *the axioms*
$$p(\mathbf{X}) \vee \neg p(\mathbf{X}), \tag{59}$$
where p is an extensional predicate symbol, and $\mathbf{X}$ is a tuple of distinct variables of appropriate sorts.

The deductive system described in clause (ii) is denoted by $SQHT^{=}$ [Fandinno and Lifschitz, 2023a, Section 5.1].

11 Proof of Theorem 3

In the special case of the signature σ_2 (Section 3.1), we designate comparison symbols (1) as extensional, and all other predicate symbols (that is, p/n, $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}$, $\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}$ and $\mathit{Start}_F^{\mathbf{X};\mathbf{V}}$) as intensional. This convention allows us to generalize some of the definitions from Section 6 to arbitrary many-sorted signatures with predicate constants classified into extensional and intensional. For any such signature σ, by σ' we denote the signature obtained from it by adding, for every intensional constant p, a new predicate constant p' of the same arity. The formula

$$\forall \mathbf{X}(p(\mathbf{X}) \to p'(\mathbf{X})), \tag{60}$$

where p is intensional and $\mathbf{X}$ is a tuple of distinct variables of appropriate sorts, is denoted by $\mathcal{A}(p)$, and $\mathcal{A}$ stands for the set of these formulas for all intensional predicate constants p. For any formula F over the signature σ, by F' we denote the formula over σ' obtained from F by replacing every occurrence of every intensional predicate symbol p by p'. Then the transformation γ is defined as in Section 6.

For any HT-interpretation $\langle \mathcal{H}, I \rangle$ of σ, $I^{\mathcal{H}}$ stands for the interpretation of σ' that has the same domains as I, interprets function constants and extensional predicate constants of σ in the same way as I, and interprets the other predicate constants p, p' as follows:

$$\begin{array}{l} I^{\mathcal{H}} \models p(\mathbf{d}^*) \text{ iff } p(\mathbf{d}^*) \in \mathcal{H}; \\ I^{\mathcal{H}} \models p'(\mathbf{d}^*) \text{ iff } I \models p(\mathbf{d}^*). \end{array} \tag{61}$$

From the second line of (61) we can derive a more general assertion:

$$I^{\mathcal{H}} \models p'(\mathbf{t}) \text{ iff } I \models p(\mathbf{t}) \tag{62}$$

for every tuple $\mathbf{t}$ of ground terms over the signature σ^I. Indeed, the value assigned to $\mathbf{t}$ by the interpretation $I^{\mathcal{H}}$ (symbolically, $\mathbf{t}^{I^{\mathcal{H}}}$) is the same as the value $\mathbf{t}^I$, assigned to $\mathbf{t}$ by I, because $I^{\mathcal{H}}$ and I interpret all symbols occurring in $\mathbf{t}$ in the same way. In the second line of (61), take $\mathbf{d}$ to be the common value of $\mathbf{t}^{I^{\mathcal{H}}}$ and $\mathbf{t}^I$. Then

$$I^{\mathcal{H}} \models p'\left(\left(\mathbf{t}^{I^{\mathcal{H}}}\right)^*\right) \text{ iff } I \models p\left(\left(\mathbf{t}^I\right)^*\right),$$

which is equivalent to (62).

Lemma 1. *An interpretation of the signature σ' satisfies $\mathcal{A}$ iff it can be represented in the form $I^{\mathcal{H}}$ for some HT-interpretation $\langle \mathcal{H}, I \rangle$.*

Proof. For the if-part, take any formula (60) from $\mathcal{A}$. We need to show that $I^{\mathcal{H}}$ satisfies all sentences of the form $p(\mathbf{d}^*) \to p'(\mathbf{d}^*)$. Assume that $I^{\mathcal{H}} \models p(\mathbf{d}^*)$. Then $p(\mathbf{d}^*) \in \mathcal{H} \subseteq I^{\downarrow}$, and consequently $I \models p(\mathbf{d}^*)$, which is equivalent to $I^{\mathcal{H}} \models p'(\mathbf{d}^*)$.

For the only-if part, take any interpretation J of σ' that satisfies $\mathcal{A}$. Let I be the interpretation of σ that has the same domains as J, interprets function constants and extensional predicate constants in the same way as J, and interprets every intensional p in accordance with the condition

$$I \models p(\mathbf{d}^*) \text{ iff } J \models p'(\mathbf{d}^*). \tag{63}$$

Take $\mathcal{H}$ to be the set of all atoms of the form $p(\mathbf{d}^*)$ with intensional p that are satisfied by J. Since J satisfies $\mathcal{A}$, J satisfies $p'(\mathbf{d}^*)$ for every atom $p(\mathbf{d}^*)$ from $\mathcal{H}$. By (63), it follows that all atoms from $\mathcal{H}$ are satisfied by I, so that $\mathcal{H}$ is a subset of $I^{\downarrow}$. It follows that $\langle \mathcal{H}, I\rangle$ is an HT-interpretation. Let us show that $I^{\mathcal{H}} = J$. Each of the interpretations $I^{\mathcal{H}}$ and J has the same domains as I and interprets all function constants and extensional predicate constants in the same way as I. For every intensional p and any tuple $\mathbf{d}$ of elements of appropriate domains, each of the conditions $I^{\mathcal{H}} \models p(\mathbf{d}^*)$, $J \models p(\mathbf{d}^*)$ is equivalent to $p(\mathbf{d}^*) \in \mathcal{H}$, and each of the conditions $I^{\mathcal{H}} \models p'(\mathbf{d}^*)$, $J \models p'(\mathbf{d}^*)$ is equivalent to $I \models p(\mathbf{d}^*)$. □

Lemma 2. *For every HT-interpretation $\langle \mathcal{H}, I\rangle$ of σ and every sentence F over the signature σ^I, $I^{\mathcal{H}} \models F'$ iff $I \models F$.*

Proof. We will consider the case when F is a ground atom $p(\mathbf{t})$; extension to arbitrary sentences by induction is straightforward. If p is intensional then F' is $p'(\mathbf{t})$, so that the assertion of the lemma turns into property (62). If p is extensional then F' is $p(\mathbf{t})$; $I^{\mathcal{H}} \models p(\mathbf{t})$ iff $I \models p(\mathbf{t})$ because $I^{\mathcal{H}}$ interprets all symbols occurring in F in the same way as I. □

Lemma 3. *For every HT-interpretation $\langle \mathcal{H}, I\rangle$ of σ and every sentence F over the signature σ^I, $I^{\mathcal{H}} \models \gamma F$ iff $\langle \mathcal{H}, I\rangle \models_{ht} F$.*

Proof. The proof is by induction on the number of propositional connectives and quantifiers in F. We consider below the more difficult cases when F is an atomic formula, a negation, or an implication.

Case 1: F is an atomic formula $p(\mathbf{t})$. Then γF is $p(\mathbf{t})$ too; we need to check that

$$I^{\mathcal{H}} \models p(\mathbf{t}) \text{ iff } \langle \mathcal{H}, I\rangle \models_{ht} p(\mathbf{t}). \tag{64}$$

Case 1.1: p is intensional. Let $\mathbf{d}$ be the common value of $\mathbf{t}^{I^{\mathcal{H}}}$ and $\mathbf{t}^I$. The left-hand side of (64) is equivalent to $I^{\mathcal{H}} \models p(\mathbf{d}^*)$ and consequently to $p(\mathbf{d}^*) \in \mathcal{H}$. The right-hand side of (64) is equivalent to $p\left(\left(\mathbf{t}^I\right)^*\right) \in \mathcal{H}$, which is equivalent to $p(\mathbf{d}^*) \in \mathcal{H}$ as well.

Case 1.2: p is extensional. Each of the conditions $I^{\mathcal{H}} \models p(\mathbf{t})$, $\langle \mathcal{H}, I\rangle \models_{ht} p(\mathbf{t})$ is equivalent to $I \models p(\mathbf{t})$.

Case 2: F is $\neg G$. Then γF is $\neg G'$; we need to check that

$$I^{\mathcal{H}} \not\models G \text{ iff } \langle \mathcal{H}, I\rangle \models_{ht} \neg G'.$$

By Lemma 2, the left-hand side is equivalent to $I \not\models G'$. From the definition of $\models_{ht}$ we can conclude that the right-hand side is equivalent to $I \not\models G'$ as well.

Case 3: F is $G \to H$. Then γF is $(\gamma G \to \gamma H) \wedge (G' \to H')$, so that the condition $I^{\mathcal{H}} \models \gamma F$ holds iff

$$I^{\mathcal{H}} \not\models \gamma G \text{ or } I^{\mathcal{H}} \models \gamma H \tag{65}$$

and

$$I^{\mathcal{H}} \models G' \to H'. \tag{66}$$

By the induction hypothesis, (65) is equivalent to

$$\langle \mathcal{H}, I\rangle \not\models_{ht} G \text{ or } \langle \mathcal{H}, I\rangle \models_{ht} H. \tag{67}$$

By Lemma 2, (66) is equivalent to

$$I \models G \to H. \tag{68}$$

The conjunction of (67) and (68) is equivalent to $\langle \mathcal{H}, I\rangle \models_{ht} G \to H$. $\square$

Proof of Theorem 3. To prove a formula in *HTC* means to derive it in first-order intuitionistic logic from (6), (7), *Std*, *Ind*, D_0 and D_1. Since the universal closures of (58) and (59) belong to *Std*, it follows that a formula is provable in *HTC* iff it can be derived from *Std*, *Ind*, D_0 and D_1 in $SQHT^=$. Consequently $F \leftrightarrow G$ is provable in *HTC* iff

$$G \text{ can be derived in } SQHT^= \text{ from } Std,\ Ind,\ D_0,\ D_1 \text{ and } F \tag{69}$$

and

$$F \text{ can be derived in } SQHT^= \text{ from } Std,\ Ind,\ D_0,\ D_1 \text{ and } G. \tag{70}$$

By the soundness and completeness theorem quoted in Section 10, (69) is equivalent to the condition

$$\begin{array}{c} G \text{ is satisfied by every HT-interpretation of } \sigma_2 \\ \text{that satisfies } Std,\ Ind,\ D_0,\ D_1 \text{ and } F. \end{array}$$

By Lemma 3, this condition can be further reformulated as follows:

$$\begin{array}{c} \text{for every HT-interpretation } \langle \mathcal{H}, I\rangle \text{ of } \sigma_2,\ I^{\mathcal{H}} \text{ satisfies } \gamma G \\ \text{if } I^{\mathcal{H}} \text{ satisfies } \gamma(Ind),\ \gamma(Std),\ \gamma D_0,\ \gamma D_1 \text{ and } \gamma F. \end{array}$$

Then, by Lemma 1, (69) is equivalent to the condition

$$\begin{array}{c} \gamma G \text{ is satisfied by every interpretation of } \sigma_2' \\ \text{that satisfies } \mathcal{A},\ \gamma(Ind),\ \gamma(Std),\ \gamma D_0,\ \gamma D_1 \text{ and } \gamma F. \end{array}$$

Similarly, (70) is equivalent to the condition

$$\begin{array}{c} \gamma F \text{ is satisfied by every interpretation of } \sigma_2' \\ \text{that satisfies } \mathcal{A},\ \gamma(Ind),\ \gamma(Std),\ \gamma D_0,\ \gamma D_1 \text{ and } \gamma G. \end{array}$$

Consequently $F \leftrightarrow G$ is provable in *HTC* iff

$$\begin{array}{c}\gamma F \leftrightarrow \gamma G \text{ is satisfied by every interpretation of } \sigma_2' \\ \text{that satisfies } \mathcal{A},\ \gamma(\mathit{Ind}),\ \gamma(\mathit{Std}),\ \gamma D_0, \text{ and } \gamma D_1.\end{array}$$

Since all predicate constants occurring in *Std* are comparisons, $\gamma(\mathit{Std})$ is equivalent to *Std*, so that $\gamma(\mathit{Std})$ here can be replaced by *Std*. It remains to observe that $\mathcal{A}$, $\gamma(\mathit{Ind})$, *Std*, γD_0 and γD_1 is the list of all axioms of HTC'. $\square$

12 Standard HT-interpretations

In preparation for the proof of Theorem 4, we describe here the class of standard HT-interpretations of the signature σ_1 and prove the soundness and completeness of HTC^ω with respect to standard HT-interpretations.

An interpretation of σ_1 is *standard* if its restriction to σ_0 is standard (see Section 2.5) and it satisfies *Defs*. For every set $\mathcal{X}$ of precomputed atoms, $\mathcal{X}^\uparrow$ stands for the standard interpretation of σ_1 defined by the following conditions:

(a) a precomputed atom is satisfied by $\mathcal{X}^\uparrow$ iff it belongs to $\mathcal{X}$;

(b) an extended precomputed atom $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{v}, r)$ is satisfied by $\mathcal{X}^\uparrow$ iff

$$\mathcal{X}^\uparrow \models (\exists_{\geq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}\ ;$$

(c) an extended precomputed atom $\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{v}, r)$ is satisfied by $\mathcal{X}^\uparrow$ iff

$$\mathcal{X}^\uparrow \models (\exists_{\leq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}\ .$$

The operation $\mathcal{X} \mapsto \mathcal{X}^\uparrow$ is opposite to the operation $I \mapsto I^\downarrow$ defined in Section 10, in the sense that

- for any standard interpretation I of σ_1, $(I^\downarrow)^\uparrow = I$, and
- for any set $\mathcal{X}$ of precomputed atoms, the set of precomputed atoms in $(\mathcal{X}^\uparrow)^\downarrow$ is $\mathcal{X}$.

This construction is extended to HT-interpretations as follows. An HT-interpretation $\langle \mathcal{H}, I\rangle$ of σ_1 is *standard* if the restriction of I to σ_0 is standard and $\langle \mathcal{H}, I\rangle$ satisfies *Defs*. For any standard HT-interpretation $\langle \mathcal{H}, I\rangle$, I satisfies *Defs* by the persistence property of HT-interpretations (Section 10), so that I is standard as well. For any pair $\mathcal{X}$, $\mathcal{Y}$ of sets of precomputed atoms such that $\mathcal{X} \subseteq \mathcal{Y}$, the pair $\langle \mathcal{X}, \mathcal{Y}^\uparrow\rangle$ is an HT-interpretation of σ_1, because $\mathcal{X} \subseteq \mathcal{Y} \subseteq (\mathcal{Y}^\uparrow)^\downarrow$. Let $\mathcal{H}$ be the superset of $\mathcal{X}$ obtained from it by adding all extended precomputed atoms $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{v}, r)$ such that

$$\langle \mathcal{X}, \mathcal{Y}^\uparrow\rangle \models_{ht} (\exists_{\geq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}$$

and all extended precomputed atoms $Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{v},r)$ such that

$$\left\langle \mathcal{X},\mathcal{Y}^{\uparrow}\right\rangle \models_{ht} (\exists_{\leq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}.$$

For every atom $Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{v},r)$ in $\mathcal{H}$,

$$\mathcal{Y}^{\uparrow} \models (\exists_{\geq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}$$

by persistence. Consequently every such atom belongs to $(\mathcal{Y}^{\uparrow})^{\downarrow}$. Similarly, every atom $Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{v},r)$ in $\mathcal{H}$ belongs to $(\mathcal{Y}^{\uparrow})^{\downarrow}$ as well. It follows that $\left\langle \mathcal{H},\mathcal{Y}^{\uparrow}\right\rangle$ is an HT-interpretation of σ_1. We denote this HT-interpretation by $\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow}$.

This HT-interpretation is standard. Indeed, a precomputed atom of the form $Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{v},r)$ belongs to $\mathcal{H}$ iff the formula $(\exists_{\geq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}$ is satisfied by $\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow}$, because this HT-interpretation interprets sentences over the signature σ_0 in the same was as $\left\langle \mathcal{X},\mathcal{Y}^{\uparrow}\right\rangle$. Similarly, $Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{v},r)$ belongs to $\mathcal{H}$ iff the formula $(\exists_{\leq r}\mathbf{X}F)^{\mathbf{V}}_{\mathbf{v}}$ is satisfied by $\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow}$. It follows that $\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow}$ satisfies *Defs*.

Conversely, every standard HT-interpretation of σ_1 can be represented in the form $\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow}$. Indeed, for any standard HT-interpretation $\langle \mathcal{H},I\rangle$ of σ_1, take $\mathcal{X}$ to be the set of precomputed atoms in $\mathcal{H}$, and take $\mathcal{Y}$ to be $I^{\downarrow}$. Then

$$\mathcal{X} \subseteq \mathcal{H} \subseteq I^{\downarrow} = \mathcal{Y}$$

and

$$\langle \mathcal{X},\mathcal{Y}\rangle^{\uparrow} = \left\langle \mathcal{H},\mathcal{Y}^{\uparrow}\right\rangle = \left\langle \mathcal{H},(I^{\downarrow})^{\uparrow}\right\rangle = \langle \mathcal{H},I\rangle.$$

Soundness and Completeness Theorem. *For any set Γ of sentences over σ_1 and any sentence F over σ_1, F can be derived from Γ in HTC^{ω} iff every standard HT-interpretation satisfying Γ satisfies F.*

The proof of the theorem refers to *ω-interpretations* of many-sorted signatures [Fandinno and Lifschitz, 2023a, Section 5.2]. In case of the signatures σ_0, σ_1 and σ_2, ω-interpretations are characterized by two conditions:

- every element of the domain of general variables is represented by a precomputed term;
- every element of the domain of integer variables is represented by a numeral.

An *ω-model* of a set Γ of sentences is an HT-interpretation $\langle \mathcal{H},I\rangle$ satisfying Γ such that I is an ω-interpretation.

Lemma 4. *An HT-interpretation of σ_1 is isomorphic to a standard HT-interpretation iff it is an ω-model of Std and Defs.*

Proof. The only-if part is obvious. If $\langle \mathcal{H},I\rangle$ is an ω-model of *Std* and *Defs* then the function that maps every precomputed term to the corresponding element of the domain of general variables in I is an isomorphism between a standard HT-interpretation and $\langle \mathcal{H},I\rangle$. □

Proof of the soundness and completeness theorem. The deductive system $SQHT^{\omega}$ [Fandinno and Lifschitz, 2023a, Section 5.3] for the signature σ_1 can be described as $SQHT^{=}$ (see Section 10) extended by the two ω-rules from Section 7. According to Theorem 4 from that paper, for any set Γ of sentences over σ_1 and any sentence F over σ_1, F can be derived from Γ in $SQHT^{\omega}$ iff every ω-model of Γ satisfies F. On the other hand, HTC^{ω} can be described as $SQHT^{\omega}$ extended by the axioms *Std* and *Defs*. It follows that for any set Γ of sentences over σ_1 and any sentence F over σ_1, F can be derived from Γ in HTC^{ω} iff every ω-model of *Std*, *Defs* and Γ satisfies F. The assertion of the theorem follows by Lemma 4.

13 Grounding

Recall that in Section 11 we subdivided the predicate symbols of the signature σ_2 into extensional and intensional. This classification applies, in particular, to the predicate symbols of σ_1: comparison symbols (1) are extensional, and the symbols p/n, $Atleast_F^{\mathbf{X};\mathbf{V}}$ and $Atmost_F^{\mathbf{X};\mathbf{V}}$ are intensional. *Extended precomputed atoms* are atomic formulas $p(\mathbf{t})$ over the signature σ_1 such that p is intensional, and $\mathbf{t}$ is a tuple of precomputed terms.

The proof of Theorem 4 refers to the grounding transformation $F \mapsto F^{\mathrm{prop}}$, which converts sentences over σ_0 into infinitary propositional combinations of precomputed atoms, and sentences over σ_1 into infinitary propositional combinations of extended precomputed atoms [Truszczynski, 2012, Section 3], [Lifschitz *et al.*, 2019, Section 5], [Lifschitz, 2022, Section 10], [Fandinno *et al.*, 2024, Section 8].[3] This transformation is defined as follows:

- if F is $p(\mathbf{t})$, where p is intensional, then F^{prop} is obtained from F by replacing each member of $\mathbf{t}$ by the precomputed term obtained from it by evaluating arithmetic functions;
- if F is $t_1 \prec t_2$, then F^{prop} is $\top$ if the values of t_1 and t_2 are in the relation $\prec$, and $\bot$ otherwise;
- $(\neg F)^{\mathrm{prop}}$ is $\neg F^{\mathrm{prop}}$;
- $(F \odot G)^{\mathrm{prop}}$ is $F^{\mathrm{prop}} \odot G^{\mathrm{prop}}$ for every binary connective $\odot$;
- $(\forall X\, F)^{\mathrm{prop}}$ is the conjunction of the formulas $(F_t^X)^{\mathrm{prop}}$ over all precomputed terms t if X is a general variable, and over all numerals t if X is an integer variable;
- $(\exists X\, F)^{\mathrm{prop}}$ is the disjunction of the formulas $(F_t^X)^{\mathrm{prop}}$ over all precomputed terms t if X is a general variable, and over all numerals t if X is an integer variable.

[3] In some of these papers, the transformation $F \mapsto F^{\mathrm{prop}}$ is denoted by *gr*. We take the liberty to identify precomputed terms t with their names t^*.

For any set Γ of sentences over σ_1, Γ^{prop} stands for $\{F^{\text{prop}} : F \in \Gamma\}$.

The lemma below relates the meaning of a sentence over σ_1 to the meaning of its grounding. It is similar to Proposition 4 from Truszczynski's article [2012] and can be proved by induction in a similar way.

Lemma 5. *For any HT-interpretation $\langle \mathcal{H}, I\rangle$ of σ_1 such that the restriction of I to σ_0 is standard, and any sentence F over σ_1,*

$$\langle \mathcal{H}, I\rangle \models_{ht} F \text{ iff } \langle \mathcal{H}, I^{\downarrow}\rangle \models_{ht} F^{\text{prop}}.$$

The next lemma relates $(\tau^*\Pi)^{\text{prop}}$ to $\tau\Pi$.

Lemma 6. *For any* MGC *program Π, every propositional HT-interpretation satisfying $Defs^{\text{prop}}$ satisfies also the formula $(\tau^*\Pi)^{\text{prop}} \leftrightarrow \tau\Pi$.*

Proof. It is sufficient to consider the case when Π is a single pure rule R. The equivalence $(\tau^*R)^{\text{prop}} \leftrightarrow \tau R$ is provable in the deductive system $HT^{\infty} + Defs^{\text{prop}}$ [Lifschitz, 2022, Theorem 1]. The assertion of the lemma follows from this theorem, because every HT-interpretation satisfies all axioms of HT^{∞}, and satisfaction is preserved by the inference rules of HT^{∞}. $\square$

14 Proof of Theorem 4

Lemma 7. *For any* MGC *program Π and any sets $\mathcal{X}$, $\mathcal{Y}$ of precomputed atoms such that $\mathcal{X} \subseteq \mathcal{Y}$,*

$$\langle \mathcal{X}, \mathcal{Y}\rangle^{\uparrow} \models_{ht} \tau^*\Pi \tag{71}$$

iff $\langle \mathcal{X}, \mathcal{Y}\rangle \models_{ht} \tau\Pi$.

Proof. Condition (71) can be rewritten as $\langle \mathcal{H}, \mathcal{Y}^{\uparrow}\rangle \models_{ht} \tau^*\Pi$, and, by Lemma 5, it is equivalent to

$$\langle \mathcal{H}, (\mathcal{Y}^{\uparrow})^{\downarrow}\rangle \models_{ht} (\tau^*\Pi)^{\text{prop}}. \tag{72}$$

On the other hand,

$$\langle \mathcal{H}, \mathcal{Y}^{\uparrow}\rangle = \langle \mathcal{X}, \mathcal{Y}\rangle^{\uparrow} \models_{ht} Defs.$$

By Lemma 5, it follows that $\langle \mathcal{H}, (\mathcal{Y}^{\uparrow})^{\downarrow}\rangle$ satisfies $Defs^{\text{prop}}$. By Lemma 6, we can conclude that (72) is equivalent to

$$\langle \mathcal{H}, (\mathcal{Y}^{\uparrow})^{\downarrow}\rangle \models_{ht} \tau\Pi.$$

This codition is equivalent to $\langle \mathcal{H}, \mathcal{Y}\rangle \models_{ht} \tau\Pi$, because the formula $\tau\Pi$ is formed from precomputed atoms. $\square$

Proof of the theorem. We need to show that the formula $\tau^*\Pi_1 \leftrightarrow \tau^*\Pi_2$ is provable in HTC^{ω} iff $\tau\Pi_1$ is strongly equivalent to $\tau\Pi_2$. This formula is provable in HTC^{ω} iff

$$\tau^*\Pi_2 \text{ is derivable in } HTC^{\omega} \text{ from } \tau^*\Pi_1 \tag{73}$$

and

$$\tau^*\Pi_1 \text{ is derivable in } HTC^\omega \text{ from } \tau^*\Pi_2. \tag{74}$$

On the other hand, the characterization of strong equivalence of infinitary propositional formulas in terms of propositional HT-interpretations [Harrison *et al.*, 2017, Theorem 3] shows that $\tau\Pi_1$ is strongly equivalent to $\tau\Pi_2$ iff

$$\begin{array}{c}\text{for every propositional HT-interpretation } \langle\mathcal{X},\mathcal{Y}\rangle,\\ \text{if } \langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_1 \text{ then } \langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_2\end{array} \tag{75}$$

and

$$\begin{array}{c}\text{for every propositional HT-interpretation } \langle\mathcal{X},\mathcal{Y}\rangle,\\ \text{if } \langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_2 \text{ then } \langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_1\end{array} \tag{76}$$

We will show that conditions (73) and (75) are equivalent to each other; the equivalence between (74) and (76) is proved in a similar way.

Assume that condition (73) is satisfied but condition (75) is not, so that $\langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_1$ and $\langle\mathcal{X},\mathcal{Y}\rangle \not\models_{ht} \tau\Pi_2$ for some propositional HT-interpretation $\langle\mathcal{X},\mathcal{Y}\rangle$. By Lemma 7, $\langle\mathcal{X},\mathcal{Y}\rangle^\uparrow \models_{ht} \tau^*\Pi_1$ and $\langle\mathcal{X},\mathcal{Y}\rangle^\uparrow \not\models_{ht} \tau^*\Pi_2$. Thus there exists a standard HT-interpretation of σ_1 that satisfies $\tau^*\Pi_1$ but does not satisfy $\tau^*\Pi_2$. This is in contradiction with the fact that HTC^ω is sound with respect to standard interpretations (Section 12).

Assume now that (73) is not satisfied. Since HTC^ω is complete with respect to standard interpretations (Section 12), there exists a standard interpretation that satisfies $\tau^*\Pi_1$ but does not satisfy $\tau^*\Pi_2$. Consider a representation of this interpretation in the form $\langle\mathcal{X},\mathcal{Y}\rangle^\uparrow$. By Lemma 7, $\langle\mathcal{X},\mathcal{Y}\rangle \models_{ht} \tau\Pi_1$ and $\langle\mathcal{X},\mathcal{Y}\rangle \not\models_{ht} \tau\Pi_2$, so that condition (75) is not satisfied either.

15 Proofs of Theorems 1 and 5

15.1 Deductive system HTC_2^ω

Proofs of Theorems 1 and 5 refer to the deductive system HTC_2^ω, which is a straightforward extension of HTC^ω to the signature σ_2. Its derivable objects are sequents over σ_2. Its axioms and inference rules are those of intuitionistic logic for the signature σ_2 extended by

- axiom schemas (6) and (7) for all formulas F, G, H over σ_2,
- axioms *Std* and *Defs*, and
- the ω-rules from Section 7 extended to sequents over σ_2.

Any sentence provable in *HTC* can be derived in HTC_2^ω from D_0 and D_1. Indeed, the only axioms of *HTC* that are not included in HTC_2^ω are *Ind*, D_0 and D_1, and all instances of *Ind* can be proved using the second ω-rule, as discussed in Section 7. We will prove a stronger assertion:

Lemma 8. *Any sentence provable in HTC can be derived in* HTC_2^ω *from* D_0.

We will prove also the following conservative extension property:

Lemma 9. *Every sentence over the signature σ_1 derivable in HTC_2^ω from D_0 is provable in HTC^ω.*

The assertion of Theorem 5 follows from these two lemmas. The lemmas are proved in Sections 15.3 and 15.4.

The assertion of Theorem 1 follows from Theorems 4 and 5.

15.2 Some formulas derivable in HTC^ω and HTC_2^ω

In Section 9.1 we showed that formula (46) is provable in HTC. All axioms of HTC used in that proof are among the axioms of HTC^ω, so that this formula is provable in HTC^ω as well.

Claim: For any formula F over σ_0 and any integers m, n such that $m \geq n$, the formula

$$\exists_{\geq \overline{m}} \mathbf{U}\, F \to \exists_{\geq \overline{n}} \mathbf{U}\, F \tag{77}$$

is provable in HTC^ω.

Proof. It is sufficient to consider the case when $m = n + 1$; then the general case will follow by induction. We can also assume that n is positive, because otherwise the consequent of (77) is $\top$. From (46),

$$\exists_{\geq \overline{n+1}} \mathbf{U}\, F \to \exists \mathbf{W} (\exists_{\geq \overline{n}} \mathbf{U} (\mathbf{W} < \mathbf{U} \wedge F)).$$

We can rewrite the consequent of this implication as

$$\exists \mathbf{W} \mathbf{U}_1 \cdots \mathbf{U}_n \left(\bigwedge_{i=1}^{n} \left(\mathbf{W} < \mathbf{U}_i \wedge F^{\mathbf{U}}_{\mathbf{U}_1} \right) \wedge \bigwedge_{i<j} \neg(\mathbf{U}_i = \mathbf{U}_j) \right).$$

It implies

$$\exists \mathbf{U}_1 \cdots \mathbf{U}_n \left(\bigwedge_{i=1}^{n} F^{\mathbf{U}}_{\mathbf{U}_1} \wedge \bigwedge_{i<j} \neg(\mathbf{U}_i = \mathbf{U}_j) \right),$$

which is the consequent of (77). □

Claim: For any formula F over σ_0, any integer n, and any precomputed term r, the formula

$$\overline{n} \geq r \wedge \exists_{\geq \overline{n}} \mathbf{U}\, F \to \exists_{\geq r} \mathbf{U}\, F \tag{78}$$

is provable in HTC^ω.

Proof. If $\overline{n} < r$ then the antecedent of (78) is equivalent to $\bot$. If $r \leq \overline{0}$ then the consequent of (78) is $\top$. If $\overline{n} \geq r > 0$ then r is a numeral $\overline{m}$, because the set of numerals is contiguous, so that (78) follows from (77). □

In Section 9.1 we showed that formula (53) is provable in HTC. All axioms of HTC used in that proof are among the axioms of HTC^ω, so that this formula is provable in HTC^ω as well.

Claim: The formula

$$\forall \mathbf{V} N(N \geq \overline{0} \to (\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \leftrightarrow \neg \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N + \overline{1}))) \tag{79}$$

is provable in HTC_2^ω.

Proof: By *Defs* and (53), for every nonnegative n,

$$\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{n}) \leftrightarrow \exists_{\leq \overline{n}} \mathbf{U}\, F \leftrightarrow \neg \exists_{\geq \overline{n+1}} \mathbf{U}\, F \leftrightarrow \neg \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{n} + \overline{1}).$$

It follows that for every integer n,

$$\forall \mathbf{V}(\overline{n} \geq \overline{0} \to (\mathit{Atmost}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{n}) \leftrightarrow \neg \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{n} + \overline{1}))).$$

Formula (79) follows by the second ω-rule. □

Claim: The formula

$$\forall \mathbf{V} Y \left(\exists N \left(N \geq Y \wedge \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \right) \leftrightarrow \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, Y) \right) \tag{80}$$

is provable in HTC_2^ω.

Proof. Left-to-right: take any integer n and precomputed term r. By (36), the universal closure of (78) can be rewritten as

$$\forall \mathbf{V} \left(\overline{n} \geq r \wedge \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{n}) \to \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \right).$$

By the ω-rules, it follows that

$$\forall N Y \mathbf{V} \left(N \geq Y \wedge \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \to \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, Y) \right),$$

which is equivalent to the implication to be proved.

Right-to-left: We will show that

$$\forall \mathbf{V} \left(\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r) \to \exists N \left(N \geq r \wedge \mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N) \right) \right) \tag{81}$$

for every precomputed term r; then the implication to be proved will follow by the second ω-rule. Since the set of numerals is contiguous, three cases are possible: (1) $r < \overline{n}$ for all integers n; (2) r is a numeral; (3) $r > \overline{n}$ for all integers n. In the last case, the antecedent of (81) is equivalent to $\perp$ by (36). Otherwise, assume $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, r)$; we need to find N such that $N \geq r$ and $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, N)$. If $r < \overline{n}$ for all n then take $N = \overline{0}$; $\mathit{Atleast}_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, \overline{0})$ follows from (36). If r is a numeral then take $N = r$. □

In Section 9.1 we showed that formula (52) is provable in HTC. The axioms D_1 are not used in that proof. It follows that formula (52) is derivable in HTC_2^ω from D_0.

Claim: The sentence

$$\forall \mathbf{V} N \left(\exists \mathbf{X} Start_F^{\mathbf{X},\mathbf{V}}(\mathbf{X},\mathbf{V},N) \leftrightarrow Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},N) \right) \tag{82}$$

is derivable in HTC_2^ω from D_0.

Proof. For every integer n, the sentence

$$\forall \mathbf{V}(\exists \mathbf{X} Start_F^{\mathbf{X},\mathbf{V}}(\mathbf{X},\mathbf{V},\overline{n}) \leftrightarrow Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},\overline{n}))$$

is derivable in HTC_2^ω from D_0, because it follows from (36) and (52). Then (82) follows by the second ω-rule. □

15.3 Proof of Lemma 8

To prove Lemma 8, we need to show that all instances of D_1 can be derived in HTC_2^ω from D_0.

Proof of (8). By (80), $Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},Y)$ is equivalent to

$$\exists N(N \geq Y \wedge Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},N)).$$

By (82), this formula is equivalent to

$$\exists N(N \geq Y \wedge \exists \mathbf{X} Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N)),$$

which can be further rewritten as

$$\exists \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N) \wedge N \geq Y).$$

Proof of (9). We will prove the equivalence

$$Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V},Y) \leftrightarrow \forall \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X},\mathbf{V},N) \rightarrow N \leq Y) \tag{83}$$

by cases, using the *Std* axiom

$$Y < \overline{0} \vee \forall M(M < Y) \vee \exists M(M = Y \wedge M \geq \overline{0}).$$

Case 1: $Y < \overline{0}$. By *Defs*, the left-hand side of (83) is equivalent to $\bot$. Furthermore, by D_0, $Start_F^{\mathbf{X},\mathbf{V}}(\mathbf{U},\mathbf{V},0)$ and thus the right-hand side of (83) is equivalent to $\bot$ as well.

Case 2: $\forall M(M < Y)$. By *Defs*, the left-hand side of (83) is equivalent to $\top$. The right-hand side is equivalent to $\top$ as well.

Case 3: $M = Y$ and $M \geq 0$. Formula (83) can be rewritten as

$$Atmost_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, M) \leftrightarrow \forall \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow N \leq M). \tag{84}$$

By (79), the left-hand side is equivalent to $\neg Atleast_F^{\mathbf{X};\mathbf{V}}(\mathbf{V}, M + \overline{1})$. Hence, by (8), it is equivalent to

$$\neg \exists \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N + \overline{1}) \wedge N + \overline{1} \geq M + \overline{1})$$

and furthermore to

$$\neg \exists \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \wedge N \geq M + \overline{1}).$$

This formula can be further rewritten as

$$\forall \mathbf{X} N(Start_F^{\mathbf{X};\mathbf{V}}(\mathbf{X}, \mathbf{V}, N) \rightarrow \neg(N \geq M + \overline{1})),$$

which is equivalent to the right-hand side of (84).

15.4 Proof of Lemma 9

Let F be a sentence over the signature σ_1 that is derivable in HTC_2^ω from D_0. We will show that every standard HT-interpretation of σ_1 satisfies F; then the provability of F in HTC^ω will follow from the completeness of HTC^ω (Section 12).

Consider a standard HT-interpretation $\langle \mathcal{H}, I \rangle$ of σ_1. Let I' be the extension of I to the signature σ_2 defined by the condition: an extended precomputed atom $Start_F^{\mathbf{X},\mathbf{V}}(\mathbf{x}, \mathbf{v}, \overline{n})$ is satisfied by I' iff $n \leq 0$ or

- $n > 0$,
- $I \models F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$, and
- there exist at least n tuples $\mathbf{y}$ of precomputed terms such that $\mathbf{y} \geq \mathbf{x}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

Since $\langle \mathcal{H}, I \rangle$ is standard, I' is an ω-iterpretation. Furthermore, let $\mathcal{H}'$ be the set of extended precomputed atoms obtained from $\mathcal{H}$ by adding the atoms $Start_F^{\mathbf{X},\mathbf{V}}(\mathbf{x}, \mathbf{v}, \overline{n})$ such that $n \leq 0$ or

- $n > 0$,
- $\langle \mathcal{H}, I \rangle \models_{ht} F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$, and
- there exist at least n tuples $\mathbf{y}$ of precomputed terms such that $\mathbf{y} \geq \mathbf{x}$ and $\langle \mathcal{H}, I \rangle \models_{ht} F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

From the persistence property of HT-interpretations (Section 10) we can conclude that each of the atoms added to $\mathcal{H}$ is satisfied by I'. Hence $\langle \mathcal{H}', I' \rangle$ is an HT-interpretation of σ_2.

We will show that

$$\langle \mathcal{H}', I' \rangle \text{ satisfies } \mathit{Std},\ \mathit{Defs}, \text{ and } D_0. \tag{85}$$

Then the assertion of the lemma will follow. Indeed, the deductive system HTC_2^ω can be described as $SQHT^\omega$ (see Section 12) over σ_2 extended by the axioms *Std* and *Defs*. Hence F is derivable in $SQHT^\omega$ over σ_2 from *Std*, *Defs* and D_0. By (85), $\langle \mathcal{H}', I' \rangle$ is an ω-model of these sentences. By the soundness of $SQHT^\omega$ [Fandinno and Lifschitz, 2023a, Theorem 4], it follows that F is satisfied by $\langle \mathcal{H}', I' \rangle$. Since F is a sentence over σ_1, we conclude that F is satisfied by $\langle \mathcal{H}, I \rangle$.

Proof of (85):

For *Defs* and *Std*, this assertion follows from the fact that $\langle \mathcal{H}', I' \rangle$ extends the interpretation $\langle \mathcal{H}, I \rangle$ of σ_1, which satisfies *Defs* and *Std* because it is standard.

For D_0, consider the more difficult axiom schema in this group, the last one. We need to check that for any tuples $\mathbf{x}$ and $\mathbf{v}$ of precomputed terms and any positive n, $\langle \mathcal{H}', I' \rangle$ satisfies

$$\mathit{Start}_F^{\mathbf{X};\mathbf{V}}(\mathbf{x}, \mathbf{v}, \overline{n+1}) \leftrightarrow F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}} \wedge \exists \mathbf{U}(\mathbf{x} < \mathbf{U} \wedge \mathit{Start}_F^{\mathbf{X},\mathbf{V}}(\mathbf{U}, \mathbf{v}, \overline{n})). \tag{86}$$

We need to check, in other words, that I' satisfies the left-hand side of (86) iff I' satisfies the right-hand side, and similarly for $\langle \mathcal{H}', I' \rangle$.

Assume that I' satisfies the left-hand side. Then $I \models F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$, and

there exist at least $n+1$ tuples $\mathbf{y}$ such that $\mathbf{y} \geq \mathbf{x}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

It follows that

there exist at least n tuples $\mathbf{y}$ such that $\mathbf{y} > \mathbf{x}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

Pick such a group of n tuples, and let $\mathbf{u}$ be the least among them. Then $\mathbf{u} > \mathbf{x}$, and

there exist at least n tuples $\mathbf{y}$ such that $\mathbf{y} \geq \mathbf{u}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

It follows that I satisfies $\mathbf{x} < \mathbf{u} \wedge \mathit{Start}_F^{\mathbf{X},\mathbf{V}}(\mathbf{u}, \mathbf{v}, \overline{n})$, and consequently satisfies the right-hand side of (86).

Assume now that I' satisfies the right-hand side of (86). Then $I' \models F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$, and there exists a tuple $\mathbf{u}$ such that $\mathbf{x} < \mathbf{u}$ and $I' \models \mathit{Start}_F^{\mathbf{X},\mathbf{V}}(\mathbf{u}, \mathbf{v}, \overline{n})$. Hence $I \models F_{\mathbf{x},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$, and

there exist at least n tuples $\mathbf{y}$ such that $\mathbf{y} \geq \mathbf{u}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$.

It follows that

there exist at least $n+1$ tuples $\mathbf{y}$ such that $\mathbf{y} \geq \mathbf{x}$ and $I \models F_{\mathbf{y},\mathbf{v}}^{\mathbf{X},\mathbf{V}}$,

so that I' satisfies the left-hand side of (86).

For the HT-interpretation $\langle \mathcal{H}, I \rangle$ the reasoning is similar.

16 Conclusion

In this paper we argue that strong equivalence of two programs with counting can be established, in many cases, by proving the equivalence of the corresponding first-order sentences in the deductive system HTC. We do not know if HTC is complete for strong equivalence, that is to say, if $\tau^*\Pi_1 \leftrightarrow \tau^*\Pi_2$ is provable in HTC for all pairs Π_1, Π_2 of strongly equivalent MGC programs. But the deductive system HTC^ω, which contains infinitary rules, is shown to be complete in this sense.

Sentences F_1, F_2 are equivalent in HTC if and only if the sentences γF_1, γF_2 are equivalent in the classical first-order theory HTC'. This fact suggests that it may be possible to use theorem provers for classical theories, such as VAMPIRE [Kovaćs and Voronkov, 2013], to verify strong equivalence of MGC programs. Extending the proof assistant ANTHEM [Fandinno *et al.*, 2020, Heuer, 2020] in this direction is a topic for future work.

A translation similar to τ^* is used in ANTHEM to verify another kind of equivalence of mini-GRINGO programs—equivalence with respect to a user guide [Fandinno *et al.*, 2023, Hansen, 2023]. We plan to extend work on user guides to programs with counting.

Finally, we would like to investigate the possibility of extending the deductive systems described in this paper to aggregates other than counting.

References

[Chen *et al.*, 2005] Yin Chen, Fangzhen Lin, and Lei Li. SELP — a system for studying strong equivalence between logic programs. In *Proceedings of International Conference on Logic Programming and Nonmonotonic Reasoning*, pages 442–446, 2005.

[Fandinno and Hansen, 2023] Jorge Fandinno and Zachary Hansen. Recursive aggregates as intensional functions. In *Proceedings of the Workshops colocated with the 39th International Conference on Logic Programming*, 2023.

[Fandinno and Lifschitz, 2023a] Jorge Fandinno and Vladimir Lifschitz. Omega-completeness of the logic of here-and-there and strong equivalence of logic programs. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning*, 2023.

[Fandinno and Lifschitz, 2023b] Jorge Fandinno and Vladimir Lifschitz. On Heuer's procedure for verifying strong equivalence. In *Proceedings of European Conference on Logics in Artificial Intelligence*, 2023.

[Fandinno *et al.*, 2020] Jorge Fandinno, Vladimir Lifschitz, Patrick Lühne, and Torsten Schaub. Verifying tight logic programs with Anthem and Vampire. *Theory and Practice of Logic Programming*, 20, 2020.

[Fandinno *et al.*, 2022] Jorge Fandinno, Zach Hansen, and Yuliya Lierler. Axiomatization of aggregates in answer set programming. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[Fandinno *et al.*, 2023] Jorge Fandinno, Zachary Hansen, Yuliya Lierler, Vladimir Lifschitz, and Nathan Temple. External behavior of a logic program and verification of refactoring. *Theory and Practice of Logic Programming*, 2023.

[Fandinno *et al.*, 2024] Jorge Fandinno, Vladimir Lifschitz, and Nathan Temple. Locally tight programs. *Theory and Practice of Logic Programming*, 2024.

[Ferraris *et al.*, 2011] Paolo Ferraris, Joohyung Lee, and Vladimir Lifschitz. Stable models and circumscription. *Artificial Intelligence*, 175:236–263, 2011.

[Gebser *et al.*, 2019] Martin Gebser, Roland Kaminski, Benjamin Kaufmann, Marius Lindauer, Max Ostrowski, Javier Romero, Torsten Schaub, and Sven Thiele. Potassco User Guide. Available at `https://github.com/potassco/guide/releases/`, 2019.

[Hansen, 2023] Zachary Hansen. Anthem-p2p: Automatically verifying the equivalent external behavior of ASP programs. In *Electronic Proceedings in Theoretical Computer Science*, volume 385, 2023.

[Harrison *et al.*, 2017] Amelia Harrison, Vladimir Lifschitz, David Pearce, and Agustin Valverde. Infinitary equilibrium logic and strongly equivalent logic programs. *Artificial Intelligence*, 246:22–33, May 2017.

[Henkin, 1954] Leon Henkin. A generalization of the concept of ω-consistency. *The Journal of Symbolic Logic*, 19:183–196, 1954.

[Heuer, 2020] Jan Heuer. Automated verification of equivalence properties in advanced logic programs. Bachelor Thesis, University of Potsdam, 2020.

[Hosoi, 1966] Tsutomu Hosoi. The axiomatization of the intermediate propositional systems S_n of Gödel. *Journal of the Faculty of Science of the University of Tokyo*, 13:183–187, 1966.

[Kovaćs and Voronkov, 2013] Laura Kovaćs and Andrei Voronkov. First-order theorem proving and Vampire. In *International Conference on Computer Aided Verification*, pages 1—35, 2013.

[Lifschitz *et al.*, 2001] Vladimir Lifschitz, David Pearce, and Agustin Valverde. Strongly equivalent logic programs. *ACM Transactions on Computational Logic*, 2:526–541, 2001.

[Lifschitz *et al.*, 2007] Vladimir Lifschitz, David Pearce, and Agustin Valverde. A characterization of strong equivalence for logic programs with variables. In *Procedings of International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, pages 188–200, 2007.

[Lifschitz *et al.*, 2008] Vladimir Lifschitz, Leora Morgenstern, and David Plaisted. Knowledge representation and classical logic. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*, pages 3–88. Elsevier, 2008.

[Lifschitz *et al.*, 2019] Vladimir Lifschitz, Patrick Lühne, and Torsten Schaub. Verifying strong equivalence of programs in the input language of gringo. In *Proceedings of the 15th International Conference on Logic Programming and Non-monotonic Reasoning*, 2019.

[Lifschitz, 2021] Vladimir Lifschitz. Here and there with arithmetic. *Theory and Practice of Logic Programming*, 2021.

[Lifschitz, 2022] Vladimir Lifschitz. Strong equivalence of logic programs with counting. *Theory and Practice of Logic Programming*, 22, 2022.

[Lin, 2002] Fangzhen Lin. Reducing strong equivalence of logic programs to entailment in classical propositional logic. In *Proceedings of International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 170–176, 2002.

[Pearce *et al.*, 2009] David Pearce, Hans Tompits, and Stefan Woltran. Characterising equilibrium logic and nested logic programs: Reductions and complexity. *Theory and Practice of Logic Programming*, 9:565–616, 2009.

[Pearce, 1997] David Pearce. A new logical characterization of stable models and answer sets. In Jürgen Dix, Luis Pereira, and Teodor Przymusinski, editors, *Non-Monotonic Extensions of Logic Programming (Lecture Notes in Artificial Intelligence 1216)*, pages 57–70. Springer, 1997.

[Pearce, 1999] David Pearce. From here to there: Stable negation in logic programming. In D. Gabbay and H. Wansing, editors, *What Is Negation?* Kluwer, 1999.

[Truszczynski, 2012] Miroslaw Truszczynski. Connecting first-order ASP and the logic FO(ID) through reducts. In Esra Erdem, Joohyung Lee, Yuliya Lierler, and David Pearce, editors, *Correct Reasoning: Essays on Logic-Based AI in Honor of Vladimir Lifschitz*, pages 543–559. Springer, 2012.