For each FOL-sentence ϕ we have:

$$\begin{array}{ll} & \phi \notin \text{FOL-SAT-FIN} \\ \text{iff} & \mathcal{A} \not\models \phi \quad \text{for all finite structures } \mathcal{A} \\ \text{iff} & \mathcal{A} \models \neg\phi \text{ for all finite structures } \mathcal{A} \\ \text{iff} & \neg\phi \in \text{FOL-VALID-FIN} \end{array}$$

Hence, if FOL-VALID-FIN would be recursively enumerable then so would be the complement of FOL-SAT-FIN. This, however, is not the case, since otherwise FOL-SAT-FIN and its complement are recursively enumerable, and thus, decidable. Thus, we have shown:

Theorem 1.3.7 (Trakhtenbrot's Theorem). FOL-VALID-FIN *is not recursively enumerable.*

As we mentioned before, the proofs of Trakhtenbrot's Theorem and the undecidability result for the validity problem of FOL could be provided by a single binary predicate symbol. Formally, this means that we have to rewrite our proof by mimicking the 4-ary predicate symbol P and the function symbols f_a and the other symbols used in the proof of Lemma 1.3.5 with a single binary predicate symbol. Without going into detail, let us mention that the relevant feature of the simulation of a Turing machine in a logical framework is that we have to encode a *two-dimensional unbounded grid*: one dimension serves to encode the *time*, the other dimension to represent the *configurations*. This gives a rough idea why a single binary predicate symbol (representing the grid) suffices for Trakhtenbrot's Theorem. Vice versa, any logic that allows to encode a two-dimensional unbounded grid likely has the power of Turing machines, in which case the standard decision problems (satisfiability, validity, consequence and equivalence problem) are undecidable.

The following two subsections present decidable fragments of FOL over relational vocabularies. The first one is monadic first-order logic where only unary predicate symbols are allowed (Section 1.3.2). The second one is FOL in a special prenex form (Section 1.3.3).

1.3.2 Monadic first-order logic

The undecidability proof for FOL does not work anymore for relational vocabularies where all predicate symbols have arity 1. Then, a two-dimensional grid cannot be encoded and our machinery to simulate Turing machines breaks down. In fact, decidability of FOL can be achieved for vocabularies that only have *monadic* (i.e., unary) predicate symbols.

Monadic first-order logic (MFO) is FOL over purely relational vocabularies (i.e., vocabularies that do not have any function symbols) where all predicate symbols have arity one. Thus, terms of MFO are just variables. Constant symbols could have been added, but are omitted here. (See Remark 1.3.9 on page 43.) We consider here MFO without equality. MFO with equality will be addressed in the exercises. The abstract syntax of MFO-formulas without equality is:

$$\phi \ ::= \ \textit{true} \ \big| \ P(x) \ \big| \ \phi_1 \wedge \phi_2 \ \big| \ \neg\phi \ \big| \ \forall x.\phi$$

where $x \in \mathit{Var}$ and P is an unary predicate symbol. For the remainder of Subsection 1.3.2, we will briefly say MFO rather than MFO without equality.

The proof of the decidability of, e.g., the satisfiability problem for MFO-formulas is fairly simple and uses the finite model property of MFO. Let ϕ be a MFO-formula with k predicate symbols $P_1, \ldots, P_k$. An algorithm that checks satisfiability or validity of ϕ can be derived from the observation that the only way to distinguish the elements of the domain of a given MFO-structure is by the atomic formulas $P_i(x)$. Hence, all elements that belong exactly to the same P_i's can be identified, or simply collapsed into a single element. But then it remains a finite structure with at most 2^k elements and the truth value can be computed by evaluating all subformulas. More precisely:

Lemma 1.3.8 (Finite model property for MFO). *If ϕ is a satisfiable MFO-formula (without equality) with k predicate symbols then ϕ has a model where the domain is a subset of $\{0,1\}^k$.*

Proof. Let $(\mathcal{A}, \mathcal{V})$ be a model for ϕ where $\mathcal{A} = (A, P_1^{\mathcal{A}}, \ldots, P_k^{\mathcal{A}})$. Given a bit-string $(b_1, \ldots, b_k) \in \{0,1\}^k$ of length k, we define

$$A_{(b_1,\ldots,b_k)} \stackrel{\text{def}}{=} \bigcap_{\substack{1 \leqslant i \leqslant k \\ b_i = 1}} P_i^{\mathcal{A}} \cap \bigcap_{\substack{1 \leqslant i \leqslant k \\ b_i = 0}} A \setminus P_i^{\mathcal{A}}$$

The sets $A_{(b_1,\ldots,b_k)}$ are pairwise disjoint and provide a partition of A. Let $h : A \to \{0,1\}^k$ be defined by

$$h(a) \stackrel{\text{def}}{=} \text{unique bit-string } (b_1, \ldots, b_k) \in \{0,1\}^k \text{ with } a \in A_{(b_1,\ldots,b_k)}.$$

Let $\mathcal{B} = (B, P_1^{\mathcal{B}}, \ldots, P_k^{\mathcal{B}})$ be the structure where

$$\begin{aligned} B &= \big\{(b_1, \ldots b_k) \in \{0,1\}^k : A_{(b_1,\ldots,b_k)} \neq \varnothing\big\} \\ P_i^{\mathcal{B}} &= \big\{(b_1, \ldots b_k) \in B : b_i = 1\big\} \quad \text{for } i = 1, \ldots, k \end{aligned}$$

Then, $h(A) = B$ and $h(P_i^{\mathcal{A}}) = P_i^{\mathcal{B}}$ for $1 \leqslant i \leqslant k$. Hence, h is a surjective homomorphism from structure $\mathcal{A}$ to $\mathcal{B}$ (see page 11). As $(\mathcal{A}, \mathcal{V}) \models \phi$ we get $(\mathcal{B}, h \circ \mathcal{V}) \models \phi$. Hence, $(\mathcal{B}, h \circ \mathcal{V})$ is a model for ϕ where the domain is contained in $\{0,1\}^k$. □

Remark 1.3.9 (MFO with constant symbols). When replacing the constant symbols with unary predicate symbols as explained on page 13, the above lemma yields the bound $2^{k+\ell}$ for smallest models of satisfiable MFO-formulas ϕ with k unary predicate symbols and ℓ constant symbols. However, satisfiable MFO-formulas with k predicate symbols over relational vocabularies even have models with 2^k or fewer elements. The upper bound 2^k for smallest models is obtained by applying exactly the same argument as in the proof of Lemma 1.3.8. Again we start with an arbitrary model $(\mathcal{A}, \mathcal{V})$ for ϕ and define the function $h : A \to \{0,1\}^k$ as before. The definition of structure $\mathcal{B}$ has to be extended by appropriate meanings for the constant symbols appearing in ϕ, namely $c^{\mathcal{B}} = h(c^{\mathcal{A}})$. As before, h is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$. Thus, $(\mathcal{A}, \mathcal{V})$ and $(\mathcal{B}, h \circ \mathcal{V})$ assign the same truth values to all FOL-formulas without equality. This yields $(\mathcal{B}, h \circ \mathcal{V}) \models \Phi$. ■

Lemma 1.3.8 implies that the satisfiability problem for MFO (i.e., the question whether ϕ is satisfiable) can be solved by considering all interpretations $\mathcal{I} = (\mathcal{B}, \mathcal{V})$ where the domain of $\mathcal{B}$ is contained in $\{0,1\}^k$ and that provide meanings for the predicate symbols that appear in ϕ and the variables $x \in \mathit{Free}(\phi)$ (note that there are only finitely many such interpretations) and evaluating the truth value of ϕ under $\mathcal{I}$. The evaluation of ϕ under an interpretation $\mathcal{I} = (\mathcal{B}, \mathcal{V})$ with a finite structure $\mathcal{B}$ can be performed by a recursive computation of the truth values of the subformulas of ϕ under $\mathcal{B}$ with an appropriate valuation that extends $\mathcal{V}$, according to the recursive definition of the satisfaction relation $\models$. Clearly, if one of these interpretations is a model for ϕ then ϕ is satisfiable. By Lemma 1.3.8, if ϕ holds under none of these interpretations then ϕ is not satisfiable.

Each deterministic algorithm that solves the satisfiability problem yields at the same time a procedure to check validity (as ϕ is valid iff $\neg\phi$ is not satisfiable), logical implication (as $\phi \Vdash \psi$ iff $\phi \rightarrow \psi$ is valid) and equivalence (as $\phi \equiv \psi$ iff $\phi \Vdash \psi$ and $\psi \Vdash \phi$). Hence, we get:

Corollary 1.3.10 (Decidability of MFO). *The satisfiability problem for MFO (without equality) is decidable. The same holds for the validity, consequence and equivalence problem for MFO (without equality).*

Although the above satisfiability checking algorithm for MFO is rather naïve, there are no efficient algorithms for the MFO-decision problems. We mention without proof that the satisfiability problem for MFO is complete for the complexity class *NEXPTIME*.

1.3.3 Decidability of Schönfinkel-Bernays formulas

There are also decidable fragments of FOL that permit predicate symbols of arity two or more. We consider here *Schönfinkel-Bernays formulas* which rely on a relational vocabulary *Voc* and are FOL-formulas without equality in prenex form where the quantifier prefix consists of a sequence of existential quantifiers, followed by a sequence of universal quantifiers and the matrix.

Definition 1.3.11 (Schönfinkel-Bernays formula, $\exists^*\forall^*$-formula). A Schönfinkel-Bernays formula is a FOL-formula of the form

$$\exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m . \phi$$

where $n, m \geqslant 0$ and ϕ is a quantifier-free FOL-formula without equality over some relational vocabulary. Schönfinkel-Bernays formulas are also called $\exists^*\forall^*$-formulas. ■

For example, $\exists x \forall y . P(x, y)$ is a satisfiable Schönfinkel-Bernays formula, while

$$\forall y \forall z . (\, P(y, y) \wedge \neg P(c, z) \,)$$

is an unsatisfiable Schönfinkel-Bernays formula, where c is a constant symbol and P a binary predicate symbol. In the sequel, we show that the question whether a given Schönfinkel-Bernays sentence is satisfiable is decidable. As for MFO, the crucial observation to establish decidability is the finite model property. For Schönfinkel-Bernays sentences we can even establish the *small model property* stating that each satisfiable formula has a model where the number of elements in the domain is linear in the length of the formula. More precisely:

Lemma 1.3.12 (Small model property). *Let ψ be a satisfiable Schönfinkel-Bernays sentence with n existential quantifiers and ℓ constant symbols. Then, ψ has a model with at most $n + \ell$ elements.*

Proof. Let $\psi = \exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m.\phi$ and let $c_1, \ldots, c_\ell$ be the constant symbols that appear in ψ. Let $\mathcal{A}$ a model for ψ and A the domain of $\mathcal{A}$. We now define a substructure $\mathcal{B}$ of $\mathcal{A}$ that yields a model for ψ and has at most $n+\ell$ elements. As $\mathcal{A} \models \psi$ there exists elements $a_1, \ldots, a_n \in A$ such that

$$(\mathcal{A}, [x_1 := a_1, \ldots, x_n := a_n]) \models \forall y_1 \ldots \forall y_m.\phi$$

We define the domain of $\mathcal{B}$ as the set consisting of these elements $a_1, \ldots, a_n$ and the interpretations of the constant symbols:

$$B \stackrel{\text{def}}{=} \{a_1, \ldots, a_n\} \cup \{c_1^{\mathcal{A}}, \ldots, c_\ell^{\mathcal{A}}\}$$

Clearly, $|B| \leqslant n+\ell$. Moreover, we put $c_i^{\mathcal{B}} \stackrel{\text{def}}{=} c_i^{\mathcal{A}}$, $i = 1, \ldots, \ell$, and $P^{\mathcal{B}} \stackrel{\text{def}}{=} P^{\mathcal{A}} \cap B^k$ for each k-ary predicate symbol P. This yields a substructure $\mathcal{B}$ of $\mathcal{A}$ with domain B. (Recall that ψ does not contain function symbols of arity one or more since the vocabulary is supposed to be relational.) Obviously, we have $\mathcal{B} \models \psi$. □

On the basis of the above lemma, satisfiability of a given Schönfinkel-Bernays sentence $\psi = \exists x_1 \ldots \exists x_n \forall y_1 \ldots \forall y_m.\phi$ with ℓ constant symbols $c_1, \ldots, c_\ell$ can be checked algorithmically by considering all structures with elements $\{1, \ldots, k\}$ for some $k \leqslant n+\ell$ and computing the truth value of ψ for this structure. If none of these structures is a model for ψ then ψ is not satisfiable. Hence:

Theorem 1.3.13. *The satisfiability problem for Schönfinkel-Bernays formulas is decidable.*

1.4 Axiomatizability of first-order theories

In the previous sections we considered decidability questions and deductive calculi for the standard semantics of FOL. This section is concerned with the decision problems and other properties of certain formula-sets. We will study the decidability and existence of Hilbert proof systems of the set of FOL-formulas ϕ that are true in some fixed structure $\mathcal{A}$. This formula-set is called the FO-theory of structure $\mathcal{A}$.

Throughout this section, we suppose a fixed vocabulary and variable-set, both being recursively enumerable.

1.4.1 First-order theories

Let us first recall some basic notations. If $\mathfrak{T}$ is a formula-set and ϕ a formula, then $\mathfrak{T} \Vdash \phi$ denotes that ϕ is a logical consequence of $\mathfrak{T}$, i.e., any model for $\mathfrak{T}$ is also a model for ϕ. Formula-set $\mathfrak{T}$ is said to be true in some structure if there exists a structure $\mathcal{A}$ such that all $\mathcal{A}$-interpretations $(\mathcal{A}, \mathcal{V})$ are models for $\mathfrak{T}$. The notation $\phi(x_1, \ldots, x_n)$, or briefly $\phi(\overline{x})$, indicates that $x_1, \ldots, x_n$ are pairwise distinct variables and ϕ is a FOL-formula with $\mathit{Free}(\phi) \subseteq \{x_1, \ldots, x_n\}$ where $\overline{x}$ denotes the n-tuple $(x_1, \ldots, x_n)$.

Definition 1.4.1 (First-order theory). Let $\mathfrak{T}$ be a set of FOL-formulas. $\mathfrak{T}$ is called *closed under logical consequences* if for all formulas ϕ: $\mathfrak{T} \Vdash \phi$ implies $\phi \in \mathfrak{T}$. $\mathfrak{T}$ is called an *FO-theory* if $\mathfrak{T}$ is true in some structure and closed under logical consequences. ■

In what follows, we often briefly say "theory" rather than "FO-theory". Obviously, each theory $\mathfrak{T}$ enjoys the following properties:

- $\mathfrak{T}$ contains all valid formulas, as they are logical consequences of any formula-set.
- For each FOL-formula ϕ at most one of the formulas ϕ and $\neg\phi$ belongs to $\mathfrak{T}$.
  (This holds for all satisfiable formula-sets.)

The set of all valid FOL-formulas (over some fixed vocabulary) constitute a FO-theory which is contained in each other FO-theory (over the same vocabulary). For each satisfiable set $\mathfrak{G}$ of FOL-formulas the *logical closure*

$$Cl(\mathfrak{G}) \stackrel{\text{def}}{=} \{\phi : \mathfrak{G} \Vdash \phi\}$$

of $\mathfrak{G}$ is a FO-theory, provided that $\mathfrak{G}$ is true in some structure. Note that (i) $Cl(\mathfrak{G})$ is true in some structure as any structure $\mathcal{A}$ which is a model for $\mathfrak{G}$ is also a model for its logical closure, and (ii) $Cl(\mathfrak{G}) \Vdash \phi$ implies $\mathfrak{G} \Vdash \phi$ and therefore $\phi \in Cl(\mathfrak{G})$. In particular, $Cl(\mathfrak{G})$ is a FO-theory for each satisfiable set of FOL-sentences.

Definition 1.4.2 (Completeness of a theory). A theory $\mathfrak{T}$ is called *complete* if it is maximal satisfiable over some structure, i.e., for all sentences ϕ: either $\phi \in \mathfrak{T}$ or $\neg\phi \in \mathfrak{T}$. ■

Note that Definition 1.4.2 ranges over all sentences, i.e., formulas without free variables, which leaves the possibility that $\phi \notin \mathfrak{T}$ and $\neg\phi \notin \mathfrak{T}$ for complete FO-theories and formulas ϕ that contain free variables.

Lemma 1.4.3. *Let $\mathfrak{T}$ be a complete FO-theory and $\phi(x_1,\ldots,x_n)$ a FOL-formula. Then:*

$$\phi(x_1,\ldots,x_n) \in \mathfrak{T} \quad \textit{if and only if} \quad \forall x_1 \ldots \forall x_n.\, \phi(x_1,\ldots,x_n) \in \mathfrak{T}$$

Proof. Let us use the tuple-notation $\overline{x}$ for $(x_1,\ldots,x_n)$. The implication "$\Longleftarrow$" holds for any FO-theory. The argument is as follows. Suppose $\forall\overline{x}.\phi(\overline{x}) \in \mathfrak{T}$. As $\forall\overline{x}.\phi(\overline{x}) \Vdash \phi(\overline{x})$ and $\mathfrak{T}$ is closed under logical consequences, we get $\phi(\overline{x}) \in \mathfrak{T}$.

To prove the implication "$\Longrightarrow$", we need the completeness of $\mathfrak{T}$. Suppose $\phi(\overline{x}) \in \mathfrak{T}$. Let us assume by contradiction that $\forall\overline{x}.\,\phi(\overline{x}) \notin \mathfrak{T}$. As $\forall\overline{x}.\,\phi(\overline{x})$ is a sentence and $\mathfrak{T}$ is complete:

$$\neg\forall\overline{x}.\,\phi(\overline{x}) \in \mathfrak{T}$$

As $\mathfrak{T}$ is true in some structure, we may pick some structure $\mathcal{A}$ with $\mathcal{A} \models \mathfrak{T}$. Then, $\mathcal{A} \models \psi$ for all formulas $\psi \in \mathfrak{T}$. In particular:

$$\mathcal{A} \models \neg\forall\overline{x}.\,\phi(\overline{x})$$

Furthermore, $\mathcal{A}$ is a model for $\phi(\overline{x})$. Hence, if A is the domain of $\mathcal{A}$ then:

$$\begin{array}{lll} & (\mathcal{A}, [\overline{x} := \overline{a}]) \models \phi(\overline{x}) & \text{for all } \overline{a} \in A^n \\ \Longrightarrow & \mathcal{A} \models \ \forall\overline{x}.\,\phi(\overline{x}) & \text{(semantics of } \forall) \\ \Longrightarrow & \mathcal{A} \not\models \neg\forall\overline{x}.\,\phi(\overline{x}) & \text{(semantics of } \neg) \end{array}$$

Contradiction. $\square$

The statement of Lemma 1.4.3 might be wrong for incomplete theories. For example, let P is a monadic predicate symbol and $\mathfrak{T}$ the closure of the atomic formula $P(x)$, i.e., $\mathfrak{T} = \big\{\phi : P(x) \Vdash \phi\big\}$. Then, $\mathfrak{T}$ is an incomplete FO-theory and we have $P(x) \in \mathfrak{T}$, while $\forall x.P(x) \notin \mathfrak{T}$.

Definition 1.4.4 (FO-theory of a structure). If $\mathcal{A}$ is a structure for some vocabulary *Voc* then the first-order theory of $\mathcal{A}$, denoted $Th_{FO}(\mathcal{A})$ or briefly $Th(\mathcal{A})$, defined by

$$Th(\mathcal{A}) \stackrel{\text{def}}{=} \big\{\phi : \mathcal{A} \models \phi\big\},$$

consists of all FOL-formulas ϕ over *Voc* that hold over $\mathcal{A}$. $\blacksquare$

Note that $Th(\mathcal{A})$, indeed, enjoys the conditions required for theories. Truth in some structure is obvious. Let us check the closure property under logical consequences. Let ϕ be a formula with $Th(\mathcal{A}) \Vdash \phi$. As $\mathcal{A}$ is a model for $Th(\mathcal{A})$, structure $\mathcal{A}$ is also a model for ϕ. But then $\phi \in Th(\mathcal{A})$.

Obviously, the theory of any structure $\mathcal{A}$ is complete since for each sentence ϕ: either $\mathcal{A} \models \phi$ or $\mathcal{A} \models \neg\phi$. We will show later that each complete theories is the theory of some structure. See Theorem 1.4.6 on page 48.

For finite vocabularies *Voc* with k predicate symbols and r function symbols, FO-theories induced by structures over *Voc* are typically written as

$$Th(A, P_1, \ldots, P_k, f_1, \ldots, f_r) \quad \text{or} \quad Th(A, P_1, \ldots, P_k, f_1, \ldots, f_r, =)$$

where A is the domain of the given structure, $P_1, \ldots, P_k$ are the predicates interpreting the predicate symbols of *Voc* and $f_1, \ldots, f_r$ are the functions (and constants) for the function symbols in *Voc*. The notation $Th(A, P_1, \ldots, P_k, f_1, \ldots, f_r, =)$ indicates that FOL-formulas with equality are considered, while $Th(A, P_1, \ldots, P_k, f_1, \ldots, f_r)$ denotes the FO-theory induced by $\mathcal{A}$ for FOL without equality.

Example 1.4.5 (FO-theory of arithmetic, FO-theory of the reals). The FO-theory of arithmetic $Th(\mathbb{N}, +, *, =)$ is the set of all FOL-formulas with equality over the vocabulary consisting of two binary function symbols $+$ and $*$ that hold in the structure $(\mathbb{N}, +, *)$. The notation $(\mathbb{N}, +, *)$ denotes the structure with the natural numbers as domain and which interpretes + by addition and * by multiplication. Although no predicate symbol for the natural order $<$ is included in the vocabulary, it is definable by:

$$z < y \stackrel{\text{def}}{=} z \neq y \wedge \exists u. z + u = y$$

Furthermore, we use $y = 0$ as an abbreviation for $\forall u. u + y = u$. It is easy to see that the derived formulas $z < y$ and $y = 0$ have the standard meaning in $(\mathbb{N}, +, *)$. Then, for example, the sentence

$$\forall x \forall y. \big(y = 0 \vee \exists q \exists r. (x = q * y + r \wedge r < y) \big)$$

is a FOL-sentence in $Th(\mathbb{N}, +, *, =)$. Let us check this. Suppose we are given two natural numbers x and y such that $y \neq 0$. Let

$$q \stackrel{\text{def}}{=} x \operatorname{div} y = \lfloor \tfrac{x}{y} \rfloor \quad \text{and} \quad r \stackrel{\text{def}}{=} x \bmod y = x - qy.$$

Then, $x = q * y + r$ and $r < y$. The sentence

$$\phi \;=\; \forall x \forall y. \big(x < y \rightarrow \exists z. (x < z \wedge z < y) \big)$$

does not hold in $(\mathbb{N}, +, *)$ as, e.g., for $x = 0$ and $y = 1$ there is no integer z strictly between x and y. However, sentence ϕ evaluates to true when we consider the structure $(\mathbb{R}, +, *)$, i.e., the FO-theory $Th(\mathbb{R}, +, *, =)$ which is often called FO-theory of *real closed fields* or *Tarski algebra*. Here, however, we need another definition of $z < y$. For instance, we may redefine $z < y$ by

$$z < y \;\stackrel{\text{def}}{=}\; z \neq y \wedge \exists u. z + (u * u) = y$$

This definition of $z < y$ yields the natural order on both structures $(\mathbb{N}, +, *)$ and $(\mathbb{R}, +, *)$. Then:

$$\phi \in Th(\mathbb{R}, +, *, =) \quad \text{and} \quad \phi \notin Th(\mathbb{N}, +, *, =)$$

Thus, ϕ is an example for a formula that holds in the FO-theory of the real closed fields, but not in the FO-theory of arithmetic. ■

For the following results, it is irrelevant whether FOL with or without equality is considered. We first show that the theories induced by structures are the only complete theories.

Theorem 1.4.6 (Characterization of complete theories). *Let $\mathfrak{T}$ be a FO-theory. Then, the following statements are equivalent:*

(a) $\mathfrak{T}$ is complete (i.e., for each sentence ϕ either $\phi \in \mathfrak{T}$ or $\neg\phi \in \mathfrak{T}$).

(b) $\mathfrak{T} = Th(\mathcal{A})$ for some structure $\mathcal{A}$.

(c) $\mathfrak{T} = Th(\mathcal{A})$ for all structures $\mathcal{A}$ such that $\mathcal{A} \models \mathfrak{T}$.

Proof. (c) $\Longrightarrow$ (b): obvious since $\mathfrak{T}$ is true in some structure.

(b) $\Longrightarrow$ (a): Suppose $Th(\mathcal{A}) = \mathfrak{T}$ for some structure $\mathcal{A}$. If ϕ is a sentence then either $\mathcal{A} \models \phi$ or $\mathcal{A} \models \neg\phi$, but not both. Hence, exactly one of the formulas ϕ or $\neg\phi$ belongs to $Th(\mathcal{A}) = \mathfrak{T}$.

(a) $\Longrightarrow$ (c): Let $\mathcal{A}$ be a structure such that $\mathcal{A} \models \mathfrak{T}$. Then, $(\mathcal{A}, \mathcal{V}) \models \mathfrak{T}$ for all valuations $\mathcal{V}$. We now show that $Th(\mathcal{A}) = \mathfrak{T}$. Since

$$\begin{array}{lll} \phi(\overline{x}) \in \mathfrak{T} & \text{iff} \quad \forall \overline{x}. \phi(\overline{x}) \in \mathfrak{T} & \text{(by Lemma 1.4.3)} \\ \mathcal{A} \models \phi(\overline{x}) & \text{iff} \quad \mathcal{A} \models \forall \overline{x}. \phi(\overline{x}) & \text{(by definition of } \mathcal{A} \models \ldots) \end{array}$$

it suffices to show that for each sentence ψ we have: $\psi \in Th(\mathcal{A})$ iff $\psi \in \mathfrak{T}$.

"$\Longrightarrow$": Let ψ be a sentence in $Th(\mathcal{A})$. Since $\mathfrak{T}$ is complete, we have $\psi \in \mathfrak{T}$ or $\neg\psi \in \mathfrak{T}$. As $\mathcal{A} \not\models \neg\psi$ and $\mathcal{A}$ is a model for $\mathfrak{T}$ we get $\psi \in \mathfrak{T}$.

"$\Longleftarrow$": As $\mathcal{A}$ is a model for $\mathfrak{T}$ we have $\mathfrak{T} \subseteq \{\psi : \mathcal{A} \models \psi\} = Th(\mathcal{A})$. Thus, $\psi \in \mathfrak{T}$ implies $\psi \in Th(\mathcal{A})$. □

Theorem 1.4.7 (Semi-decidablity and decidability of complete theories). *Let $\mathfrak{T}$ be a complete theory. Then, $\mathfrak{T}$ is recursively enumerable if and only if $\mathfrak{T}$ is decidable.*

Proof. If $\mathfrak{T}$ is decidable then $\mathfrak{T}$ is recursively enumerable (this holds for any set). Let us now suppose that $\mathfrak{T}$ is recursively enumerable. We describe a decision procedure for $\mathfrak{T}$. Let $\phi = \phi(x_1, \ldots, x_n)$ be a FOL-formula for which we want to check whether $\phi \in \mathfrak{T}$. By Lemma 1.4.3:

$$\phi \in \mathfrak{T} \quad \text{iff} \quad \psi \stackrel{\text{def}}{=} \forall x_1 \ldots \forall x_n . \phi \in \mathfrak{T}$$

We then call a procedure that generates a recursive enumeration $\theta_1, \theta_2, \theta_3, \ldots$ of $\mathfrak{T}$ and stops as soon as the generated formula θ_i is ψ or $\neg\psi$. If $\theta_i = \psi$ then the algorithm returns "yes", otherwise it returns "no". Since $\mathfrak{T}$ is complete and ψ is a sentence, we have: $\psi \in \mathfrak{T}$ or $\neg\psi \in \mathfrak{T}$. Thus, the above algorithm terminates with the correct answer for each input-formula. □

1.4.2 Axiomatizability and decidability

Definition 1.4.8 (Axiomatizable theories). A theory $\mathfrak{T}$ is said to be *axiomatizable* if there exists a decidable set $\mathfrak{F}$ of formulas such that $\mathfrak{T}$ agrees with the logical closure of $\mathfrak{F}$, i.e.,

$$\mathfrak{T} \; = \; \big\{ \phi : \mathfrak{F} \Vdash \phi \big\} \; = \; Cl(\mathfrak{F}).$$

In this case, $\mathfrak{F}$ is called an *axiomatization* for $\mathfrak{T}$. If $\mathfrak{F}$ is finite then $\mathfrak{T}$ is called *finitely axiomatizable*. ■

Note that if $\mathfrak{F}$ is a finite axiomatization of $\mathfrak{T}$ then there is even a single formula that yields an axiomatization for $\mathfrak{T}$: simply take the conjunction of the formulas in $\mathfrak{F}$.