Intermediate Logic

Lecture notes

Sandy Berkovski

Bilkent University
Fall 2010

Chapter 5

Predicate calculus: Syntax

In this chapter we begin investigating the properties of predicate calculus. We shall see that, while some concepts and results of propositional calculus straightforwardly carry over to the case of predicate calculus, still many other are significantly generalised or else are completely novel.

5.1 Quantifiers and variables

It is clear that the resources of propositional calculus are inadequate for identifying valid arguments in many areas of discourse. The inference 'If Socrates is male, then he is not female' appears to be a valid one. But all we can do within the limits of $\mathsf{H_s}$ is to symbolise it as 'If P, then Q', or perhaps as 'If P, then $\neg Q$', but in either case there is no reason why these arguments must be recognised as valid. The apparent validity of our inference derives from a link between Socrates, the property of being female, and the property of being male. That link remains inscrutable so far as we are confined to $\mathsf{H_s}$. Thus we must enrich our syntactic and semantic means to talk about objects and relations (recall that properties are unary relations). This will be achieved in predicate calculus.

Some of the elements of predicate calculus should be familiar from the introductory course. Here we start by drawing attention to just one specific issue of philosophical, rather than mathematical, significance—namely, the role of variables—and proceed straightaway to the formal exposition.

The true claim 'The number 9 is less than, or equal, or greater than 0' can be paraphrased as a disjunction:

$$(9 < 0) \vee (9 = 0) \vee (9 > 0).$$

To write this disjunction in our canonical propositional notation we may simply treat each disjunct as an atomic sentence denoting it by a sentence parameter. But consider another true statement, 'Every real number is less than, or equal to, or greater than 0'. Here an attempt to ape our disjunction:

$$(\text{Every real number} < 0) \vee (\text{Every real number} = 0) \vee (\text{Every real number} > 0)$$

fails, since this paraphrase is obviously false. Rather, we have to labour a bit, first representing the original statements as:

Whatever real number is selected, it is either less than, or equal to, or greater than 0.

Therefore:

$$\text{Whatever real number is selected, } ((\text{it} < 0) \vee (\text{it} = 0) \vee (\text{it} > 0)).$$

However, suppose we advance further and wish to make a claim previously made about 0 about every number. That will be another true claim, since, mathematically, there is nothing unique about 0 in this respect. Then we may say:

Whatever real number is selected, (every real number is either less than it, or equal to it, or greater than it).

But the parenthetical expression still demands a paraphrase. According to our previous strategy, we may expect the following:

$$\text{Whatever real number is selected, (whatever real number is selected, } ((\text{it} < \text{it}) \vee (\text{it} = \text{it}) \vee (\text{it} > \text{it}))).$$

This will not do: unless we separate between two selections—that is, two occurrences of 'whatever' in our statement—we will be making a false claim. To distinguish between them, we may resort to indexing. We shall attach the same indices to 'whatever' and to the locutions of 'it' fixed by 'whatever'. Accordingly:

$$\text{Whatever}_1 \text{ real number is selected, (whatever}_2 \text{ real number is selected, ((it}_2 < \text{it}_1) \vee (\text{it}_2 = \text{it}_1) \vee (\text{it}_2 > \text{it}_1))).$$

But instead of indexing it would be more convenient to distinguish the *scope* of the occurrence of 'whatever' by different letters. Thus, the latest paraphrase will take the form:

$$\text{Whatever real number } x \text{ is selected, (whatever real number } y \text{ is selected, } ((y < x) \vee (y = x) \vee (y > x))).$$

On the other hand, the locution 'whatever entity is selected' is synonymous with the locution 'for all entities'. We further abbreviate 'for all' as '$\forall$'. Yet the paraphrase:

$$\forall x(\forall y((y > x) \vee (y = x) \vee (y < x))).$$

would be too quick: we have omitted the provision made for real numbers. The remedy is not difficult: the claim 'For all Fs, $\cdots$' may be paraphrased as the claim 'For all x, if x is F, then $\cdots$'. By inserting the conditional we avoid making claims about any non-F. Hence the correct paraphrase would be as follows:

$$\forall x(x \text{ is a real number } \supset (\forall y(y \text{ is a real number } \supset ((y > x) \vee (y = x) \vee (y < x))))).$$

We shall have an opportunity to explore the properties of quantifiers in more detail later on. What is important right now is to notice that the variables as introduced by us have the sole role of cross-referencing the quantifier. The indexing tool we abandoned for the purpose of convenience displayed just that. In this way variables are distinguished from names, or 'individual constants'. This may seem obvious: the whole purpose of moving from the specific claims about 9 and 0 above was in eliminating the names for these numbers in our statements. None the less it is not uncommon to encounter claims, such as 'Take any number—say, 22', or 'Take any politician—say, Tony Blair'. The speaker may go on and predicate properties of the number 22 or Tony Blair. Thus, if the resulting claim has the form 'x is F', x would appear to be a proper name—either for 22, or Tony Blair. Such a practice would be *logically* fallacious for a number of reasons. The simplest fallacy, to cut the story short, is that the original intention of the speaker was to reason about properties of *every* number or politician, whereas 22 and Tony Blair may not share properties with every number or politician. In general, the use of variables allows us making claims about *arbitrary* entities, but it does not validate replacing those variables by names of specific entities. (In daily contexts, the speaker is likely to be interpreted as making an inductive inference: by examining the properties of Tony Blair and, say, Jacques Chirac, he leaps to a claim about politicians in general.)

5.2 First-order theory

We shall present the system $\mathsf{T_p}$ of predicate calculus. To begin with, the above discussion suggests a revision of our concept of signature. We formulate it in the most general way, and will simplify later.

Definition 5.1. Let Σ_v be a denumerable set of individual variables. Let Σ_{ip} be a denumerable set of individual parameters. Let Σ_c be a set of individual constants. Let Σ_p be a denumerable set containing i-ary predicate parameters for each $i \geq 0$. Let Σ_f be a denumerable set containing i-ary function parameters for each $i \geq 0$. The *signature* for $\mathsf{T_p}$ is the set $\Sigma = \Sigma_v \cup \Sigma_{ip} \cup \Sigma_c \cup \Sigma_p \cup \Sigma_f$.

Remark. The sets Σ_v, Σ_{ip}, Σ_c, Σ_p, and Σ_f are mutually disjoint.

Predicate parameters having the arity n should be thought as standing for the sentences in the object language that have n blanks. When we wish to identify the predicate, those blanks will be represented by circled numerals (see Exercises for an example). Consequently, an 0-ary predicate parameter will be nothing but a sentence parameter. Similarly, constants can be thought as 0-ary function parameters, as we have mentioned already whilst discussing tautologies and contradictions. On occasions, it is convenient to specify the arity of predicates (and functions) explicitly. For an i-ary predicate P_j we shall then write 'P_j^i'.

The sign $\forall$ will stand for the universal quantifier. The existential quantifier will be introduced as an abbreviation. We can now define terms and formulae of our theory as follows.

Definition 5.2. The set $\mathbf{T}_\Sigma$ of terms of the signature Σ for $\mathsf{T_p}$ is the smallest set of expressions determined as follows:

1. $\Sigma_v \cup \Sigma_{ip} \cup \Sigma_c \subseteq \mathbf{T}_\Sigma$;
2. If $t_1, \ldots, t_n \in \mathbf{T}_\Sigma$ and $f \in \Sigma_f$, then $f(t_1, \ldots, t_n) \in \mathbf{T}_\Sigma$.

Definition 5.3. The set $\mathbf{F}_\Sigma$ of formulae of the signature Σ for $\mathsf{T_p}$ is then determined by the following rules:

1. If $P \in \Sigma_p$ is an 0-ary predicate parameter, then $P \in \mathbf{F}_\Sigma$.
2. If $P \in \Sigma_p$ is an n-ary predicate parameter, where $n > 0$, and $t_1, \ldots, t_n \in \mathbf{T}_\Sigma$, then $Pt_1 \cdots t_n \in \mathbf{F}_\Sigma$.
3. If $A \in \mathbf{F}_\Sigma$, then $\neg A \in \mathbf{F}_\Sigma$.
4. If $A, B \in \mathbf{F}_\Sigma$, then $(A \supset B) \in \mathbf{F}_\Sigma$.
5. If $A \in \mathbf{F}_\Sigma$, $x \in \Sigma_v$, then $\forall x A \in \mathbf{F}_\Sigma$.

Remark. Unless explicitly indicated to the contrary, we shall assume that $\Sigma_f = \Sigma_c = \varnothing$. Note also that each n-ary function $f\colon X \to Y$ can be 'represented' by an $(n+1)$-ary predicate as follows:

$$f(x_1, \ldots, x_n) = y \iff P(x_1, \ldots, x_n, y),$$

where $x_1, \ldots, x_n \in X$, $y \in Y$. This will allow us talking about various mathematical structures without using function parameters.

Definition 5.4. Let $\mathbf{Sm}_\Sigma = \Sigma \cup \{\neg, \supset, \forall, (,)\}$. Then $\mathbf{Sm}_\Sigma$ is the set of *symbols* of the signature Σ.

An *expression* of Σ is any string of elements of $\mathbf{Sm}_\Sigma$. An *atomic* formula is a formula containing no logical connectives or quantifiers. Similarly to the propositional case, we can now give an interpretation of the complexity of formulae.

Definition 5.5. By the degree $d(A)$ of the formula A we understand the number of occurrences of logical connectives and quantifiers in A, with every atomic formula assigned the degree 0.

Definition 5.6. For every $A \in \mathbf{F}_\Sigma$, $x \in \Sigma_v$, and $a \in \Sigma_{ip}$ we define the formula $A^{x/a}$ as follows:

1. If A is atomic, then $A^{x/a}$ obtains by substituting a for every occurrence of x in A.
2. $(A \supset B)^{x/a} = A^{x/a} \supset B^{x/a}$.
3. $(\neg A)^{x/a} = \neg A^{x/a}$.
4. $(\forall x A)^{x/a} = \forall x A^{x/a}$.
5. $(\forall y A)^{x/a} = \forall y A^{x/a}$.

A *closed* formula, or a *sentence*, is a formula A such that, for every $a \in \Sigma_{ip}$ and every $x \in \Sigma_v$, $A^{x/a} = A$.

The notion of substitution may be easier to understand with the aid of the notion of free and bound occurrences of variables. The *scope* of an occurrence of a quantified variable, *i.e.* a variable immediately preceded by a quantifier, is the smallest formula following that occurrence.

Example 5.7. $(\forall x P x) \supset (\forall x (Qxy \supset Ry))$. Here we identify two occurrences of the universal quantifier...

The variable x has a *bound* occurrence in the formula A if it either falls within the scope of an occurrence of the quantifier in A, or else is immediately preceded by a quantifier. The variable x has a *free* occurrence if it is not bound. Then we regard the formula $A^{x/a}$ as a result of substituting a for every free occurrence of x. Equivalently, we can say that x has a free occurrence in A if $A^{x/a}$ is not the same formula as $\neq A$.

Example 5.8. If A is the formula $\forall x P x \supset \forall y Q x y$, then $A^{x/a} = \forall x P x \supset \forall y Q a y$.

Definition 5.9. '$\exists x A^{x/a}$' $\iff$ '$\neg \forall x \neg A^{x/a}$'.

Remark. Metatheorems on deducibility and the deduction theorem are proved for the Hilbert-type axiomatisation of $\mathsf{T_p}$ in exactly the same way they are proved for $\mathsf{H_s}$.

[Qui51, Bos97, Men64]

Chapter 6

Predicate calculus: Semantics

6.1 Models and satisfiability

We shall now introduce several key notions of the predicate calculus. We start from afar by introducing a very general notion of algebraic system. Intuitively, an algebraic system is a set of objects, the elements of which we use in interpreting predicate parameters or functional parameters.

Definition 6.1. An *algebraic system* is a triple $\mathfrak{M} = \langle M, \Omega_F, \Omega_P \rangle$, where M is a non-empty set of individuals, Ω_F is a set of operations on M, and Ω_P is a set of predicates on M. If $\Omega_P = \varnothing$, then $\mathfrak{M}$ is called an *algebra*. If $\Omega_F = \varnothing$, then $\mathfrak{M}$ is called a *model*.

Definition 6.2. The set M of the system $\mathfrak{M} = \langle M, \Omega_F, \Omega_P \rangle$ is the *domain* of $\mathfrak{M}$. Ω_F is the *operator domain*. Ω_P is the *predicate domain*. The cardinality of $\mathfrak{M}$ is the number $|M|$.

Suppose, however, that we have to make claims about the elements of M, Ω_F, and Ω_P. We shall need a language for making those claims. Thus, an alternative formulation of the notion of algebraic system, albeit a less intuitive one, highlights the role of the signature and the distinction between object-language and metalanguage. Generally speaking, the signature may be finite.

Definition 6.3. An *algebraic system* of the countable signature Σ is a pair $\mathfrak{M} = \langle M, I \rangle$, where M is a non-empty set of individuals, and I is a mapping defined on M with the following conditions:

1. For every n-ary predicate parameter $P \in \Sigma_p$, $I(P) \subseteq M^n$;
2. For every n-ary function parameter $f \in \Sigma_f$, $I(f)\colon M^n \to M$;
3. For every $c \in \Sigma_c$, $I(c) \in M$.

We may explicitly distinguish between symbols (linguistic items) and their interpretations in $\mathfrak{M}$ by writing, for each symbol s, '$\overline{s}$' to denote $I(s)$. Sometimes it may be more convenient to replace I with the list of its values and to write $\mathfrak{M} = \langle M; \overline{P_1}, \ldots; \overline{f_1}, \ldots; \overline{c_1}, \ldots \rangle$. This notation shows also the equivalence of our two definitions of algebraic system. Moreover, let the arity of $\overline{P_i}$ and $\overline{f_i}$ be denoted by $n(\overline{P_i})$ and $n(\overline{f_i})$ respectively. Then the system $\mathfrak{M}$ is said to be of the *type* $\langle n(\overline{P_1}), \ldots; n(\overline{f_1}), \ldots \rangle$.

Example 6.4. The system $\mathfrak{Z} = \langle Z; +; \leq \rangle$ has the set of integers as the domain, the binary operation of addition in the set Ω_F and the binary predicate '① is less or equal to ②' in the set Ω_P. $\mathfrak{M}$ is of the type $\langle 2; 2 \rangle$.

Remark. In accordance with our earlier remark on representing n-ary functions with $(n+1)$-ary predicates, it is possible to transform algebras (or generally, algebraic systems) into models 'representing' them.

Here we shall only be interested in models. Given a model, we can define a valuation V for sentences of Σ.

Definition 6.5. Let $\mathbf{S}_\Sigma$ be the set of all sentences of the signature Σ. Let $\mathfrak{M} = \langle M, \Omega_P \rangle$. Let $P_M \in \Omega_P$. Then V is a *first-order* valuation of $\mathbf{S}_\Sigma$ in the model $\mathfrak{M}$ if the following holds:

1. $V(Pa_1 \cdots a_n) = \begin{cases} \mathbf{1} & \text{if } \langle a_1, \ldots, a_n \rangle \in \overline{P} \\ \mathbf{0} & \text{otherwise.} \end{cases}$
2. (a) $V(\neg A) = \begin{cases} \mathbf{1} & \text{if } V(A) = \mathbf{0} \\ \mathbf{0} & \text{if } V(A) = \mathbf{1} \end{cases}$

$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathbf{T}\forall xA$	$\mathbf{F}\exists xA$	$\mathbf{F}\forall xA$	$\mathbf{T}\exists xA$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathbf{T}A^{x/a}$	$\mathbf{F}A^{x/a}$	$\mathbf{F}A^{x/a}$	$\mathbf{T}A^{x/a}$
(a is an arbitrary parameter)		(a is a 'new' parameter)	

Table 6.1: Signed tableau processing rules for quantifiers

(b) $V(A \supset B) = \begin{cases} \mathbf{1} & \text{if } V(A) = \mathbf{0} \text{ or } V(B) = \mathbf{1} \\ \mathbf{0} & \text{if } V(A) = \mathbf{1} \text{ and } V(B) = \mathbf{0}. \end{cases}$

3. $V(\forall xA) = \begin{cases} \mathbf{1} & \text{if } V(A^{x/a}) = \mathbf{1} \text{ for all } a \in M \\ \mathbf{0} & \text{if } V(A^{x/a}) = \mathbf{0} \text{ for at least one } a \in M. \end{cases}$

4. $V(\exists xA) = \begin{cases} \mathbf{1} & \text{if } V(A^{x/a}) = \mathbf{1} \text{ for at least one } a \in M \\ \mathbf{0} & \text{if } V(A^{x/a}) = \mathbf{0} \text{ for all } a \in M. \end{cases}$

We can now define satisfiability and validity.

Definition 6.6. Let $A \in \mathbf{F}_\Sigma$. Then A is *satisfied*, or *true*, in the model $\mathfrak{M}$ if there is a valuation V of $\mathfrak{M}$ such that $V(A) = \mathbf{1}$. (Sometimes it is convenient to indicate this fact as $\mathfrak{M} \models A$.)

Definition 6.7. Let $A \in \mathbf{F}_\Sigma$. Then A is *valid* in the model $\mathfrak{M}$ if for all valuations V of $\mathfrak{M}$, $V(A) = \mathbf{1}$.

Definition 6.8. Let $A \in \mathbf{F}_\Sigma$. Then A is *satisfiable* if there is a model $\mathfrak{M}$ and a valuation V of $\mathfrak{M}$ such that $V(A) = \mathbf{1}$.

6.2 Rules for semantic tableaux

Tableaux have the same application in predicate calculus as they did in sentence calculus. To the earlier rules we add the rules for developing the quantifiers.

Example 6.9. Let us find a proof for $\exists x(Px \vee Qx) \vdash \exists xPx \vee \exists xQx$:

$$\mathbf{T}\exists x(Px \vee Qx)$$
$$\mathbf{F}\exists xPx \vee \exists xQx$$

$$\mathbf{F}\exists xPx$$
$$\mathbf{F}\exists xQx$$

$$\mathbf{T}Pa \vee Qa$$

$$\mathbf{F}Pa$$
$$\mathbf{F}Qa$$

$$\begin{array}{cc} \mathbf{T}Pa & \mathbf{T}Qa \\ \times & \times \end{array}$$

To justify the tableau method, we shall first establish the soundness of tableau proofs.

Definition 6.10. Let $A \in \mathbf{F}_\Sigma$ and $\mathfrak{M} = \langle M, I\rangle$. Then the C-sentence $A^c = A^{a_1/c_1 \cdots a_n/c_n}$ is a formula obtained from A by replacing each occurrence of individual parameters with individual constants $c_1, \ldots, c_n$ such that $\overline{c_1}, \ldots, \overline{c_n} \in M$.

The notions of satisfiability and simultaneous satisfiability naturally carry over to the case of A^c. Let us say further that a tableau $\mathscr{T}$ is simultaneously satisfiable if at least one path of it is simultaneously satisfiable.

Proposition 6.11. *Let $\mathscr{T}$ and $\mathscr{T}'$ be tableaux such that $\mathscr{T}'$ is an immediate extension of $\mathscr{T}$. Then, if $\mathscr{T}$ is simultaneously satisfiable, $\mathscr{T}'$ is also simultaneously satisfiable.*

Proof. We should consider cases for each of the tableau rules. Let us consider only selected rules.

1. Suppose $\mathscr{T}'$ is obtained from $\mathscr{T}$ by applying the rule

$$\begin{array}{c} \vdots \\ \mathbf{T}A \supset B \\ \vdots \end{array} \qquad\qquad \mathbf{F}A \qquad\qquad \mathbf{T}B$$

to the path θ of $\mathscr{T}$. Since $\mathscr{T}$ is simultaneously satisfiable, it contains a simultaneously satisfiable branch τ. If $\tau \neq \theta$, then τ is in $\mathscr{T}'$, and so $\mathscr{T}'$ is simultaneously satisfiable. Suppose $\tau = \theta$. Then θ is satisfiable. Let the model $\mathfrak{M} = \langle M, I\rangle$ simultaneously satisfy θ. Then, for a valuation V of $\mathfrak{M}$, we have that $V((A \supset B)^c) = \mathbf{1}$, where every $c_i \in M$. Thus $V((\neg A)^c) = \mathbf{1}$ or $V(B^c) = \mathbf{1}$. Clearly, then, the valuation V simultaneously satisfies either the pair $\langle \theta, \neg A\rangle$, or the pair $\langle \theta, B\rangle$. It follows that $\mathscr{T}'$ is simultaneously satisfiable, as at least one of its paths is simultaneously satisfiable.

2. Suppose $\mathscr{T}'$ is obtained from $\mathscr{T}$ by applying the rule

$$\begin{array}{c} \vdots \\ \mathbf{T}\forall x A \\ \vdots \end{array} \qquad\qquad \mathbf{T}A^{x/a}$$

to the path θ of $\mathscr{T}$. Since $\mathscr{T}$ is simultaneously satisfiable, it contains a simultaneously satisfiable branch τ. If $\tau \neq \theta$, then τ is in $\mathscr{T}'$, and so $\mathscr{T}'$ is simultaneously satisfiable. Suppose $\tau = \theta$. Then θ is satisfiable. Let the model $\mathfrak{M} = \langle M, I\rangle$ simultaneously satisfy θ. Then, for a valuation V of $\mathfrak{M}$, we have that $V(\forall x(A)^c) = V((\forall x A)^c) = \mathbf{1}$, where every $c_i \in M$. Hence, in particular, $\mathfrak{M}$ satisfies $A^{x/a}$. Thus $\mathscr{T}'$ is simultaneously satisfiable. □

Proposition 6.12 (Soundness for tableaux)**.** *Let $\Gamma = \{A_1, \ldots, A_n\}$ be a set of sentences with parameters. If there is a closed finite tableau starting with $A_1, \ldots, A_n$, then Γ is not simultaneously satisfiable.*

Proof. Let $\mathscr{T}$ be a tableau starting with $A_1, \ldots, A_n$. Then there is a sequence of tableaux $\mathscr{T}_1, \mathscr{T}_2, \ldots, \mathscr{T}_n$ such that $\mathscr{T}_1$ has the sole branch $\langle A_1, \ldots, A_n\rangle$, $\mathscr{T}_n = \mathscr{T}$, and for each $0 < i < n$, $\mathscr{T}_{i+1}$ is an immediate extension of $\mathscr{T}_i$. Suppose, for *reductio*, that $\{A_1, \ldots, A_n\}$ is simultaneously satisfiable. Then $\mathscr{T}_1$ is also simultaneously satisfiable. Using induction on i and Proposition 6.11 we derive that, for each $0 < i < n + 1$, $\mathscr{T}_i$ is satisfiable. But `this is impossible, as` $\mathscr{T}_n$ `is closed by assumption`. Hence $\{A_1, \ldots, A_n\}$ is not simultaneously satisfiable. why? … □

6.3 Logical equivalences

Before we show completeness of our method, let us record some useful equivalences of first-order calculus.

Definition 6.13. Let $A \in \mathbf{F}_\Sigma$ and let $x_1, \ldots, x_n$ be the variables occurring freely in A and $a_1, \ldots, a_n$ be the parameters not occurring in A. Let $A^a = A^{x_1/a_1 \cdots x_n/a_n}$. Then A is *satisfiable* if and only if A^a is satisfiable. And A is *logically valid* if and only if A^a is logically valid.

We can now register some properties of satisfiability and validity.

Proposition 6.14. *Let $A \in \mathbf{F}_\Sigma$. Then A is logically valid if and only if $\neg A$ is not satisfiable. And A is logically satisfiable if and only if $\neg A$ is not not logically valid.*

Proof. Straightforward from definitions. □

Proposition 6.15. *Let $A \in \mathbf{F}_\Sigma$. Then A is logically valid if and only if $\forall A$ is logically valid. A is satisfiable if and only if $\exists x A$ is satisfiable.*

Proof. Exercise. □

Definition 6.16. Let $A, B \in \mathbf{F}_\Sigma$. Then A is *logically equivalent* to B ($A \simeq B$) if and only if $A \leftrightarrow B$ is logically valid.

Proposition 6.17. *Let $A, B \in \mathbf{F}_\Sigma$ such that A and B contain a free occurrence of x. If $A \simeq B$, then $\forall x A \simeq \forall x B$ and $\exists x A \simeq \exists x B$.*

Proof. Let the formula A contain the free occurrences of the variables $x, x_1, \ldots, x_n, y_1, \ldots, y_k$, and let B contain the free occurrences of the variables $x, x_1, \ldots, x_n, z_1, \ldots, z_l$, so that the common free variables are $x, x_1, \ldots, x_n$. Since $A \simeq B$, on any arbitrary model $\mathfrak{M} = \langle M; \cdots \rangle$ the formula A will be logically valid only for those arrays $\langle \overline{x_1}, \ldots, \overline{x_n}, \overline{y_1}, \ldots, \overline{y_k}, \overline{z_1}, \ldots, \overline{z_l} \rangle$, for which B is also logically valid. But this means precisely that $\forall x A \simeq \forall x B$.

The case for the existential quantifier is proven analogously. □

Let us now formulate important logical equivalences for $\mathsf{T_p}$. We shall split them into four groups.

Proposition 6.18. *Swapping quantifiers:*

1. $\forall x \forall y P x y \simeq \forall y \forall x P x y$.
2. $\exists x \exists y P x y \simeq \exists y \exists x P x y$.

Proof. To show the first equivalence, we notice that $\forall x \forall y P x y$ is true just in case $P x y$ is logically valid. The second equivalence is left as an exercise. □

Remark. Another way of putting this claim is as follows:

1. $\forall x A \simeq A$;
2. $\exists x A \simeq A$,

assuming in both cases that x does not occur freely in A.

Proposition 6.19. *Linking existential and universal quantifiers:*

1. $\neg \forall x A \simeq \exists x \neg A$;
2. $\neg \exists x A \simeq \forall x \neg A$.

Proof. To prove 1, we notice that if the formula $\neg \forall x A$ is true, then $\forall x A$ is false. Thus the formula A is not logically valid. But then $\neg A$ is satisfiable. That is, for some $\overline{x}$ it is true. Hence, $\exists x \neg A$ is true. The other direction is proved similarly.

To prove 2, we notice that $\neg \exists x A$ is true just in case $\exists x A$ is false. This means that A is not satisfiable. And this is so just in case $\neg A$ is logically valid. And $\neg A$ is logically valid just in case $\forall x \neg A$ is logically valid. □

Remark. In the finite case, there is an analogy here with de Morgan's laws. The proved proposition may be regarded as a generalisation into the infinite case.

Proposition 6.20. *Let $A \in \mathbf{F}_\Sigma$ such that x does not occur freely in it. The distribution rules for quantifiers are as follows:*

1. $\forall x (A \wedge P x) \simeq A \wedge \forall x P x$;

2. $\forall x(A \vee Px) \simeq A \vee \forall xPx$;

3. $\exists x(A \wedge Px) \simeq A \wedge \exists xPx$;

4. $\exists x(A \vee Px) \simeq A \vee \exists xPx$;

5. $\exists x(A \supset Px) \simeq A \supset \exists xPx$;

6. $\exists x(Px \supset A) \simeq \forall xPx \supset A$;

7. $\forall x(A \supset Px) \simeq A \supset \forall xPx$;

8. $\forall x(Px \supset A) \simeq \exists xPx \supset A$.

The same claims hold if we replace the formula Px by an arbitrary formula B.

Proof. All these claims may be verified by examining the relevant tableaux. □

Proposition 6.21. *Let $A \in \mathbf{F}_\Sigma$. The rules for renaming bound variables (or rules of relettering) are as follows:*

1. $\forall xA \simeq \forall yA^{y/a}$;

2. $\exists xA \simeq \exists yA^{y/a}$.

Proof. By examining the relevant tableaux. □

6.4 Prenex normal forms

We shall now prove that any formula can be transformed into *prenex* normal form. We shall need several auxiliary notions and a lemma.

Definition 6.22. The notion of *immediate subformulae* is determined according to the following rules:

1. A and B are immediate subformulae of $A \wedge B$, $A \vee B$, $A \supset B$, whilst A is an immediate subformula of $\neg A$.

2. For any parameter a, variable x, and formula A, $A^{x/a}$ is an immediate subformula of $\forall xA$ and of $\exists xA$.

Definition 6.23. *Subformulae* are determined by the following rules:

1. If A is an immediate subformula of B, or $A = B$, then A is a subformula of B;

2. If A is a subformula of B, and B is a subformula of C, then A is a subformula of C.

Proposition 6.24. *Let $A \in \mathbf{F}_\Sigma$. If we replace its subformula B by a subformula C, such that $B \simeq C$, then the resulting formula A' will be such that $A \simeq A'$.*

Proof. Omitted. Hint: we should induce on the complexity of A. □

Definition 6.25. Let $A \in \mathbf{F}_\Sigma$. Then A is said to be in *prenex normal form* if A has the form $Q_1x_1 \cdots Q_nx_nB$, where each Q_i is a quantifier $\forall$ or $\exists$, $x_i \neq x_j$ if $i \neq j$, and B contains no quantifiers.

Proposition 6.26. *For each $A \in \mathbf{F}_\Sigma$ there is $B \in \mathbf{F}_\Sigma$ such that $A \simeq B$ and B is in prenex normal form.*

Proof. To facilitate understanding, let us first state the proof informally. We shall use the logical equivalences for $\mathsf{T_p}$ established earlier. To begin with, relying on de Morgan laws, we eliminate all symbols for material conditionals and bi-conditionals. To the resulting formula we shall apply two types of transformation. In the first transformation, we find a subformula A' of A having the form of either $C \wedge \forall xB$, or $C \vee \forall xB$, or $C \wedge \exists xB$, or $C \vee \exists xB$. For instance, let $A' = C \wedge \forall xB$ (other cases are done analogously). If C has a free occurrence of x, then we replace x by some z not occurring in A. If not, then replace A' by $\forall x(C \wedge B)$. We repeat this procedure the required number of times. In the second type of transformation, we replace the subformulae having the form $\neg\forall xB$ or $\neg\exists xB$ by $\exists x\neg B$ or $\forall x\neg B$ respectively. In this way we are able to transform A into $Q_1x_1 \cdots Q_nx_nB$ with B containing no quantifiers.

The same procedure allows a formal presentation along the following lines. Let $A \in \mathbf{F}_\Sigma$. Let $\lambda(A) \in \{0, 1, 2, \ldots\}$ be the number of occurrences of the quantifiers in A. We shall use induction on n and prove

that for a formula A with $\lambda(A) \leq n$ there is $B \in \mathbf{F}_\Sigma$ such that B is in prenex normal form, and $A \simeq B$, $\lambda(A) = \lambda(B)$, and the number of the free occurrences of variables in B is equal to the number of the free occurrences of variables in A.

Thus let $n = 0$. In this case, we may let B to be just the formula A. Now let $n > 0$. Suppose $\lambda(A) \leq n$. The quantifier-free case is trivial. Then let A contain quantifiers. If A has the form $\neg C$ and $\lambda(A) > 0$, then $\lambda(C) = \lambda(A) > 0$. By induction hypothesis there is a formula of the form QxD which is a prenex normal form for C and where $\lambda(D) = \lambda(A)$ and where D may contain quantifiers. Introduce the notation $\forall^{-1} = \exists$ and $\exists^{-1} = \forall$. We have that $A \simeq Q^{-1}x\neg D$. Now, since $\lambda(\neg D) = \lambda(QxD) = \lambda(QxA) - 1 \leq n - 1$, there is a formula $B \simeq \neg D$ which is in prenex normal form and is such that $\lambda(B) = \lambda(\neg D)$. By our equivalences above, $Q^{-1}xB \simeq A$ and $Q^{-1}xB$ satisfies the properties of the prenex normal form for A.

The proof w.r.t. the number of free variables is left as an exercise. □

Example 6.27. Let us bring the formula $\neg\exists y\neg\exists u((\exists xPxyz \supset \forall xRxy) \wedge \neg\forall zPzuz)$ to a prenex normal form:

$$\begin{aligned}
&\neg\exists y\neg\exists u((\exists xPxyz \supset \forall xRxy) \wedge \neg\forall zPzuz) \simeq\\
&\forall y\neg\neg\exists u((\neg\exists xPxyz \vee \forall xRxy) \wedge \exists z\neg Pzuz) \simeq\\
&\forall y\exists u((\forall x\neg Pxyz \vee \forall tRty) \wedge \exists v\neg Pvuv) \simeq\\
&\forall y\exists u(\forall t(\forall x\neg Pxyz \vee Rty) \wedge \exists v\neg Pvuv) \simeq\\
&\forall y\exists u(\forall t\forall x(\neg Pxyz \vee Rty) \wedge \exists v\neg Pvuv) \simeq\\
&\forall y\exists u\forall t\forall x((\neg Pxyz \vee Rty) \wedge \exists v\neg Pvuv) \simeq\\
&\forall y\exists u\forall t\forall x\exists v((\neg Pxyz \vee Rty) \wedge \neg Pvuv).
\end{aligned}$$

6.5 Skolem forms

There is a common mathematical practice of picking elements depending on the prior choice of some other elements. For instance, if we have shown that for each x there is y such that $\phi(x, y)$, then it is natural to introduce a function f^1 picking y for each x. We will then replace $\phi(x, y)$ with $\phi(x, f(x))$. Such a technique calls for the employment of a special device.

Discuss tableau rules

Consider a prenex formula $A \in \mathbf{F}_\Sigma$. It contains pairwise distinct variables $x_1, \ldots, x_n$, quantifiers $Q_1, \ldots, Q_n$, and a quantifier-free formula $B \in \mathbf{F}_\Sigma$. Then we can first identify the indices of the existential quantifiers: $\{j_1, \ldots, j_i, \ldots, j_p \mid Q_{j_i} = \exists, 1 \leq i \leq n\}$. Given such a set, we can now expand the signature Σ into Σ^A_{Sk} by adding p new symbols for parameters or functions. Those will be symbols for *Skolem functions* (sometimes also called 'Herbrand functions') associated with A. We may also compute the arity of the particular symbol f_h. For $1 \leq h \leq p$, its arity will equal the number of times the universal quantifier occurs to the left of Q_{j_h} in the prefix of A. That would be exactly the number $j_h - h$. We also note that constants may be regarded as 0-ary function symbols. Therefore, for f_h to be a constant, we require that $j_h = h$, or equivalently, that the first h quantifications be existential.

Example 6.28. Let the prefix of A be:

$$\forall x_0\forall x_1\forall x_2\exists x_3\exists x_4\forall x_5\exists x_6\forall x_7\forall x_8\exists x_9\forall x_{10}.$$

Then we expand Σ into Σ^A_{Sk} by adding four new function symbols f_1, f_2, f_3, f_4, whose respective arities will be $3, 3, 4, 6$.

We can now build the *Skolem form* A_{Sk} of the formula A. Obviously it will be a prenex formula containing only universal quantifiers. Let u_h be a term of Σ^A_{Sk} consisting of the function symbol f_h followed by $j_h - h$ universally quantified variables such that they occur to the left of the variable x_{j_h} in the prefix of the formula A. In general, u_h will take the form:

$$f_h x_1 x_2 \cdots x_{j_1-1} x_{j_1+1} \cdots x_{j_2-1} x_{j_2+1} \cdots x_{j_{h-1}-1} x_{j_{h-1}+1} \cdots x_{j_h-1}.$$

Then, for each $1 \leq h \leq n$ we replace each occurrence of x_{j_h} in B by the term u_h. And in front of this formula we put the prefix of A from which each occurrence of the existential quantifier has been deleted.

Example 6.29. Suppose the signature Σ contains predicate parameters P^1 and R^2. Consider $A \in \mathbf{F}_\Sigma$:

$$\exists x_0\exists x_1\forall x_2\exists x_3\forall x_4\forall x_5\exists x_6((Rx_0x_2 \wedge Px_5) \supset (Rx_6x_2 \vee (Rx_1x_5 \wedge Rx_4x_3))).$$

The signature Σ^A_{Sk} will then contain four new symbols: two constants f_1 and f_2, a unary function parameter f_3 and a ternary function parameter f_4. The formula A_{Sk} will take the form:

$$\forall x_2 \forall x_4 \forall x_5((Rf_1x_2 \wedge Px_5) \supset (Rf_4(x_2, x_4, x_5)x_2 \vee (Rf_2x_5 \wedge Rx_4f_3(x_2)))).$$

We must keep in mind that A_{Sk} belongs to a richer signature than A. Hence it is wrong to say that A is equivalent to its Skolem form. What is true, however, is that when A is considered as a formula belonging to Σ^A_{Sk}, then it will be semantically entailed by its Skolem form.

Example 6.30. Let us illustrate the claim of entailment. Let A be the formula $\forall x_0 \exists x_1 Rx_0x_1$. We obtain its Skolem form by adding the function parameter g to its signature Σ. The formula A_{Sk} will be $\forall x_0 Rx_0g(x_o)$. Let $\mathfrak{M} = \langle M; R, g\rangle$ be the algebraic system such that $\mathfrak{M} \models A_{\mathrm{Sk}}$. Thus we have that for every $a \in M$ we have that $\langle a, g(a)\rangle \in R$. But this means that $\mathfrak{M} \models A$. Therefore, $A_{\mathrm{Sk}} \vDash A$.

Skolem functions and Skolem forms have an interesting application in proof theory, as we shall see later in our discussion of Gentzen systems. A special case of Skolem functions is also used in one of the proofs of first-order completeness. But their most vital role belongs in model theory. One important fact there is that for a closed formula to be satisfiable it is necessary and sufficient that its Skolem form be satisfiable. Most of these applications will fall outside the scope of our concerns. Here we shall only formulate a basic property of Skolem forms:

Proposition 6.31. *Let $y_1, \ldots, y_n$ be pairwise distinct variables and let $A \in \mathbf{F}_\Sigma$ be a prenex formula with free occurrences of $y_1, \ldots, y_n$. Then the formula $A_{\mathrm{Sk}} \supset A$ of the signature Σ^A_{Sk} is valid.*

Proof. To be supplied. □

6.6 Completeness for tableaux

We now resume our enquiry into the completeness of the tableau method. Let M be a domain of individuals and let Γ be a set of signed C-sentences (which, we recall, should be regarded as closed C-formulae) associated with it. We say that Γ is *closed* if it contains a conjugate pair of C-sentences. Of particular interest to us will be those sets of C-sentences where, so to speak, all the sentences have been processed according to the tableau rules. Thus:

Definition 6.32. Let Γ be a set of signed C-sentences. Then Γ is *well-developed on M* if Γ obeys the rules for semantic tableaux, so that the following conditions hold:

1. If $\mathbf{T}\neg A \in \Gamma$, then $\mathbf{F}A \in \Gamma$, and if $\mathbf{F}\neg A \in \Gamma$, then $\mathbf{T}A \in \Gamma$;
2. If $\mathbf{T}A \wedge B \in \Gamma$, then $\mathbf{T}A \in \Gamma$ and $\mathbf{T}B \in \Gamma$, and if $\mathbf{F}A \wedge B \in \Gamma$, then either $\mathbf{F}A \in \Gamma$ or $\mathbf{F}B \in \Gamma$;
3. If $\mathbf{T}A \vee B \in \Gamma$, then either $\mathbf{T}A \in \Gamma$ or $\mathbf{T}B \in \Gamma$, and if $\mathbf{F}A \wedge B \in \Gamma$, then $\mathbf{F}A \in \Gamma$ and $\mathbf{F}B \in \Gamma$;
4. If $\mathbf{T}A \supset B \in \Gamma$, then either $\mathbf{F}A \in \Gamma$ or $\mathbf{T}B \in \Gamma$, and if $\mathbf{F}A \wedge B \in \Gamma$, then $\mathbf{T}A \in \Gamma$ and $\mathbf{F}B \in \Gamma$;
5. (The condition for the bi-conditional is left as an exercise.)
6. If $\mathbf{T}\exists xA \in \Gamma$, then $\mathbf{T}A^{x/a} \in \Gamma$ for at least one $a \in M$, and if $\mathbf{F}\exists xA \in \Gamma$, then $\mathbf{F}A^{x/a} \in \Gamma$ for all $a \in M$;
7. If $\mathbf{T}\forall xA \in \Gamma$, then $\mathbf{T}A^{x/a} \in \Gamma$ for all $a \in M$, and if $\mathbf{F}\forall xA \in \Gamma$, then $\mathbf{F}A^{x/a} \in \Gamma$ for at least one $a \in M$.

(Well-developed sets are called 'Hintikka sets' in [Smu68].)

To prove completeness, we shall first establish several lemmas.

Proposition 6.33 (Hintikka's Lemma)**.** *Let Γ be a set of signed C-sentences. If Γ is well-developed on M and open, then Γ is simultaneously satisfiable.*

Proof. Let us assume that Γ is well-developed on M and open. Consider a model $\mathfrak{M} = \langle M, I\rangle$ of the signature Σ, such that for any predicate parameter $P^n \in \Sigma_p$, $\overline{P^n} = \{\langle a_1, \ldots, a_n\rangle \in M^n \mid \mathbf{T}Pa_1 \cdots a_n \in \Gamma\}$. We now claim that for any C-sentence A the following holds:

i. If $\mathbf{T}A \in \Gamma$, then $V(A) = \mathbf{1}$;

ii. If $\mathbf{F}A \in \Gamma$, then $V(A) = \mathbf{0}$.

The proof should be by induction on the complexity of A. As the basic step, we consider the formula A such that $d(A) = 0$. The inductive hypothesis will be that if $d(A) > 0$, then for any formula B such that $d(B) < d(A)$ our claims hold. We shall have to consider all the cases corresponding to each of the tableau rules. Let us consider here some of them, and the rest will be left as an exercise.

1. Let $d(A) = 0$. Then A will have the form $Pa_1 \cdots a_n$.
   i. If $\mathbf{T}Pa_1 \cdots a_n \in \Gamma$, then we already we have that $\langle a_1, \ldots, a_n \rangle \in \overline{P^n}$, and so it follows that $V(Pa_1 \cdots a_n) = \mathbf{1}$.
   ii. If $\mathbf{F}Pa_1 \cdots a_n \in \Gamma$, then we already we have that $\langle a_1, \ldots, a_n \rangle \notin \overline{P^n}$, and so it follows that $V(Pa_1 \cdots a_n) = \mathbf{0}$.
2. Suppose $d(A) > 0$ and A has the form $\neg B$. Then $d(A) > d(B)$, and so the inductive hypothesis holds for B.
3. Suppose $d(A) > 0$ and A has the form $B \wedge C$. Then $d(A) > d(B)$ and $d(A) > d(C)$, and so the inductive hypothesis holds for B and C.
   i. If $\mathbf{T}B \wedge C \in \Gamma$, then, since Γ is well-developed, we already we have that $\mathbf{T}B, \mathbf{T}C \in \Gamma$, and so it follows that $V(B) = V(C) = \mathbf{1}$. Hence $V(B \wedge C) = \mathbf{1}$.
   ii. If $\mathbf{F}B \wedge C \in \Gamma$, then we already we have that $\mathbf{F}B \in \Gamma$ or $\mathbf{F}C \in \Gamma$, and so it follows that either $V(B) = \mathbf{0}$ or $V(C) = \mathbf{0}$. Hence $V(B \wedge C) = \mathbf{0}$.
4. Suppose $d(A) > 0$ and A has the form $\exists x B$. Then for all $a \in M$ we have $d(B^{x/a}) < d(A)$, and so the inductive hypothesis holds for $B^{x/a}$.
5. Suppose $d(A) > 0$ and A has the form $\forall x B$. Then for all $a \in M$ we have $d(B^{x/a}) < d(A)$, and so the inductive hypothesis holds for $B^{x/a}$. □

For the following lemma we let $N = \{a_1, \ldots, a_n, \ldots\}$ be a set of parameters.

Proposition 6.34. *Let $\mathscr{T}_0$ be a finite tableau. By applying tableau rules it is possible to extend $\mathscr{T}_0$ to a possibly infinite tableau $\mathscr{T}$ such that every closed path of $\mathscr{T}$ is finite and every open path of it is well-developed on N.*

Proof. We construct an array of finite extensions of $\mathscr{T}_0$: $\mathscr{T}_1, \mathscr{T}_2, \ldots, \mathscr{T}_i, \ldots,$. If for some i the tableau $\mathscr{T}_i$ is closed, then the procedure halts and we let $\mathscr{T} = \mathscr{T}_i$. But in any case we may set $\mathscr{T} = \mathscr{T}_\infty = \bigcup_{i=0}^{\infty} \mathscr{T}_i$.

We proceed as follows. Let us call the node $X \in \mathscr{T}_0$ 'semi-universal', if it has the form $\mathbf{T}\forall x A$ of $\mathbf{F}\exists x A$. Suppose we have constructed $\mathscr{T}_i$. Then for each semi-universal node $X \in \mathscr{T}_i$ and each $n \leq i$, we apply the relevant tableau rule to extend each open path of $\mathscr{T}_i$ containing X by $\mathbf{T}A^{x/a_n}$ or $\mathbf{F}A^{x/a_n}$. Let $\mathscr{T}_{i+1}$ be the tableau obtained as a result. Then for each node X of $\mathscr{T}_{i+1}$ which is not semi-universal we extend each open path containing X by applying the relevant tableau rule. And then again repeat the procedure for the finite tableau $\mathscr{T}_{i+2}$. Therefore, a closed path is never extended, so that all closed paths of $\mathscr{T}_\infty$ are finite. It is also clear from our construction that each open path of $\mathscr{T}_\infty$ is well-developed on N. Therefore, $\mathscr{T}_\infty$ has the desired properties. □

Finally, let us prove another lemma related to the properties of trees generally. Recall that in a finitely generated tree each node has finitely many successors (*i.e.* 'immediate' successors).

Proposition 6.35 (König's Lemma)**.** *Let $\mathscr{T}$ be a finitely generated tree with infinitely many nodes. Then $\mathscr{T}$ contains at least one infinite path.*

Proof. Let $\mathscr{T}_x$ be the set of all nodes $x \in \mathscr{T}$ such that x has infinitely many descendants (*i.e.* both 'immediate' and non-'immediate' successors). Then $\mathscr{T}_x$ is a sub-tree of $\mathscr{T}$. Each $x \in \mathscr{T}_x$ will have at least one successor. Let x_1 be the root of $\mathscr{T}$. By assumption, since $\mathscr{T}$ is infinite, the root will have infinitely many descendants. If all of the successors of x_1 have finitely many descendants, then x_1 would have had finitely many descendants—which is false. Hence there is at least one successor of x_1 which has infinitely many descendants. Let it be x_2. We thus construct inductively a sequence $x_1, x_2, \ldots,$. Clearly such a sequence will constitute an infinite path of $\mathscr{T}$. □

No need for the axiom of choice

We are now in a position to prove completeness. Similarly to the propositional case (Proposition 4.47), we are going to connect the semantic notion of validity with the syntactic notion of provability. Recall that the provability of A by the tableau method would mean that there is a closed tableau with $\mathbf{F}A$ at the root.

Proposition 6.36 (Completeness)**.** *Let $\Gamma = \{A_1, \ldots, A_n\}$ a set of sentences with parameters. If Γ is not simultaneously satisfiable, then there is a finite closed tableau with Γ at the root.*

Proof. By Proposition 6.34 there is a tableau $\mathscr{T}$ such that every closed path of it is finite and every open path of it is well-developed on M. If $\mathscr{T}$ is closed, then by König's Lemma $\mathscr{T}$ is finite. Suppose $\mathscr{T}$ is open. Let Δ be its open path. Then Δ is well-developed. But by Hintikka's Lemma Δ is simultaneously satisfiable in M. Since $\Gamma \subseteq \Delta$, Γ is simultaneously satisfiable in M. □

The following claims are obtained as simple corollaries of the completeness result as stated above.

Proposition 6.37 (Löwenheim)**.** *Let $\Gamma = \{A_1, \ldots, A_n\}$ a set of sentences with parameters. If Γ is simultaneously satisfiable, then it is simultaneously satisfiable in a countable domain.*

Proof. Follows from the proof of Proposition 6.36. □

Proposition 6.38. *Let $\Gamma = \{A_1, \ldots, A_n\}$ a set of sentences with parameters. Γ is not simultaneously satisfiable if and only if there exists a finite closed signed tableau starting with $\mathbf{T}A_1, \ldots, \mathbf{T}A_k$.*

Proposition 6.39. *The sentence A is logically valid if and only if there exists a finite closed signed tableau starting with $\mathbf{F}A$.*

Proposition 6.40. *The sentence B is a logical consequence of $A_1, \ldots, A_k$ if and only if there exists a finite closed signed tableau starting with $\mathbf{T}A_1, \ldots, \mathbf{T}A_k, \mathbf{F}B$.*

We shall now prove the compactness theorem for predicate calculus, a much deeper result than compactness for propositional calculus. It was originally proven by Mal'cev, of course using different techniques. We again prove it for the countable case, since the uncountable case invokes the axiom of choice.

Proposition 6.41 (Compactness)**.** *Let Γ be a countably infinite set of first-order sentences. Then Γ is simultaneously satisfiable if and only if each finite subset of Γ is simultaneously satisfiable.*

Proof. Let $\Gamma = \{A_0, A_1, \ldots, A_i, \ldots\}$. We start by letting $\mathscr{T}_0$ be the empty tableau. Suppose we have constructed $\mathscr{T}_i$. Extend $\mathscr{T}_i$ to $\mathscr{T}_i'$ by appending A_i to each open path of $\mathscr{T}_i$. Since $\{A_0, A_1, \ldots, A_i\}$ is simultaneously satisfiable, $\mathscr{T}_i'$ has at least one open path. Now extend $\mathscr{T}_i'$ to $\mathscr{T}_{i+1}$ and then to $\mathscr{T}_{i+2}$ as in the proof of Proposition 6.34. Finally we let $\mathscr{T} = \mathscr{T}_\infty = \bigcup_{i=0}^{\infty} \mathscr{T}_i$. We have that every closed path of $\mathscr{T}$ is finite, and every open path of $\mathscr{T}$ is well-developed on M. Note also that $\mathscr{T}$ is an infinite, finitely branching tree. By König's Lemma we let Γ' be an infinite path in $\mathscr{T}$. Then Γ' is well-developed on M and $\Gamma \subseteq \Gamma'$. By Hintikka's Lemma, Γ' is simultaneously satisfiable. Hence Γ is simultaneously satisfiable. □

We can now prove another well-known theorem which strengthens the result of Proposition 6.37.

Proposition 6.42 (Skolem-Löwenheim)**.** *Let Γ be a countably infinite set of first-order sentences. If Γ is simultaneously satisfiable, then Γ is simultaneously satisfiable in a countable domain.*

Proof. Follows from the proof of the compactness theorem. Notice that Γ is well-developed in the countably infinite domain M, and therefore, simultaneously satisfiable on it. □

Let us now go back to the completeness result we have obtained. It appears as though the tableau method provides a test for logical validity of sentences of the predicate calculus. Unfortunately, the test would only be partially effective. If a sentence A is logically valid, we will certainly find a finite closed tableau starting with $\mathbf{F}A$. But if A is not logically valid, we will not necessarily find a finite tableau to show this.

Example 6.43. Consider the formula:

$$A : \forall x \exists y Pxy \wedge \forall x \forall y (Pxy \supset \neg Pyx) \wedge \forall x \forall y \forall z (Pxy \supset (Pyz \supset Pxz)).$$

This formula is satisfiable on an infinite model, but is not satisfiable on any finite one. Consider $\mathfrak{M} = \langle \{0, 1, 2, \ldots\}; \overline{<} \rangle$. Clearly $\mathfrak{M} \models A$. To see that A is not satisfiable on a finite model, let $\mathfrak{M} = \langle M; \overline{P} \rangle$ such that $\mathfrak{M} \models A$. Note that $V(Paa) = \mathbf{0}$ for any $\overline{a} \in M$. Let $\overline{a_0} \in M$. Now select $\overline{a_1} \in M$ such that $V(Pa_0a_1) = \mathbf{1}$. Then select $\overline{a_2} \in M$ such that $V(Pa_1a_2) = \mathbf{1}$. By repeating the process we get a sequence $\langle a_0, a_1, a_2, \ldots \rangle$. All the members of this sequence are different: if $i < j$, then $V(P(a_ia_{i+1})) = V(P(a_{i+1}a_{i+2})) = \cdots = V(P(a_{j-1}a_j)) = \mathbf{1}$. Since $\mathfrak{M} \models \forall x \forall y \forall z (Pxy \supset (Pyz \supset Pxz))$, we have that $V(P(a_ia_j)) = \mathbf{1}$. So $a_i \neq a_j$. Therefore, M is (countably) infinite.

The same conclusion may be reached by examining the tableau for A in Figure 6.1. Its infinite open path gives rise (see the proof of Proposition 6.33) to an infinite model, such as $\mathfrak{M}$ above.

$$\mathbf{T}\forall x\exists y Pxy$$
$$\mathbf{T}\forall x\forall y(Pxy \supset \neg Pyx)$$
$$\mathbf{T}\forall x\forall y\forall z(Pxy \supset (Pyz \supset Pxz))$$
$$\mathbf{T}\exists y Pa_1 y$$
$$\mathbf{T}Pa_1a_2$$

$$\mathbf{T}\forall y(Pa_1y \supset \neg Pya_1)$$
$$\mathbf{T}Pa_1a_2 \supset \neg Pa_2a_1$$

$$\mathbf{F}Pa_1a_2 \qquad \mathbf{T}\neg Pa_2a_1$$

$$\mathbf{T}\exists y Pa_2 y$$
$$\mathbf{T}Pa_2a_3$$
$$\vdots$$
$$\mathbf{T}\neg Pa_3a_2$$

$$\mathbf{T}\forall y\forall z(Pa_1y \supset (Pyz \supset Pa_1z))$$
$$\mathbf{T}\forall z(Pa_1a_2 \supset (Pa_2z \supset Pa_1z))$$

$$\mathbf{T}(Pa_1a_2 \supset (Pa_2a_3 \supset Pa_1a_3))$$

$$\mathbf{F}Pa_1a_2 \;\; \mathbf{T}Pa_2a_3 \supset Pa_1a_3$$

$$\mathbf{F}Pa_2a_3 \qquad \mathbf{T}Pa_1a_3$$
$$\vdots$$
$$\mathbf{T}\neg Pa_3a_1$$

$$\mathbf{T}\exists y Pa_3 y$$
$$\mathbf{T}Pa_3a_4$$
$$\vdots$$

Figure 6.1: Tableau for a formula satisfiable in an infinite domain.

.

6.7 Normal models

[Bos97, Smu68]