Logic in Computer Science

Michael Winter
Brock University

September 19, 2014

Chapter 1

Propositional Logic

The first language we consider is the language of propositional logic. It is based on propositions (or declarative sentences) which can either be true or false. Some examples are:

1. Grass is green.
2. The sky is yellow.
3. Every natural number $n > 2$ is the sum of two prime numbers.
4. If I do not study then I get an F in this course.

Sentence (1) is obviously true, whereas sentences (2) is false (at least on earth). Sentence (3) is the so-called Goldbach conjecture. Nobody actually knows whether that sentence is true or false (it is an open problem in mathematics). Nevertheless, it is either one - true or false. Sentence (4) is remarkable since it built from smaller sentences by using certain constructions. Two two propositions 'I do not study' and 'I get an F in this course' are combined using an 'If ... then ...' construction. Even the proposition 'I do not study' can be considered to be the result of applying a 'not ...' construction to the proposition 'I do study'.

Some examples for sentences that are not propositions are:

- May the force be with you.
- Go, clean up your room!
- How are you today?

The sentences above are formulated in plain English. Such a representation of propositions is not very suitable for a computer. Therefore, we are going to introduce a formal language of propositions. This language can easily be manipulated by a program.

1.1 Syntax

We are going to consider certain declarative sentences as being atomic. For example, the sentences (1)-(3) are atomic. They cannot be decomposed into smaller propositions. We will use the symbol $\perp$ to denote falsehood, and distinct symbols $p, q, r, \ldots$ for arbitrary atomic sentences.

Definition 1 *Let P be a set of propositional variables. The set* Prop *of propositional formulas are those which can be obtained by using the following rules finitely many times:*

1. *Each element $p \in P$ is a propositional formula, i.e., $P \subseteq \mathrm{Prop}$.*
2. *$\perp$ is a propositional formula, i.e, $\perp \in \mathrm{Prop}$.*
3. *If $\varphi \in \mathrm{Prop}$ then $\neg\varphi \in \mathrm{Prop}$.*
4. *If $\varphi_1, \varphi_2 \in \mathrm{Prop}$ then*
   (a) $\varphi_1 \wedge \varphi_2 \in \mathrm{Prop}$ *and*
   (b) $\varphi_1 \vee \varphi_2 \in \mathrm{Prop}$ *and*
   (c) $\varphi_1 \rightarrow \varphi_2 \in \mathrm{Prop}$.

The previous definition defines the set of formulas by giving a set of rules which may be applied to a base set finitely many times. We also say that the set Prop is defined recursively by those rules. A set defined in such a way always provides a principle of induction. In the example of Prop that principle reads as follows. If we want to show that a certain property N is true for all elements in Prop it is sufficient to

Base case: show the property N for the elements in P and the special formula $\perp$;

Induction step I: show the property N for $\neg\varphi$ by assuming that it is already true for φ (induction hypothesis);

Induction step II: show the property N for $\varphi_1 \otimes \varphi_2$ for $\otimes \in \{\wedge, \vee, \rightarrow\}$ by assuming that it is already true for φ_1 and φ_2 (induction hypothesis).

Why is this principle valid? Intuitively we may argue the following: Every formula is constructed by applying the rules finitely many times to the base set. The base case of the induction shows that the property N is true in the base set, and the induction step shows that if we apply any rule N remains to be true.

It is possible to prove that this induction principle is valid in general by relating it to the well-known mathematical induction (on the natural numbers). But this is out of the scope of this lecture.

We adopt certain precedence rules of the logical symbols. $\neg$ binds more tightly than $\wedge$, $\wedge$ tighter than $\vee$, and $\vee$ tighter than $\rightarrow$. For example, the proposition

$$p \wedge q \vee \neg r \rightarrow p$$

has to be read as

$$((p \wedge q) \vee (\neg r)) \rightarrow p.$$

Last but not least, we will use the following abbreviations. We write $\top$ for $\neg\bot$ and $\varphi_1 \leftrightarrow \varphi_2$ for $(\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$.

1.2 Semantics

We want to define what it means for a formula to be valid (or true). Formulas are built from propositional variables representing arbitrary atomic propositions. To determine the validity of the formula we have to replace those variables by actual propositions, which are either true or false. We can eliminate the intermediate step and just consider truth assignments.

Definition 2 *A truth assignment is a function $v : P \rightarrow \mathbb{B}$ from the propositional variables into the set of truth values $\mathbb{B} = \{\mathrm{T}, \mathrm{F}\}$.*

A given truth assignment can be extended to the full set of propositional formulas.

Definition 3 *Let v be a truth assignment. The extension $\bar{v} : \mathrm{Prop} \rightarrow \mathbb{B}$ of v is defined by:*

1. *$\bar{v}(p) = v(p)$ for every $p \in P$.*

2. $\bar{v}(\bot) = \mathrm{F}$.

3. $\bar{v}(\neg\varphi) = \begin{cases} \mathrm{T} & \textit{if } \bar{v}(\varphi) = \mathrm{F}, \\ \mathrm{F} & \textit{otherwise.} \end{cases}$

4. $\bar{v}(\varphi_1 \wedge \varphi_2) = \begin{cases} \mathrm{T} & \textit{if } \bar{v}(\varphi_1) = \mathrm{T} \textit{ and } \bar{v}(\varphi_2) = \mathrm{T}, \\ \mathrm{F} & \textit{otherwise.} \end{cases}$

5. $\bar{v}(\varphi_1 \vee \varphi_2) = \begin{cases} \mathrm{T} & \textit{if } \bar{v}(\varphi_1) = \mathrm{T} \textit{ or } \bar{v}(\varphi_2) = \mathrm{T}, \\ \mathrm{F} & \textit{otherwise.} \end{cases}$

6. $\bar{v}(\varphi_1 \rightarrow \varphi_2) = \begin{cases} \mathrm{F} & \textit{if } \bar{v}(\varphi_1) = \mathrm{T} \textit{ and } \bar{v}(\varphi_2) = \mathrm{F}, \\ \mathrm{T} & \textit{otherwise.} \end{cases}$

v is said to satisfy a formula φ iff[1] $\bar{v}(\varphi) = \mathrm{T}$*. Furthermore, v is said to satisfy a set $\Sigma \subseteq \mathrm{Prop}$ of formulas iff v satisfies φ for all $\varphi \in \Sigma$.*

Notice that v satisfies φ is equivalent to v satisfies $\{\varphi\}$. Based on the previous definition we now may state the main definition of this section.

Definition 4 *Let $\varphi \in \mathrm{Prop}$ be a formula, and $\Sigma \subseteq \mathrm{Prop}$ be a set of formulas. Σ tautologically implies φ (written $\Sigma \models \varphi$) iff every truth assignment that satisfies Σ also satisfies φ.*

Consider the special case where $\Sigma = \emptyset$. In that case every truth assignment satisfies all elements in Σ (there are none). Hence we are left with $\Sigma \models \varphi$ iff every truth satisfies φ. In this situation we say that φ is a tautology or φ is valid (written $\models \varphi$). On the other hand, we call φ satisfiable if there is a truth assignment satisfying φ. Obviously, satisfiability is weaker concept since it requires just the existence of one particular truth assignment whereas validity quantifies over all truth assignments. Both concepts are closely related shown by the following lemma.

Lemma 5 *Let φ be a formula. Then φ is satisfiable iff $\neg\varphi$ is not valid.*

Proof. $\Rightarrow$: Assume φ is satisfiable. Then there is a truth assignment v with $\bar{v}(\varphi) = \mathrm{T}$. We conclude $\bar{v}(\neg\varphi) = \mathrm{F}$, and, hence, $\neg\varphi$ is not valid.

$\Leftarrow$: If $\neg\varphi$ is not valid then there is a truth assignment v with $\bar{v}(\neg\varphi) = \mathrm{F}$. We conclude $\bar{v}(\varphi) = \mathrm{T}$, and, hence, φ is satisfiable. □

To determine the truth value of a given formula and a truth assignment v we just need to know the values of v for a finite set of propositional variables.

[1] We use the abbreviation iff for 'if and only if'.

Lemma 6 *Let $\varphi \in \mathrm{Prop}$ be a formula, and v_1 and v_2 be truth assignments which coincide on all propositional variables that occur in φ, i.e., $v_1(p) = v_2(p)$ for all $p \in P$ in φ. Then $\bar{v}_1(\varphi) = \bar{v}_2(\varphi)$.*

Proof. The proof is done by induction. If φ is a propositional variable we get

$$\bar{v}_1(\varphi) = v_1(p) = v_2(p) = \bar{v}_2(\varphi)$$

by the assumption on v_1 and v_2. If $\varphi = \perp$ we get immediately

$$\bar{v}_1(\varphi) = \mathrm{F} = \bar{v}_2(\varphi).$$

Now, assume $\varphi = \neg\varphi'$ for a formula $\varphi' \in \mathrm{Prop}$. Every propositional variable occurring in φ' occurs in a formula of Σ (namely φ). Therefore, $\bar{v}_1(\varphi') = \bar{v}_2(\varphi')$ by the induction hypothesis. We conclude

$$\begin{aligned}
\bar{v}_1(\varphi) &= \begin{cases} \mathrm{T} & \text{if } \bar{v}_1(\varphi') = \mathrm{F}, \\ \mathrm{F} & \text{otherwise.} \end{cases} \\
&= \begin{cases} \mathrm{T} & \text{if } \bar{v}_2(\varphi') = \mathrm{F}, \\ \mathrm{F} & \text{otherwise.} \end{cases} \\
&= \bar{v}_2(\varphi).
\end{aligned}$$

The remaining cases are shown similarly. □

The last lemma immediately implies the following corollary.

Corollary 7 *Let $\Sigma \subseteq \mathrm{Prop}$ be a set of formulas, and v_1 and v_2 be truth assignments which coincide on all propositional variables that occur in formulas of Σ. Then $\bar{v}_1(\varphi) = \bar{v}_2(\varphi)$ for all formulas $\varphi \in \Sigma$.*

Notice, in the situation of the last corollary v_1 satisfies Σ iff v_2 satisfies Σ.

If $\Sigma = \{\varphi_1, \ldots, \varphi_n\}$ is finite we may use truth tables to check whether or not

$$\{\varphi_1, \ldots, \varphi_n\} \models \varphi.$$

Let $V = \{p_1, \ldots, p_k\}$ be the set of all propositional variables occurring in $\Sigma \cup \{\varphi\}$. By the last lemma we just have to consider truth assignments that differ on at least one element of V. There are exactly 2^k different truth assignments. These truth assignments and their extension to the formulas in $\Sigma \cup \{\varphi\}$ can be listed in a table.

Example 8 *Let $\Sigma = \{\neg q, \neg p \to q\}$ and $\varphi = p$. In this case $V = \{p, q\}$ so that we have to consider $2^2 = 4$ different truth assignments leading to the initial table*

p	q
T	T
T	F
F	T
F	F

Each row corresponds to one truth assignment (To be precise, each row corresponds to all truth assignments with the given values for p and q). This table can now be extended to formulas $\Sigma \cup \{\varphi\}$:

p	q	$\neg p$	$\neg q$	$\neg p \to q$
T	T	F	F	T
T	F	F	T	T
F	T	T	F	T
F	F	T	T	F

Just the truth assignment corresponding to row number 2 satisfies Σ (entry T *in that row for each formula in Σ). The entry for $\varphi(= p)$ in row 2 is also* T *so that we can conclude $\Sigma \models \varphi$.*

Truth tables can also be used to visualize the logical connectives $\neg, \wedge, \vee, \to$ as well as the derived connective $\leftrightarrow$. The truth table is as follows:

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \to q$	$p \leftrightarrow q$
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T

If we want to determine the truth value of a long formula we will occasionally use a more compact form of a truth table. Here we will denote the truth value of subformulas just beneath the operator symbol. The following example shows that the formula $((p \vee q) \wedge ((p \to r) \wedge (q \to r))) \to r$ is a tautology.

p	q	r	$(($p$	$\vee q)$	$\wedge((p \to r)$	$\wedge(q \to r)))$		$\to r$
T	T	T	T	T	T	T	T	T
T	T	F	T	F	F	F	F	T
T	F	T	T	T	T	T	T	T
T	F	F	T	F	F	F	T	T
F	T	T	T	T	T	T	T	T
F	T	F	T	F	T	F	F	T
F	F	T	F	F	T	T	T	T
F	F	F	F	F	T	T	T	T

Theorem 9 *Let* $\Sigma \subseteq \mathrm{Prop}$ *be a finite set of formulas, and* $\varphi \in \mathrm{Prop}$ *be a formula. Then the statement* $\Sigma \models \varphi$ *is decidable.*

Proof. The truth table method serves as a decision algorithm. □

Let us consider an example that show how a problem can be formulated and solved using the language of propositions and truth tables.

Example 10 *John, a zoologist, is searching for a rare bird named Oona on an archipelago of islands inhabited by knights and knaves. Knights always tell the truth, and knaves always lie. On the first island John meets two inhabitants, Alan and Robert. He asks both wether Oona is on that island. Their answers are:*

Alan: *'If Robert and I are both knights, then Oona is on the island.'*

Robert: *'If Alan and I are both knights, then Oona is on the island.'*

John is completely puzzled and needs our help.

Take A to mean 'Alan is a knight', R to mean 'Robert is a knight', and O to mean 'Oona is on the island'. What Alan said can be translated as $A \wedge R \to O$*. This is, of course, true if and only if Alan is a knight so that we get* $A \leftrightarrow (A \wedge R \to O)$*. Similarly we get* $R \leftrightarrow (A \wedge R \to O)$ *so that the formula*

$$(A \leftrightarrow (A \wedge R \to O)) \wedge (R \leftrightarrow (A \wedge R \to O))$$

is true in the current situation. Consequently, we are interested in those truth assignments v (to be precise in the value $v(O)$) that satisfy this formula.

A	R	O	$(A \leftrightarrow$	$(A \wedge R$	$\to O))$	$\wedge$	$(R \leftrightarrow$	$(A \wedge R$	$\to O))$
T	T	T	T	T	T	T	T	T	T
T	T	F	F	T	F	F	F	T	F
T	F	T	T	F	T	F	F	F	T
T	F	F	T	F	T	F	F	F	T
F	T	T	F	F	T	F	T	F	T
F	T	F	F	F	T	F	T	F	T
F	F	T	F	F	T	F	F	F	T
F	F	F	F	F	T	F	F	F	T

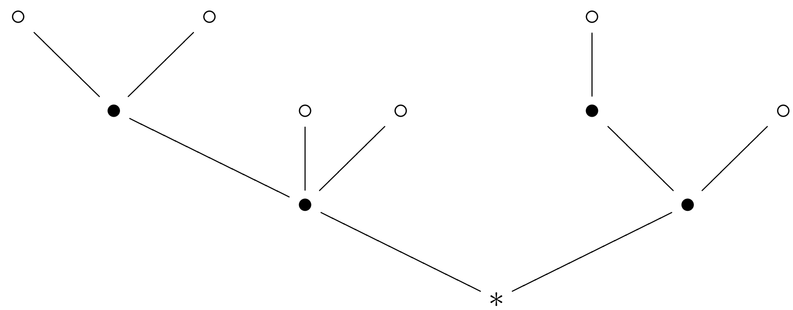
The truth table shows that there is one truth assignment satisfying the formula. The truth value for O is T *meaning that Oona is on the island. In addition, we are also able to derive that Alan and Robert are knights.*

1.3 Natural Deduction

In this section we want to investigate a formal calculus for reasoning about propositions. This calculus, called natural deduction, uses proof rules, which allow to infer formulas from other formulas. By applying these rules in succession, we may infer a conclusion from a finite set of premises. Suppose a set $\{\varphi_1, \ldots, \varphi_n\}$ of formulas is given. We start to apply a proof rule of the calculus to certain elements of the set of premises generating a new formula ψ_1. In the next step we apply a rule to certain elements of the set $\{\varphi_1, \ldots, \varphi_n, \psi_1\}$ generating a new formula ψ_2. Continuous application of the rules to a growing set of formulas will finally end in the intended result ψ - the conclusion. In this case we are successful in deriving ψ from the set of premises, and we will denote that by

$$\varphi_1, \ldots, \varphi_n \vdash \psi.$$

Some rules allow us to make temporary assumptions, i.e., such a rule enlarges the set of premises temporarily. The derivation itself is actually a tree with the premises and temporary assumptions we as leafs, applications of rules as nodes, and the conclusion as the root. The skeleton of a proof may look as follows ($\circ$ denotes a premises or assumption, $\bullet$ denotes an intermediate formula generated by a certain rule, $*$ denotes the conclusion):



The proof rules of the natural deduction calculus are grouped by the logical operators and constants of the propositional language. For each operator we have introduction and elimination rules, and for the constant $\perp$ a single rule. Introduction rules are used to infer a formula containing the operator as the outermost symbol. Elimination rules are used to derive other properties from a formula containing the operator. We want to discuss four rules in detail.

And introduction: This rule is used to infer a formula of the form $\varphi \wedge \psi$. It seems obvious that we are allowed to conclude this formula if we have already concluded both φ and ψ. Therefore, the rule becomes

$$\frac{\varphi \quad \psi}{\varphi \wedge \psi}\wedge\mathrm{I}.$$

The rule is binary, i.e, it has to be applied to two subtrees, the first deriving φ, the second deriving ψ. To the right of the line we denote the name of the rule (I means introduction, E means elimination).

And elimination: This rule is used to infer other properties from a formula of the form $\varphi \wedge \psi$. We have to such elimination rules given by

$$\frac{\varphi \wedge \psi}{\varphi}\wedge\mathrm{E1} \qquad \frac{\varphi \wedge \psi}{\psi}\wedge\mathrm{E2}.$$

Implication introduction: In order to infer a formula of the form $\varphi \to \psi$ we are allowed to temporarily make the assumption φ. From this assumption we have to derive the formula ψ. If we are successful we denote this derivation by

$$\begin{array}{c}\varphi\\ \vdots\\ \psi.\end{array}$$

In that case we are allowed to conclude $\varphi \to \psi$. In addition, the temporary assumption φ is not longer needed. We are allowed to discard it, denoted by $[\varphi]$. The rule finally is:

$$\dfrac{\begin{array}{c}[\varphi]\\ \vdots\\ \psi\end{array}}{\varphi \to \psi} \to\mathrm{I}.$$

PBC: This rule is neither an introduction nor an elimination rule. PBCis an abbreviation for proof by contradiction. If we are able to show that $\neg\varphi$ leads to a contradiction, the formula $\perp$, then we are allowed to conclude φ. The rule reads:

$$\dfrac{\begin{array}{c}[\neg\varphi]\\ \vdots\\ \perp\end{array}}{\varphi} \mathrm{PBC}.$$

Table 1.1 lists the rules of natural deduction for propositional logic.

Example 11 (Example 8 revisited) *In Example 8 we have shown using truth tables that $\neg p \to q, \neg q \models p$. The following derivation verifies the related property $\neg p \to q, \neg q \vdash p$:*

$$\dfrac{\dfrac{\dfrac{\neg p \to q \quad [\neg p]^1}{q} \to E \quad \neg q}{\perp} \neg E}{p} PBC^1$$

We use the operator $\leftrightarrow$ as an abbreviation, i.e., the formula $\varphi \leftrightarrow \psi$ means $(\varphi \to \psi) \wedge (\psi \to \varphi)$. Occasionally we will use the following (derived) rule for this operator.

	introduction rule	elimination rule	
$\leftrightarrow$	$\dfrac{\varphi \to \psi \quad \psi \to \varphi}{\varphi \leftrightarrow \psi} \leftrightarrow\mathrm{I}$	$\dfrac{\varphi \leftrightarrow \psi}{\varphi \to \psi} \leftrightarrow\mathrm{E1}$	$\dfrac{\varphi \leftrightarrow \psi}{\psi \to \varphi} \leftrightarrow\mathrm{E2}$

	introduction rule	elimination rule
$\wedge$	$\dfrac{\varphi \quad \psi}{\varphi \wedge \psi}\ \wedge\mathrm{I}$	$\dfrac{\varphi \wedge \psi}{\varphi}\ \wedge\mathrm{E1} \qquad \dfrac{\varphi \wedge \psi}{\psi}\ \wedge\mathrm{E2}$
$\vee$	$\dfrac{\varphi}{\varphi \vee \psi}\ \vee\mathrm{I1} \qquad \dfrac{\psi}{\varphi \vee \psi}\ \vee\mathrm{I2}$	$\dfrac{\varphi \vee \psi \quad \begin{array}{c}[\varphi]\\ \vdots \\ \chi\end{array} \quad \begin{array}{c}[\psi]\\ \vdots \\ \chi\end{array}}{\chi}\ \vee\mathrm{E}$
$\rightarrow$	$\dfrac{\begin{array}{c}[\varphi]\\ \vdots \\ \psi\end{array}}{\varphi \rightarrow \psi}\ \rightarrow\mathrm{I}$	$\dfrac{\varphi \quad \varphi \rightarrow \psi}{\psi}\ \rightarrow\mathrm{E}$
$\neg$	$\dfrac{\begin{array}{c}[\varphi]\\ \vdots \\ \bot\end{array}}{\neg\varphi}\ \neg\mathrm{I}$	$\dfrac{\varphi \quad \neg\varphi}{\bot}\ \neg\mathrm{E}$
PBC	$\dfrac{\begin{array}{c}[\neg\varphi]\\ \vdots \\ \bot\end{array}}{\varphi}\ \mathrm{PBC}$	

Table 1.1: Rules of natural deduction for propositional logic

Example 12 (Example 10 revisited) *In Example 10 we wanted to know whether Oona is on that particular island. We were able to use the two propositions $A \leftrightarrow (A \wedge R \to O)$ and $R \leftrightarrow (A \wedge R \to O)$, which we derived from the statements of the inhabitants. Actually, we verified using a truth table that*

$$A \leftrightarrow (A \wedge R \to O), R \leftrightarrow (A \wedge R \to O) \models O.$$

Again, we want to provide the corresponding derivation verifying

$$A \leftrightarrow (A \wedge R \to O), R \leftrightarrow (A \wedge R \to O) \vdash O.$$

First, we give a derivation for $A \leftrightarrow (A \wedge R \to O) \vdash A$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[A \wedge R]^1}{A}\wedge E1 \qquad [\neg A]^2}{\bot}\neg E}{O}PBC}{A \wedge R \to O}\to I^1 \qquad \dfrac{A \leftrightarrow (A \wedge R \to O)}{(A \wedge R \to O) \to A}\leftrightarrow E2}{A}\to E \qquad [\neg A]^2}{\bot}\neg E}{A}PBC^2$$

$R \leftrightarrow (A \wedge R \to O) \vdash R$ can be derived analogously. In the final derivation below we replaced those derivations by $\begin{array}{c}F_1\\ \vdots\\ A\end{array}$ and $\begin{array}{c}F_2\\ \vdots\\ R\end{array}$, respectively.

$$\dfrac{\dfrac{\begin{array}{c}F_1\\ \vdots\\ A\end{array} \quad \begin{array}{c}F_2\\ \vdots\\ R\end{array}}{A \wedge R}\wedge I \qquad \dfrac{\begin{array}{c}F_1\\ \vdots\\ A\end{array} \quad \dfrac{A \leftrightarrow (A \wedge R \to O)}{A \to (A \wedge R \to O)}\leftrightarrow E2}{A \wedge R \to O}\to E}{O}\to E$$

The property of natural deduction we are interested in is its soundness.

Theorem 13 (Soundness) *Let $\varphi_1, \ldots, \varphi_n$ and ψ be propositional formulas. If $\varphi_1, \ldots, \varphi_n \vdash \psi$ is valid, then $\varphi_1, \ldots, \varphi_n \models \psi$ holds.*

Proof. The proof is done by induction on the derivation $\varphi_1, \ldots, \varphi_n \vdash \psi$.

(Base case): In this case the proof is just a premises, i.e., we have $\psi \in \{\varphi_1, \ldots, \varphi_n\}$. Assume v is a truth assignment satisfying the set of premises, i.e., $\bar{v}(\varphi_i) = \mathrm{T}$ for all $i \in \{1, \dots, n\}$. Then we conclude $\bar{v}(\psi) = \mathrm{T}$, and, hence $\varphi_1, \dots, \varphi_n \models \psi$.

(Induction step): We distinguish several cases according the last rule applied.

$\wedge$**I** : In this case $\psi = \psi_1 \wedge \psi_2$ and we have derivations $\varphi_1, \dots, \varphi_n \vdash \psi_1$ and $\varphi_1, \dots, \varphi_n \vdash \psi_2$. From the induction hypothesis we get $\varphi_1, \dots, \varphi_n \models \psi_1$ and $\varphi_1, \dots, \varphi_n \models \psi_2$. Now, assume v is a truth assignment satisfying the set of premises. Then we have $\bar{v}(\psi_1) = \mathrm{T}$ and $\bar{v}(\psi_2) = \mathrm{T}$. We conclude $\bar{v}(\psi) = \mathrm{T}$, and, hence $\varphi_1, \dots, \varphi_n \models \psi$.

$\wedge$**E1** : In this case we have a derivation $\varphi_1, \dots, \varphi_n \vdash \psi \wedge \psi'$ for some formula ψ'. Now, assume v is a truth assignment satisfying the set of premises. By the induction hypothesis we conclude $\bar{v}(\psi \wedge \psi') = \mathrm{T}$, which implies $\bar{v}(\psi) = \mathrm{T}$, and, hence, $\varphi_1, \dots, \varphi_n \models \psi$.

$\wedge$**E2** : Analogously to $\wedge$E1.

$\vee$**I1** : In this case $\psi = \psi_1 \vee \psi_2$ and we have a derivation $\varphi_1, \dots, \varphi_n \vdash \psi_1$. Now, assume v is a truth assignment satisfying the set of premises. By the induction hypothesis we conclude $\bar{v}(\psi_1) = \mathrm{T}$, which implies $\bar{v}(\psi) = \mathrm{T}$, and, hence, $\varphi_1, \dots, \varphi_n \models \psi$.

$\vee$**I2** : Analogously to $\vee$I1.

$\vee$**E** : In this case we have the following derivations

$$\begin{aligned} &\varphi_1, \dots, \varphi_n &&\vdash \psi_1 \vee \psi_2 \\ &\varphi_1, \dots, \varphi_n, \psi_1 &&\vdash \psi \\ &\varphi_1, \dots, \varphi_n, \psi_2 &&\vdash \psi \end{aligned}$$

Now, assume v is a truth assignment satisfying $\{\varphi_1, \dots, \varphi_n\}$. We get $\bar{v}(\psi_1 \vee \psi_2) = \mathrm{T}$ from the induction hypothesis, i.e., either $\bar{v}(\psi_1) = \mathrm{T}$ or $\bar{v}(\psi_2) = \mathrm{T}$. in the first case we conclude that v satisfies $\{\varphi_1, \dots, \varphi_n, \psi_1\}$. Using the induction hypothesis again we get $\bar{v}(\psi) = \mathrm{T}$. If $\bar{v}(\psi_2) = \mathrm{T}$ we conclude $\bar{v}(\psi) = \mathrm{T}$ analogously.

$\rightarrow$**I** : In this case $\psi = \psi_1 \rightarrow \psi_2$ and we have a derivation $\varphi_1, \dots, \varphi_n, \psi_1 \vdash \psi_2$. Now, assume v is a truth assignment satisfying $\{\varphi_1, \dots, \varphi_n\}$. If v also satisfies ψ_1 we conclude $\bar{v}(\psi_2) = \mathrm{T}$ from the induction hypothesis, and, hence, $\bar{v}(\psi) = \mathrm{T}$. If v does not satisfy ψ_1 we immediately get $\bar{v}(\psi) = \mathrm{T}$.

$\to$**E** : In this case we have derivations $\varphi_1, \ldots, \varphi_n \vdash \psi'$ and $\varphi_1, \ldots, \varphi_n \vdash \psi' \to \psi$ for some formula ψ'. Now, assume v is a truth assignment satisfying the set of premises. From the induction hypothesis we get $\bar{v}(\psi') = \mathrm{T}$ and $\bar{v}(\psi' \to \psi) = \mathrm{T}$. We conclude $\bar{v}(\psi) = \mathrm{T}$.

$\neg$**I** : In this case $\psi = \neg\psi'$ and we have a derivation $\varphi_1, \ldots, \varphi_n, \psi' \vdash \bot$. Now, assume v is a truth assignment satisfying $\{\varphi_1, \ldots, \varphi_n\}$. If v also satisfies ψ' we conclude $\bar{v}(\bot) = \mathrm{T}$ from the induction hypothesis. The last statement is a contradiction so that we conclude $\bar{v}(\psi') = \mathrm{F}$, and, hence, $\bar{v}(\psi) = \mathrm{T}$.

$\neg$**E** : In this case $\psi = \bot$ and we have derivations $\varphi_1, \ldots, \varphi_n \vdash \psi'$ and $\varphi_1, \ldots, \varphi_n \vdash \neg\psi'$ for some formula ψ'. Now, assume v is a truth assignment satisfying the set of premises. From the induction hypothesis we get $\bar{v}(\psi') = \mathrm{T}$ and $\bar{v}(\neg\psi') = \mathrm{T}$. This is a contradiction so that such a v does not exist showing $\varphi_1, \ldots, \varphi_n \models \bot$.

PBC : In this case we have a derivation $\varphi_1, \ldots, \varphi_n, \neg\psi \vdash \bot$. Now, assume v is a truth assignment satisfying $\{\varphi_1, \ldots, \varphi_n\}$. If v does not satisfy ψ we conclude $\bar{v}(\bot) = \mathrm{T}$ from the induction hypothesis. The last statement is a contradiction so that we conclude $\bar{v}(\psi) = \mathrm{T}$.

This completes the proof. □

In the remainder of this section we want to show completeness of the calculus. Assuming that $\varphi_1, \ldots, \varphi_n \models \psi$ holds, the proof proceeds in three steps:

1. Show that $\models \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$ holds.
2. Show that $\vdash \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$.
3. Show that $\varphi_1, \ldots, \varphi_n \vdash \psi$.

The first and the last step are not very hard and shown in the following lemma.

Lemma 14 *Let $\varphi_1, \ldots, \varphi_n$ and ψ be propositional formulas. Then we have:*

1. $\varphi_1, \ldots, \varphi_n \models \psi$ *iff* $\models \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$.
2. $\varphi_1, \ldots, \varphi_n \vdash \psi$ *iff* $\vdash \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$.

Proof. In this proof we will denote the formula $\varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$ by χ.

1. $\Rightarrow$: Assume v is a truth assignment. If v satisfies the set of formulas $\{\varphi_1, \ldots, \varphi_n\}$ we conclude $\bar{v}(\psi) = \mathrm{T}$, and, hence, $\bar{v}(\chi) = \mathrm{T}$. If there is an i with $\bar{v}(\varphi_i) = \mathrm{F}$ we immediately conclude $\bar{v}(\chi) = \mathrm{T}$.

   $\Leftarrow$: Assume v is a truth assignment with $\bar{v}(\varphi_i) = \mathrm{T}$ for all $i \in \{1, \ldots, n\}$. Since χ is a tautology we have $\bar{v}(\chi) = \mathrm{T}$. From the fact that χ is a chain of implications we conclude $\bar{v}(\psi)$, and, hence, $\varphi_1, \ldots, \varphi_n \models \psi$.

2. $\Rightarrow$: Assume there is a derivation $\varphi_1, \ldots, \varphi_n \vdash \psi$. By applying the rule $\to$I n-times we get a derivation $\vdash \chi$.

   $\Leftarrow$: Assume there is a derivation $\vdash \chi$. By applying the rule $\to$E n-times with the formula φ_i in the ith application we get a derivation $\varphi_1, \ldots, \varphi_n \vdash \psi$. $\square$

In order to verify Step 2. we need derivations of some common formulas. They are collected in the next lemma.

Lemma 15 *Let $\varphi, \varphi_1, \varphi_2 \in \mathrm{Prop}$ be propositional formulas. Then we have:*

1. $\vdash \varphi \leftrightarrow \neg\neg\varphi$.
2. $\vdash \neg\varphi_1 \wedge \neg\varphi_2 \leftrightarrow \neg(\varphi_1 \vee \varphi_2)$.
3. $\vdash \neg\varphi_1 \vee \neg\varphi_2 \leftrightarrow \neg(\varphi_1 \wedge \varphi_2)$.
4. $\vdash \varphi \vee \neg\varphi$.
5. $\vdash (\varphi_1 \to \varphi_2) \leftrightarrow \neg\varphi_1 \vee \varphi_2$.
6. $\vdash \neg(\varphi_1 \to \varphi_2) \leftrightarrow \varphi_1 \wedge \neg\varphi_2$.

Proof. In each case we give derivations for both implications.

1.

$$\dfrac{\dfrac{\dfrac{[\neg\varphi]^1 \quad [\varphi]^2}{\bot}\,\neg\mathrm{E}}{\neg\neg\varphi}\,\neg\mathrm{I}^1}{\varphi \to \neg\neg\varphi}\,\to\mathrm{I}^2 \qquad \dfrac{\dfrac{\dfrac{[\neg\neg\varphi]^2 \quad [\neg\varphi]^1}{\bot}\,\neg\mathrm{E}}{\varphi}\,\mathrm{PBC}^1}{\neg\neg\varphi \to \varphi}\,\to\mathrm{I}^2$$

2.

$$
\dfrac{\dfrac{\dfrac{[\varphi_1 \vee \varphi_2]^2 \qquad \dfrac{\dfrac{[\neg\varphi_1 \wedge \neg\varphi_2]^3}{\neg\varphi_1}\,\wedge\mathrm{E1} \qquad [\varphi_1]^1}{\bot}\,\neg\mathrm{E} \qquad \dfrac{\dfrac{[\neg\varphi_1 \wedge \neg\varphi_2]^3}{\neg\varphi_2}\,\wedge\mathrm{E2} \qquad [\varphi_2]^1}{\bot}\,\neg\mathrm{E}}{\bot}\,\vee\mathrm{E}^1}{\neg(\varphi_1 \vee \varphi_2)}\,\neg\mathrm{I}^2}{\neg\varphi_1 \wedge \neg\varphi_2 \to \neg(\varphi_1 \vee \varphi_2)}\,\to\mathrm{I}^3
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{[\neg(\varphi_1 \vee \varphi_2)]^3 \qquad \dfrac{[\varphi_1]^1}{\varphi_1 \vee \varphi_2}\,\vee\mathrm{I1}}{\bot}\,\neg\mathrm{E}}{\neg\varphi_1}\,\neg\mathrm{I}^1 \qquad \dfrac{\dfrac{[\neg(\varphi_1 \vee \varphi_2)]^3 \qquad \dfrac{[\varphi_2]^2}{\varphi_1 \vee \varphi_2}\,\vee\mathrm{I2}}{\bot}\,\neg\mathrm{E}}{\neg\varphi_2}\,\neg\mathrm{I}^2}{\neg\varphi_1 \wedge \neg\varphi_2}\,\wedge\mathrm{I}}{\neg(\varphi_1 \vee \varphi_2) \to \neg\varphi_1 \wedge \neg\varphi_2}\,\to\mathrm{I}^3
$$

3.

$$
\dfrac{\dfrac{\dfrac{[\neg\varphi_1 \vee \neg\varphi_2]^3 \qquad \dfrac{[\neg\varphi_1]^1 \qquad \dfrac{[\varphi_1 \wedge \varphi_2]^2}{\varphi_1}\,\wedge\mathrm{E1}}{\bot}\,\neg\mathrm{E} \qquad \dfrac{[\neg\varphi_2]^1 \qquad \dfrac{[\varphi_1 \wedge \varphi_2]^2}{\varphi_2}\,\wedge\mathrm{E1}}{\bot}\,\neg\mathrm{E}}{\bot}\,\vee\mathrm{E}^1}{\neg(\varphi_1 \wedge \varphi_2)}\,\neg\mathrm{I}^2}{\neg\varphi_1 \vee \neg\varphi_2 \to \neg(\varphi_1 \wedge \varphi_2)}\,\to\mathrm{I}^3
$$

For the second derivation we first show that $\neg(\neg\varphi_1 \vee \neg\varphi_2) \vdash \varphi_i$ for $i = 1, 2$.

$$
\dfrac{\dfrac{\neg(\neg\varphi_1 \vee \neg\varphi_2) \qquad \dfrac{[\neg\varphi_i]^1}{\neg\varphi_1 \vee \neg\varphi_2}\,\vee\mathrm{Ii}}{\bot}\,\neg\mathrm{E}}{\varphi_i}\,\mathrm{PBC}^1
$$

Using the derivation above we get

$$
\dfrac{\dfrac{\dfrac{[\neg(\varphi_1 \wedge \varphi_2)]^2 \qquad \dfrac{\begin{matrix}[\neg(\neg\varphi_1 \vee \neg\varphi_2)]^1 \\ \vdots \\ \varphi_1\end{matrix} \qquad \begin{matrix}[\neg(\neg\varphi_1 \vee \neg\varphi_2)]^1 \\ \vdots \\ \varphi_2\end{matrix}}{\varphi_1 \wedge \varphi_2}\,\wedge\mathrm{I}}{\bot}\,\neg\mathrm{E}}{\neg\varphi_1 \vee \neg\varphi_2}\,\mathrm{PBC}^1}{\neg(\varphi_1 \wedge \varphi_2) \to \neg\varphi_1 \vee \neg\varphi_2}\,\to\mathrm{I}^2
$$

4.

$$\dfrac{\dfrac{[\neg(\varphi \vee \neg\varphi)]^2 \qquad \dfrac{\dfrac{\dfrac{[\neg(\varphi \vee \neg\varphi)]^2 \quad \dfrac{[\varphi]^1}{\varphi \vee \neg\varphi}\,\vee\mathrm{I1}}{\bot}\,\neg\mathrm{E}}{\neg\varphi}\,\neg\mathrm{I}^1}{\varphi \vee \neg\varphi}\,\vee\mathrm{I2}}{\bot}\,\neg\mathrm{E}}{\varphi \vee \neg\varphi}\ \mathrm{PBC}^2$$

5. The first derivation uses 4.

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \varphi_1 \vee \neg\varphi_1\end{array} \qquad \dfrac{\dfrac{[\varphi_1 \to \varphi_2]^2 \quad [\varphi_1]^1}{\varphi_2}\,\to\mathrm{E}}{\neg\varphi_1 \vee \varphi_2}\,\vee\mathrm{I2} \qquad \dfrac{[\neg\varphi_1]^1}{\neg\varphi_1 \vee \varphi_2}\,\vee\mathrm{I1}}{\neg\varphi_1 \vee \varphi_2}\,\vee\mathrm{E}^1}{(\varphi_1 \to \varphi_2) \to \neg\varphi_1 \vee \varphi_2}\,\to\mathrm{I}^2$$

$$\dfrac{\dfrac{\dfrac{[\neg\varphi_1 \vee \varphi_2]^3 \qquad \dfrac{\dfrac{[\neg\varphi_1]^1 \quad [\varphi_1]^2}{\bot}\,\neg\mathrm{E}}{\varphi_2}\ \mathrm{PBC} \qquad [\varphi_2]^1}{\varphi_2}\,\vee\mathrm{E}^1}{\varphi_1 \to \varphi_2}\,\to\mathrm{I}^2}{\neg\varphi_1 \vee \varphi_2 \to (\varphi_1 \to \varphi_2)}\,\to\mathrm{I}^3$$

6.

$$\dfrac{\dfrac{\dfrac{\dfrac{[\neg(\varphi_1 \to \varphi_2)]^4 \qquad \dfrac{\dfrac{\dfrac{[\neg\varphi_1]^2 \quad [\varphi_1]^1}{\bot}\,\neg\mathrm{E}}{\varphi_2}\ \mathrm{PBC}}{\varphi_1 \to \varphi_2}\,\to\mathrm{I}^1}{\bot}\,\neg\mathrm{E}}{\varphi_1}\ \mathrm{PBC}^2 \qquad \dfrac{\dfrac{[\neg(\varphi_1 \to \varphi_2)]^4 \quad \dfrac{[\varphi_2]^3}{\varphi_1 \to \varphi_2}\,\to\mathrm{I}}{\bot}\,\neg\mathrm{E}}{\neg\varphi_2}\,\neg\mathrm{I}^3}{\varphi_1 \wedge \neg\varphi_2}\,\wedge\mathrm{I}}{\neg(\varphi_1 \to \varphi_2) \to \varphi_1 \wedge \neg\varphi_2}\,\to\mathrm{I}^4$$

$$\dfrac{\dfrac{\dfrac{\dfrac{[\varphi_1 \wedge \neg\varphi_2]^2}{\neg\varphi_2}\,\wedge\mathrm{E2} \qquad \dfrac{[\varphi_1 \to \varphi_2]^1 \quad \dfrac{[\varphi_1 \wedge \neg\varphi_2]^2}{\varphi_1}\,\wedge\mathrm{E1}}{\varphi_2}\,\to\mathrm{E}}{\bot}\,\neg\mathrm{E}}{\neg(\varphi_1 \to \varphi_2)}\,\neg\mathrm{I}^1}{\varphi_1 \wedge \neg\varphi_2 \to \neg(\varphi_1 \to \varphi_2)}\,\to\mathrm{I}^2$$

This completes the proof. $\square$

The basic idea of Step 2. is the following. For each line of the truth table we going to construct a proof. If the formula has n propositional variables, the truth table has 2^n rows. We assemble those 2^n proofs into a single proof.

Lemma 16 *Let φ be a formula with propositional variables among $p_1, \ldots, p_n$, and v be a truth assignment. Define*

$$\hat{p}_i := \begin{cases} p_i & \text{if } v(p_i) = \mathrm{T} \\ \neg p_i & \text{if } v(p_i) = \mathrm{F}. \end{cases}$$

Then we have:

1. $\hat{p}_1, \ldots, \hat{p}_n \vdash \varphi$ *if* $\bar{v}(\varphi) = \mathrm{T}$.
2. $\hat{p}_1, \ldots, \hat{p}_n \vdash \neg\varphi$ *if* $\bar{v}(\varphi) = \mathrm{F}$.

Proof. The proof is done by induction on the structure of φ.

$\varphi = p$: In this case the assertions $p \vdash p$ and $\neg p \vdash \neg p$ are trivial.

$\varphi = \bot$: In this case we know $\bar{v}(\varphi) = \mathrm{F}$. We conclude

$$\frac{[\bot]}{\neg\bot}\ \neg\mathrm{I}$$

$\varphi = \neg\varphi'$: The propositional variables in φ' are those from φ. We distinguish two cases. If $\bar{v}(\varphi) = \mathrm{T}$, then $\bar{v}(\varphi') = \mathrm{F}$. By the induction hypothesis there is a derivation $\hat{p}_1, \ldots, \hat{p}_n \vdash \neg\varphi'$. Since $\neg\varphi' = \varphi$ we are done. If $\bar{v}(\varphi) = \mathrm{F}$, then $\bar{v}(\varphi') = \mathrm{T}$. By the induction hypothesis there is a derivation $\hat{p}_1, \ldots, \hat{p}_n \vdash \varphi'$. We conclude

$$\dfrac{\begin{matrix}\text{Lemma 15(1)} \\ \vdots \\ \varphi' \to \neg\neg\varphi'\end{matrix} \qquad \begin{matrix}\vdots \\ \varphi'\end{matrix}}{\neg\neg\varphi'}\ \to\mathrm{E}$$

which is a derivation $\hat{p}_1, \ldots, \hat{p}_n \vdash \neg\varphi$.

$\varphi = \varphi_1 \wedge \varphi_2$: If $\bar{v}(\varphi) = \mathrm{T}$, then $\bar{v}(\varphi_1) = \mathrm{T}$ and $\bar{v}(\varphi_2) = \mathrm{T}$. By the induction hypothesis we get derivations $\hat{p}_1, \dots, \hat{p}_n \vdash \varphi_1$ and $\hat{p}_1, \dots, \hat{p}_n \vdash \varphi_2$. We combine those derivations using $\wedge$I:

$$\dfrac{\begin{matrix}\vdots \\ \varphi_1\end{matrix} \quad \begin{matrix}\vdots \\ \varphi_2\end{matrix}}{\varphi_1 \wedge \varphi_2}\ \wedge\mathrm{I}$$

If $\bar{v}(\varphi) = \mathrm{F}$, then $\bar{v}(\varphi_1) = \mathrm{F}$ or $\bar{v}(\varphi_2) = \mathrm{F}$. Assume, w.l.o.g., $\bar{v}(\varphi_1) = \mathrm{F}$. From the induction hypothesis we get a derivation $\hat{p}_1, \dots, \hat{p}_n \vdash \neg\varphi_1$ so that we conclude

$$\dfrac{\begin{matrix}\text{Lemma 15(3)} \\ \vdots \\ \neg\varphi_1 \vee \neg\varphi_2 \to \neg(\varphi_1 \wedge \varphi_2)\end{matrix} \qquad \dfrac{\begin{matrix}\vdots \\ \neg\varphi_1\end{matrix}}{\neg\varphi_1 \vee \neg\varphi_2}\ \vee\mathrm{I}1}{\neg(\varphi_1 \wedge \varphi_2)}\ \to\mathrm{E}$$

$\varphi = \varphi_1 \vee \varphi_2$: If $\bar{v}(\varphi) = \mathrm{T}$ then $\bar{v}(\varphi_1) = \mathrm{T}$ or $\bar{v}(\varphi_2) = \mathrm{T}$. Assume, w.l.o.g., $\bar{v}(\varphi_1) = \mathrm{T}$. From the induction hypothesis we get a derivation $\hat{p}_1, \dots, \hat{p}_n \vdash \varphi_1$ so that we conclude

$$\dfrac{\begin{matrix}\vdots \\ \varphi_1\end{matrix}}{\varphi_1 \vee \varphi_2}\ \vee\mathrm{I}1$$

If $\bar{v}(\varphi) = \mathrm{F}$, then $\bar{v}(\varphi_1) = \mathrm{F}$ and $\bar{v}(\varphi_2) = \mathrm{F}$. By the induction hypothesis we get derivations $\hat{p}_1, \dots, \hat{p}_n \vdash \neg\varphi_1$ and $\hat{p}_1, \dots, \hat{p}_n \vdash \neg\varphi_2$. We conclude

$$\dfrac{\begin{matrix}\text{Lemma 15(2)} \\ \vdots \\ \neg\varphi_1 \wedge \neg\varphi_2 \to \neg(\varphi_1 \vee \varphi_2)\end{matrix} \qquad \dfrac{\begin{matrix}\vdots \\ \neg\varphi_1\end{matrix} \quad \begin{matrix}\vdots \\ \neg\varphi_2\end{matrix}}{\neg\varphi_1 \wedge \neg\varphi_2}\ \wedge\mathrm{I}}{\neg(\varphi_1 \vee \varphi_2)}\ \to\mathrm{E}$$

$\varphi = \varphi_1 \to \varphi_2$: If $\bar{v}(\varphi) = \mathrm{T}$ then $\bar{v}(\varphi_1) = \mathrm{F}$ or $\bar{v}(\varphi_2) = \mathrm{T}$. From the induction hypothesis we get either a derivation $\hat{p}_1, \dots, \hat{p}_n \vdash \neg\varphi_1$ or $\hat{p}_1, \dots, \hat{p}_n \vdash \varphi_2$. In either case get are able to derive $\neg\varphi_1 \vee \varphi_2$ using $\vee$I1 or $\vee$I2:

$$\dfrac{\begin{matrix}\vdots \\ \neg\varphi_1\end{matrix}}{\neg\varphi_1 \vee \varphi_2}\ \vee\mathrm{I}1 \qquad\qquad \dfrac{\begin{matrix}\vdots \\ \varphi_2\end{matrix}}{\neg\varphi_1 \vee \varphi_2}\ \vee\mathrm{I}2$$

We conclude

$$\dfrac{\begin{array}{c}\text{Lemma 15(5)}\\ \vdots \\ \neg\varphi_1 \vee \varphi_2 \to (\varphi_1 \to \varphi_2)\end{array} \qquad \begin{array}{c}\text{see above}\\ \vdots \\ \neg\varphi_1 \vee \varphi_2\end{array}}{\varphi_1 \to \varphi_2} \to\mathrm{E}$$

If $\bar{v}(\varphi) = \mathrm{F}$, then $\bar{v}(\varphi_1) = \mathrm{T}$ and $\bar{v}(\varphi_2) = \mathrm{F}$. By the induction hypothesis we get derivations $\hat{p}_1, \ldots, \hat{p}_n \vdash \varphi_1$ and $\hat{p}_1, \ldots, \hat{p}_n \vdash \neg\varphi_2$. We conclude

$$\dfrac{\begin{array}{c}\text{Lemma 15(6)}\\ \vdots \\ \varphi_1 \wedge \neg\varphi_2 \to \neg(\varphi_1 \to \varphi_2)\end{array} \qquad \dfrac{\begin{array}{c}\vdots\\ \varphi_1\end{array} \quad \begin{array}{c}\vdots\\ \neg\varphi_2\end{array}}{\varphi_1 \wedge \neg\varphi_2} \wedge\mathrm{I}}{\neg(\varphi_1 \to \varphi_2)} \to\mathrm{E}$$

This completes the proof. $\square$

The following lemma shows how to combine the basic proofs derived so far.

Lemma 17 *Let $\varphi_1, \ldots, \varphi_n, \varphi, \psi$ be propositional formulas. If $\varphi_1, \ldots, \varphi_n, \varphi \vdash \psi$ and $\varphi_1, \ldots, \varphi_n, \neg\varphi \vdash \psi$ then $\varphi_1, \ldots, \varphi_n \vdash \psi$*

Proof. Consider the following derivation:

$$\dfrac{\begin{array}{c}\text{Lemma 15(4)}\\ \vdots \\ \varphi \vee \neg\varphi\end{array} \qquad \begin{array}{c}[\varphi]^1\\ \vdots \\ \psi\end{array} \qquad \begin{array}{c}[\neg\varphi]^1\\ \vdots \\ \psi\end{array}}{\psi} \vee\mathrm{E}$$

This completes the proof. $\square$

Finally, we are able to proof the completeness theorem.

Theorem 18 (Completeness) *Let $\varphi_1, \ldots, \varphi_n$ and ψ be propositional formulas. If $\varphi_1, \ldots, \varphi_n \models \psi$ holds, then $\varphi_1, \ldots, \varphi_n \vdash \psi$ is valid.*

Proof. Define

$$\varphi := \varphi_1 \to (\varphi_2 \to (\varphi_3 \to \ldots (\varphi_n \to \psi) \ldots))$$

and let $\{p_1, \ldots, p_m\}$ be the set of propositional variables in φ. By Lemma 14(1) we have $\models \varphi$. Now, let v_1, v_2 be two truth assignments with $v_1(p_m) =$

T, $v_2(p_m) = $ F and $v_1(p_i) = v_2(p_i)$ for all $i \in \{1, \ldots, m-1\}$. Using the definition from Lemma 16 we get derivations $\hat{p}_1, \ldots, \hat{p}_{m-1}, p_m \vdash \varphi$ and $\hat{p}_1, \ldots, \hat{p}_{m-1}, \neg p_m \vdash \varphi$. Lemma 17 shows $\hat{p}_1, \ldots, \hat{p}_{m-1} \vdash \varphi$. After m repetitions of the last argument we end up with $\vdash \varphi$. Finally, we conclude $\varphi_1, \ldots, \varphi_n \vdash \psi$ using Lemma 14(2). $\square$

Example 19 (Example 8 revisited again) *This time we want to follow the completeness proof to find a derivation $\neg p \to q, \neg q \vdash p$. The first step is to actually consider the formula $(\neg p \to q) \to (\neg q \to p)$ and its truth table:*

p	q	$(\neg p$	$\to q)$	$\to$	$(\neg q$	$\to p)$
T	T	F	T	T	F	T
T	F	F	T	T	T	T
F	T	T	T	T	F	T
F	F	T	F	T	T	F

For each line in the table we construct a derivation as shown in Lemma 16. The first row results in a derivation $p, q \vdash (\neg p \to q) \to (\neg q \to p)$ (see derivation (1)). Notice that the premises q is not used in that derivation so that the same derivation also serves for the second row, i.e., is a derivation $p, \neg q \vdash (\neg p \to q) \to (\neg q \to p)$. Following Lemma 17 we combine two copies of that derivation to a derivation of $p \vdash (\neg p \to q) \to (\neg q \to p)$ (see derivation (2)). This is a redundant step, of course, since (1) is already such a derivation. We just want to follow the algorithm outlined in the completeness proof literally without any modification (or optimization).

We construct derivations for $\neg p, q \vdash (\neg p \to q) \to (\neg q \to p)$ and $\neg p, \neg q \vdash (\neg p \to q) \to (\neg q \to p)$, similarly (see derivation (3) and (4)). Those derivations are combined in derivation (5).

In the next step we combine derivation (2) and (5) leading to a derivation of $\vdash (\neg p \to q) \to (\neg q \to p)$. The final step is to modify the derivation to get a derivation $\neg p \to q, \neg q \vdash p$ (see derivation (7)).

1.4 Normal forms of formulas

Two formulas are said to be equivalent if they have the same meaning. More formally we get the following definition.

$$(1)\qquad \dfrac{\dfrac{\dfrac{\dfrac{p}{\neg\neg q \vee p}\,\vee\mathrm{I}2 \qquad \begin{matrix}\text{Lemma } 15(3)\\ \vdots \\ \neg\neg q \vee p \to (\neg q \to p)\end{matrix}}{\neg q \to p}\,\to\mathrm{E}}{\neg(\neg p \to q) \vee (\neg q \to p)}\,\vee\mathrm{I}2 \qquad \begin{matrix}\text{Lemma } 15(3)\\ \vdots \\ \neg(\neg p \to q) \vee (\neg q \to p) \to ((\neg p \to q) \to (\neg q \to p))\end{matrix}}{(\neg p \to q) \to (\neg q \to p)}\,\to\mathrm{E}$$

$$(2)\qquad \dfrac{\begin{matrix}\text{Lemma } 15(4)\\ \vdots \\ q \vee \neg q\end{matrix} \qquad \begin{matrix}(1)\quad p \ldots [q]^1\\ \vdots \\ (\neg p \to q) \to (\neg q \to p)\end{matrix} \qquad \begin{matrix}(1)\quad p \ldots [\neg q]^1\\ \vdots \\ (\neg p \to q) \to (\neg q \to p)\end{matrix}}{(\neg p \to q) \to (\neg q \to p)}\,\vee\mathrm{E}^1$$

$$(3)\qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{q \qquad \begin{matrix}\text{Lemma } 15(1)\\ \vdots \\ q \to \neg\neg q\end{matrix}}{\neg\neg q}\,\to\mathrm{E}}{\neg\neg q \vee p}\,\vee\mathrm{I}1 \qquad \begin{matrix}\text{Lemma } 15(3)\\ \vdots \\ \neg\neg q \vee p \to (\neg q \to p)\end{matrix}}{\neg q \to p}\,\to\mathrm{E}}{\neg(\neg p \to q) \vee (\neg q \to p)}\,\vee\mathrm{I}2 \qquad \begin{matrix}\text{Lemma } 15(3)\\ \vdots \\ \neg(\neg p \to q) \vee (\neg q \to p) \to ((\neg p \to q) \to (\neg q \to p))\end{matrix}}{(\neg p \to q) \to (\neg q \to p)}\,\to\mathrm{E}$$

$$(4)\qquad \dfrac{\dfrac{\dfrac{\dfrac{\neg p \quad \neg q}{\neg p \wedge \neg q}\,\wedge\mathrm{I} \qquad \begin{matrix}\text{Lemma } 15(6)\\ \vdots \\ \neg p \wedge \neg q \to \neg(\neg p \to q)\end{matrix}}{\neg(\neg p \to q)}\,\to\mathrm{E}}{\neg(\neg p \to q) \vee (\neg q \to p)}\,\vee\mathrm{I}1 \qquad \begin{matrix}\text{Lemma } 15(3)\\ \vdots \\ \neg(\neg p \to q) \vee (\neg q \to p) \to ((\neg p \to q) \to (\neg q \to p))\end{matrix}}{(\neg p \to q) \to (\neg q \to p)}\,\to\mathrm{E}$$

$$(5)\qquad \dfrac{\begin{array}{c}\text{Lemma 15(4)}\\ \vdots\\ q\vee\neg q\end{array}\qquad \begin{array}{c}(3)\quad p\ldots[q]^1\\ \vdots\\ (\neg p\to q)\to(\neg q\to p)\end{array}\quad \begin{array}{c}(4)\quad p\ldots[\neg q]^1\\ \vdots\\ (\neg p\to q)\to(\neg q\to p)\end{array}}{(\neg p\to q)\to(\neg q\to p)}\ \vee\mathrm{E}^1$$

$$(6)\qquad \dfrac{\begin{array}{c}\text{Lemma 15(4)}\\ \vdots\\ p\vee\neg p\end{array}\qquad \begin{array}{c}(2)\quad [p]^1\\ \vdots\\ (\neg p\to q)\to(\neg q\to p)\end{array}\quad \begin{array}{c}(5)\quad [\neg p]^1\\ \vdots\\ (\neg p\to q)\to(\neg q\to p)\end{array}}{(\neg p\to q)\to(\neg q\to p)}\ \vee\mathrm{E}^1$$

$$(7)\qquad \dfrac{\dfrac{\begin{array}{c}(6)\\ \vdots\\ (\neg p\to q)\to(\neg q\to p)\end{array}\quad \neg p\to q}{\neg q\to p}\ \to\mathrm{E}\qquad \neg q}{p}\ \to\mathrm{E}$$

Definition 20 *Let* $\varphi, \psi \in \mathrm{Prop}$ *be propositional formulas.* φ *and* ψ *are equivalent iff* $\varphi \models \psi$ *and* $\psi \models \varphi$.

Due to the soundness and completeness of natural deduction φ and ψ are equivalent iff $\varphi \vdash \psi$ and $\psi \vdash \varphi$. Furthermore, using Lemma 14(2) and the abbreviation $\leftrightarrow$ this property is equivalent to $\vdash \varphi \leftrightarrow \psi$. Notice that Lemma 15 shows the equivalence of certain formulas.

In this section we want to transform a given formula into one, which allows an easy validity checks. We want to start with an example.

Example 21 *Consider the formula*

$$\neg((p \to q) \vee \bot) \vee \neg p.$$

It seems not to be obvious whether that formula is a tautology, satisfiable or neither one. The first step is to remove the symbols $\to$ *and* $\bot$ *from the formula. We have already shown that* $\varphi \to \psi$ *is equivalent to* $\neg\varphi \vee \psi$. *Later (see Lemma 22(1)) we will also provide a derivation for* $\bot \leftrightarrow \varphi \wedge \neg\varphi$ *for an arbitrary formula* φ. *By using those equivalences we get the formula*

$$\neg(\neg p \vee q \vee (\neg p \wedge p)) \vee \neg p.$$

Notice that we also implicitly used the associativity of $\vee$ *(cf. see Lemma 22(3)). In the next step we distribute the negation symbol* $\neg$ *over the operation* $\wedge$ *and* $\vee$ *as indicated in Lemma 15. During this step we will also remove any occurrences of* $\neg\neg$*. We end up with*

$$(p \wedge \neg q \wedge (p \vee \neg p)) \vee \neg p.$$

Notice that negation symbol is always attached to a propositional variable. A propositional variable or the negation of a variable is called a literal. Using this term, the formula above is built from literals and the operations $\wedge$ *and* $\vee$*. The final step in the normalization process is to separate* $\wedge$ *and* $\vee$ *into two layers. This can be done since we are going to show certain distributivity laws. Depending on which symbol is used in the top layer we get the conjunctive normal form (cnf)*

$$(p \vee \neg p) \wedge (\neg q \vee \neg p) \wedge (p \vee \neg p \vee \neg p)$$

and the disjunctive normal form (dnf)

$$(p \wedge \neg q \wedge p) \vee (p \wedge \neg q \wedge \neg p) \vee \neg p.$$

The cnf can be used to conclude that the given formula is not a tautology since its subformula $\neg q \vee \neg p$ *can be false. On the other hand, the dnf shows that the formula is satisfiable since either of the subformulas* $p \wedge \neg q \wedge p$ *and* $\neg p$ *can be satisfied.*

First, we have to provide the remaining derivations.

Lemma 22 *Let* $\varphi, \varphi_1, \varphi_2, \varphi_3 \in \mathrm{Prop}$ *be propositional formulas. Then we have:*

1. $\vdash \varphi_1 \wedge \varphi_2 \leftrightarrow \varphi_2 \wedge \varphi_1$.
2. $\vdash \varphi_1 \vee \varphi_2 \leftrightarrow \varphi_2 \vee \varphi_1$.
3. $\vdash \bot \leftrightarrow \varphi \wedge \neg\varphi$.
4. $\vdash \varphi_1 \wedge (\varphi_2 \wedge \varphi_3) \leftrightarrow (\varphi_1 \wedge \varphi_2) \wedge \varphi_3$.
5. $\vdash \varphi_1 \vee (\varphi_2 \vee \varphi_3) \leftrightarrow (\varphi_1 \vee \varphi_2) \vee \varphi_3$.
6. $\vdash \varphi_1 \wedge (\varphi_2 \vee \varphi_3) \leftrightarrow (\varphi_1 \wedge \varphi_2) \vee (\varphi_1 \wedge \varphi_3)$.

7. $\vdash \varphi_1 \vee (\varphi_2 \wedge \varphi_3) \leftrightarrow (\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)$.

Proof. Again we provide derivations for both implications.

1.

$$\dfrac{\dfrac{\dfrac{[\varphi_1 \wedge \varphi_2]^1}{\varphi_2}\,\wedge\mathrm{E2} \qquad \dfrac{[\varphi_1 \wedge \varphi_2]^1}{\varphi_1}\,\wedge\mathrm{E1}}{\varphi_2 \wedge \varphi_1}\,\wedge\mathrm{I}}{\varphi_1 \wedge \varphi_2 \to \varphi_2 \wedge \varphi_1}\,\to\mathrm{I}^1$$

The second derivation is similar.

2.

$$\dfrac{\dfrac{[\varphi_1 \vee \varphi_2]^2 \qquad \dfrac{[\varphi_1]^1}{\varphi_2 \vee \varphi_1}\,\vee\mathrm{I2} \qquad \dfrac{[\varphi_2]^1}{\varphi_2 \vee \varphi_1}\,\vee\mathrm{I1}}{\varphi_2 \vee \varphi_1}\,\vee\mathrm{E}^1}{\varphi_1 \vee \varphi_2 \to \varphi_2 \vee \varphi_1}\,\to\mathrm{I}^2$$

The second derivation is similar.

3.

$$\dfrac{\dfrac{[\bot]^1}{\varphi \wedge \neg\varphi}\,\mathrm{PBC}}{\bot \to \varphi \wedge \neg\varphi}\,\to\mathrm{I}^1 \qquad\qquad \dfrac{\dfrac{\dfrac{[\varphi \wedge \neg\varphi]^1}{\neg\varphi}\,\wedge\mathrm{E2} \qquad \dfrac{[\varphi \wedge \neg\varphi]^1}{\varphi}\,\wedge\mathrm{E1}}{\bot}\,\neg\mathrm{E}}{\varphi \wedge \neg\varphi \to \bot}\,\to\mathrm{I}^1$$

4.

$$\dfrac{\dfrac{\dfrac{\dfrac{[\varphi_1 \wedge (\varphi_2 \wedge \varphi_3)]^1}{\varphi_1}\,\wedge\mathrm{E1} \qquad \dfrac{\dfrac{[\varphi_1 \wedge (\varphi_2 \wedge \varphi_3)]^1}{\varphi_2 \wedge \varphi_3}\,\wedge\mathrm{E2}}{\varphi_2}\,\wedge\mathrm{E1}}{\varphi_1 \wedge \varphi_2}\,\wedge\mathrm{I} \qquad \dfrac{\dfrac{[\varphi_1 \wedge (\varphi_2 \wedge \varphi_3)]^1}{\varphi_2 \wedge \varphi_3}\,\wedge\mathrm{E2}}{\varphi_3}\,\wedge\mathrm{E2}}{(\varphi_1 \wedge \varphi_2) \wedge \varphi_3}\,\wedge\mathrm{I}}{\varphi_1 \wedge (\varphi_2 \wedge \varphi_3) \to (\varphi_1 \wedge \varphi_2) \wedge \varphi_3}\,\to\mathrm{I}^1$$

The second derivation is similar.

5.

$$\dfrac{\dfrac{[\varphi_1 \lor (\varphi_2 \lor \varphi_3)]^3 \quad \dfrac{\dfrac{[\varphi_1]^2}{\varphi_1 \lor \varphi_2}\,{\lor}\mathrm{I}1}{(\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\lor}\mathrm{I}1 \quad \dfrac{[\varphi_2 \lor \varphi_3]^2 \quad \dfrac{\dfrac{[\varphi_2]^1}{\varphi_1 \lor \varphi_2}\,{\lor}\mathrm{I}2}{(\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\lor}\mathrm{I}1 \quad \dfrac{[\varphi_3]^1}{(\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\lor}\mathrm{I}2}{(\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\lor}\mathrm{E}^1}{(\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\lor}\mathrm{E}^2}{\varphi_1 \lor (\varphi_2 \lor \varphi_3) \to (\varphi_1 \lor \varphi_2) \lor \varphi_3}\,{\to}\mathrm{I}^3$$

The second derivation is similar.

6.

$$\dfrac{\dfrac{\dfrac{[\varphi_1 \land (\varphi_2 \lor \varphi_3)]^2}{\varphi_2 \lor \varphi_3}\,{\land}\mathrm{E}2 \quad \dfrac{\dfrac{\dfrac{[\varphi_1 \land (\varphi_2 \lor \varphi_3)]^2}{\varphi_1}\,{\land}\mathrm{E}1 \quad [\varphi_2]^1}{\varphi_1 \land \varphi_2}\,{\land}\mathrm{I}}{(\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3)}\,{\lor}\mathrm{I}1 \quad \dfrac{\dfrac{\dfrac{[\varphi_1 \land (\varphi_2 \lor \varphi_3)]^2}{\varphi_1}\,{\land}\mathrm{E}1 \quad [\varphi_3]^1}{\varphi_1 \land \varphi_3}\,{\land}\mathrm{I}}{(\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3)}\,{\lor}\mathrm{I}2}{(\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3)}\,{\lor}\mathrm{E}^1}{\varphi_1 \land (\varphi_2 \lor \varphi_3) \to (\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3)}\,{\to}\mathrm{I}^2$$

$$\dfrac{\dfrac{[(\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3)]^2 \quad \dfrac{\dfrac{[\varphi_1 \land \varphi_2]^1}{\varphi_1}\,{\land}\mathrm{E}1 \quad \dfrac{\dfrac{[\varphi_1 \land \varphi_2]^1}{\varphi_2}\,{\land}\mathrm{E}2}{\varphi_2 \land \varphi_3}\,{\lor}\mathrm{I}1}{\varphi_1 \land (\varphi_2 \lor \varphi_3)}\,{\land}\mathrm{I} \quad \dfrac{\dfrac{[\varphi_1 \land \varphi_3]^1}{\varphi_1}\,{\land}\mathrm{E}1 \quad \dfrac{\dfrac{[\varphi_1 \land \varphi_3]^1}{\varphi_3}\,{\land}\mathrm{E}2}{\varphi_2 \land \varphi_3}\,{\lor}\mathrm{I}2}{\varphi_1 \land (\varphi_2 \lor \varphi_3)}\,{\land}\mathrm{I}}{\varphi_1 \land (\varphi_2 \lor \varphi_3)}\,{\lor}\mathrm{E}^1}{(\varphi_1 \land \varphi_2) \lor (\varphi_1 \land \varphi_3) \to \varphi_1 \land (\varphi_2 \lor \varphi_3)}\,{\to}\mathrm{I}^2$$

7.

$$\dfrac{\dfrac{[\varphi_1 \vee (\varphi_2 \wedge \varphi_3)]^2 \quad \dfrac{\dfrac{[\varphi_1]^1}{\varphi_1 \vee \varphi_2}\,\vee\mathrm{I}1 \quad \dfrac{[\varphi_1]^1}{\varphi_1 \vee \varphi_3}\,\vee\mathrm{I}1}{(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}\,\wedge\mathrm{I} \quad \dfrac{\dfrac{\dfrac{[\varphi_2 \wedge \varphi_3]^1}{\varphi_2}\,\wedge\mathrm{E}1}{\varphi_1 \vee \varphi_2}\,\vee\mathrm{I}2 \quad \dfrac{\dfrac{[\varphi_2 \wedge \varphi_3]^1}{\varphi_3}\,\wedge\mathrm{E}2}{\varphi_1 \vee \varphi_3}\,\vee\mathrm{I}2}{(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}\,\wedge\mathrm{I}}{(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}\,\vee\mathrm{E}^1}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3) \to (\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}\,\to\mathrm{I}^2$$

For the second derivation we first give a derivation $(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3), \varphi_3 \vdash \varphi_1 \vee (\varphi_2 \wedge \varphi_3)$.

$$\dfrac{\dfrac{(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}{\varphi_1 \vee \varphi_2}\,\wedge\mathrm{E}1 \quad \dfrac{[\varphi_1]^1}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\vee\mathrm{I}1 \quad \dfrac{\dfrac{[\varphi_2]^1 \quad \varphi_3}{\varphi_2 \wedge \varphi_3}\,\wedge\mathrm{I}}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\vee\mathrm{I}2}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\vee\mathrm{E}^1$$

Using the result above we get the following.

$$\dfrac{\dfrac{\dfrac{[(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)]^2}{\varphi_1 \vee \varphi_3}\,\wedge\mathrm{E}2 \quad \dfrac{[\varphi_1]^1}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\vee\mathrm{I}1 \quad \begin{array}{c}[(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)]^2 \quad [\varphi_3]^1 \\ \vdots \\ \varphi_1 \vee (\varphi_2 \wedge \varphi_3)\end{array}}{\varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\vee\mathrm{E}^1}{(\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3) \to \varphi_1 \vee (\varphi_2 \wedge \varphi_3)}\,\to\mathrm{I}^2$$

This completes the proof. □

The next step is to formally define the two notions of normal forms of formulas.

Definition 23 *Let $l_{11}, \ldots, l_{mn_m}$ be literals, i.e., each l_{ij} is either a propositional variable p or its negation $\neg p$.*

1. *A formula is in conjunctive normal form (cnf) iff it is a conjunction of disjunctions of literals, i.e., it is of the form*

$$(l_{11} \vee \cdots \vee l_{1n_1}) \wedge \cdots \wedge (l_{m1} \vee \cdots \vee l_{mn_m}).$$

2. *A formula is in disjunctive normal form (dnf) iff it is a disjunction of conjunctions of literals, i.e., it is of the form*

$$(l_{11} \wedge \cdots \wedge l_{1n_1}) \vee \cdots \vee (l_{m1} \wedge \cdots \wedge l_{mn_m}).$$

The following lemma guarantees the existence of both normal forms. Notice that those normal forms are not unique.

Lemma 24 *Let $\varphi \in \mathrm{Prop}$ be a formula. Then there is a formula φ_c (φ_d) in conjunctive (disjunctive) normal form equivalent to φ.*

Proof. The proof is left as an exercise. □

The following lemma shows our intuitive argument in the example at the beginning of this section.

Lemma 25

1. *A disjunction $l_1 \vee \cdots \vee l_n$ of literals is valid iff there is a propositional variable p so that $l_i = p$ and $l_j = \neg p$ for some $i, j \in \{1, \ldots, n\}$.*

2. *A conjunction $l_1 \wedge \cdots \wedge l_n$ of literals is satisfiable iff it not contains a pair $l_i = p$ and $l_j = \neg p$ for some propositional variable p and $i, j \in \{1, \ldots, n\}$.*

Proof.

1. Assume $l_i = p$ and $l_j = \neg p$. In this case the formula $l_1 \vee \cdots \vee l_n$ evaluates to T for all truth assignments. For the converse implication assume that there is no pair i, j so that $l_i = p$ and $l_j = \neg p$. Then choose a truth assignment satisfying

$$v(p) = \begin{cases} \mathrm{F} & \text{iff } l_i = p \text{ for some } i \in \{1, \ldots, n\} \\ \mathrm{T} & \text{iff } l_i = \neg p \text{ for some } i \in \{1, \ldots, n\} \end{cases}$$

   This is possible since both cases are exclusive. We get $\bar{v}(l_1 \vee \cdots \vee l_n) = \mathrm{F}$, and, hence, $l_1 \vee \cdots \vee l_n$ is not valid.

2. This follows immediately from 1. using Lemma 5. □

This leads immediately to the following corollary.

Corollary 26 *1. A formula $(l_{11} \vee \cdots \vee l_{1n_1}) \wedge \cdots \wedge (l_{m1} \vee \cdots \vee l_{mn_m})$ in cnf is valid iff each disjunction $l_{i1} \vee \cdots \vee l_{in_i}$ $(i \in \{1 \ldots m\})$ contains a pair l_{ix_i}, l_{iy_i} with $l_{ix_i} = p$ and $l_{iy_i} = \neg p$ for some $x_i, y_i \in \{1, \ldots, n_i\}$ and propositional variable p.*

2. A formula $(l_{11} \wedge \cdots \wedge l_{1n_1}) \vee \cdots \vee (l_{m1} \wedge \cdots \wedge l_{mn_m})$ in dnf is satisfiable iff each conjunction $l_{i1} \wedge \cdots \wedge l_{in_i}$ $(i \in \{1 \ldots m\})$ does not contain a pair l_{ix_i}, l_{iy_i} with $l_{ix_i} = p$ and $l_{iy_i} = \neg p$ for some $x_i, y_i \in \{1, \ldots, n_i\}$ and propositional variable p.

Chapter 2

First-Order Logic

The language of propositional logic is quite restricted. Cross references between individuals in a statement are usually out of the scope of the language. For example, the statement

> 'If a person has a sibling and that sibling has a child then the person is an aunt or an uncle.'

can not be expressed since it refers to individual properties of persons. The naive approach by using $S \wedge C \to A$ with the following interpretation

1. S: a person has a sibling.

2. C: a sibling has a child.

3. A: a person is an aunt or an uncle.

does not work since the person in the first and last proposition, as well as the sibling in the first and the second proposition, are not related in that formula. $S \wedge C \to A$ just reads as:

> 'If a person has a sibling and a sibling has a child then a person is an aunt or an uncle.'

First-order logic is built to handle statements as the one above. For example, that statement could be rewritten (a little bit more formally) as:

> 'If x is person and has a sibling y and y has a child then x is an aunt or an uncle.'

Notice that we just made the cross references in the sentences more obvious. Using a similar interpretation as introduced above we may use $S(x, y) \wedge C(y) \rightarrow A(x)$ as a representation. Notice that this also enables us to introduce quantifications, i.e., we can talk about all x and some y.

2.1 Syntax

In order to provide a suitable language we need more than just propositional variables. For first-order logic we require the following components.

1. X a set of variables.
2. F a set of function symbols. Each symbol has its arity.
3. P a set of predicate symbols. Each symbol has its arity.

0-ary functions symbols are called constant symbols. Such a symbol corresponds to a function requiring no parameter at all, i.e., the function can be identified with the element it returns. Similar the propositional variable of propositional logic can be identified with 0-ary predicate symbols so that first-order logic becomes a extension of propositional logic.

Definition 27 *The set* Term *of terms is recursively defined by the following.*

1. *Each variable $x \in X$ is a term, i.e., $X \subseteq \text{Term}$.*
2. *If $f \in F$ is an n-ary function symbol and $t_1, \ldots, t_n \in \text{Term}$ are terms, then $f(t_1, \ldots, t_n) \in \text{Term}$.*

Examples of terms are $f(x, y, z)$ or $f(f(x, x, x), f(y, y, y), z)$ assuming that f is a ternary function symbol and x, y, z are variables.

Definition 28 *The set* FOL *of first-order formulas (or formulas) is recursively defined by the following.*

1. *If $p \in P$ is an n-ary predicate symbol and $t_1, \ldots, t_n \in \text{Term}$ are terms, then $p(t_1, \ldots, t_n) \in \text{FOL}$. Formulas of this kind are called atomic formulas.*
2. *$\perp$ is a formula, i.e, $\perp \in \text{FOL}$.*

3. *If* $\varphi \in \mathrm{FOL}$ *then* $\neg\varphi \in \mathrm{FOL}$.
4. *If* $\varphi_1, \varphi_2 \in \mathrm{FOL}$ *then*
   (a) $\varphi_1 \wedge \varphi_2 \in \mathrm{FOL}$ *and*
   (b) $\varphi_1 \vee \varphi_2 \in \mathrm{FOL}$ *and*
   (c) $\varphi_1 \rightarrow \varphi_2 \in \mathrm{FOL}$.
5. *If* $\varphi \in \mathrm{FOL}$ *and* $x \in X$ *then*
   (a) $\forall x{:}\varphi \in \mathrm{FOL}$ *and*
   (b) $\exists x{:}\varphi \in \mathrm{FOL}$.

Notice that propositional formulas are first-order formulas, i.e, we have $\mathrm{Prop} \subseteq \mathrm{FOL}$. The precedence of $\forall$ and $\exists$ is the same as $\neg$.

In the case of binary function and predicate symbols we will also use infix notation. For example, instead of writing $+(x, y)$ and $\leq (x, y)$ we use $x + y$ and $x \leq y$.

Example 29 *In this example we want to express some properties of the natural numbers in first-order logic. For this purpose we assume that* 1 *is a constant symbol (*0*-ary function symbol),* s *a unary function symbol, and* $=$ *a binary predicate symbol. With the interpretation in mind that* 1 *denotes the number one,* s *is the successor function, and* $=$ *denotes equality we may state the following formulas:*

$$\forall x{:}\forall y{:}(s(x) = s(y) \rightarrow x = y)$$
$$\forall x{:}(\neg(x = 1) \rightarrow \exists y{:}x = s(y)).$$

Notice that we cannot express the principle of induction since it quantifies over all properties. Such a statement is covered by second-order logic.

We adapt the usual conventions for some negated atomic formulas. For example, instead of writing $\neg(x = y)$ and $\neg(x \leq y)$ we use $x \neq y$ and $x \not\leq y$. In addition, we will group quantifications, i.e., we write $\forall x, y, z{:}\varphi$ instead of $\forall x{:}\forall y{:}\forall z{:}\varphi$.

Definition 30 *An occurrence of a variable* $x \in X$ *in a formula is called bounded iff it is in a subformula of the form* $\forall x{:}\varphi$ *or* $\exists x{:}\varphi$*. An occurrence is called free iff it is not bounded.*

Consider the formula

$$\forall x{:}P(x) \wedge \exists y{:}Q(x,y).$$

The occurrence of x in $P(x)$ and the occurrence of y are bounded. The second occurrence of x in $Q(x,y)$ is free. Another example is

$$\forall y{:}(y|x \wedge y \neq 1 \rightarrow x = y).$$

Here all occurrences of y are bounded, and all occurrences of x are free. With the standard interpretation the formula above expresses that x is prime.

Since variables are place holders we need some means to replace them with a concrete object. For example, we may want to replace x in the last formula by the constant symbol 2 in order to state that 2 is prime. In general we want to replace a variable by a term. Unfortunately, we have to be careful because of some undesired side effects of that operation. If we replace x by y in the last example we get

$$\forall y{:}(y|y \wedge y \neq 1 \rightarrow y = y).$$

This formula does not stand for 'y is prime'. The problem is that the term we going to substitute contains a variable y, and that a free occurrence of x is under the scope of $\forall y{:}$ or $\exists y{:}$. The variable is free so that any variable contained in the term we are going to substitute should also be free.

Definition 31 *Let $x \in X$ be a variable, and $\varphi \in \mathrm{FOL}$ be a formula. A $t \in \mathrm{Term}$ is called free for x in φ iff no free occurrence of x is in a subformula $\forall y{:}\varphi'$ or $\exists y{:}\varphi'$ for a variable y occurring in t.*

Now we are ready to introduce the notion of substitution.

Definition 32 *Let $x \in X$ be a variable, $t, t' \in \mathrm{Term}$ be terms, and $\varphi \in \mathrm{FOL}$ be a formula.*

1. *By $t'[t/x]$ we denote the result of replacing all occurences of x in t' by t.*

2. *If t is free for x in φ, then we denote by $\varphi[t/x]$ the result of replacing any free occurrence of x in φ by t.*

If we write $\varphi[t/x]$ we always assume that t is free for x. Later we will see that this can always be achieved by renaming bounded variables.

2.2 Semantics

Since we are now able to talk about individuals or elements the simple universe of truth values is not rich enough to provide a suitable interpretation of the entities of the language.

Definition 33 *Let F be a set of function symbols, and P be a set of predicate symbols. A model $\mathcal{M}$ consists of the following data:*

1. *$|\mathcal{M}|$ a non-empty set, called the universe.*
2. *For each function symbol $f \in F$ with arity n a n-ary function $f^{\mathcal{M}} : |\mathcal{M}|^n \to |\mathcal{M}|$.*
3. *For each predicate symbol $p \in P$ with arity n a subset $p^{\mathcal{M}} \subseteq |\mathcal{M}|^n$.*

Notice that constant symbols are interpreted by elements. Furthermore, 0-ary predicate symbols are mapped to subsets of $|\mathcal{M}|^0$, which is a set containing just one element by definition. This set has exactly two subsets, the empty set and itself, modelling F and T, respectively.

Similar to a truth assignment we need values for the free variables in order to define the semantics of terms and formulas.

Definition 34 *Let $\mathcal{M}$ be a model. An environment $\sigma : X \to |\mathcal{M}|$ is a function from the set of variables X to the universe of the model. For an environment σ, a variable x, and a value $a \in |\mathcal{M}|$ denote by $\sigma[a/x]$ the environment defined by*

$$\sigma[a/x](y) = \begin{cases} a & \textit{iff } x = y, \\ \sigma(y) & \textit{iff } x \neq y. \end{cases}$$

We start with the interpretation of the term. Naturally, a term should denote an element so that the interpretation of a term is an element of the universe.

Definition 35 *Let $\mathcal{M}$ be a model, and σ be an environment. The extension $\bar{\sigma} : \mathrm{Term} \to |\mathcal{M}|$ of σ is defined by:*

1. *$\bar{\sigma}(x) = \sigma(x)$ for every $x \in X$.*
2. *$\bar{\sigma}(f(t_1, \ldots, t_n)) = f^{\mathcal{M}}(\bar{\sigma}(t_1), \ldots, \bar{\sigma}(t_n))$.*

The next step is to define the validity of formulas.

Definition 36 *Let $\mathcal{M}$ be a model, and σ be an environment. The satisfaction relation $\models_{\mathcal{M}} \varphi[\sigma]$ is recursively defined by:*

1. $\models_{\mathcal{M}} p(t_1, \ldots, t_n)[\sigma]$ *iff* $(\bar{\sigma}(t_1), \ldots, \bar{\sigma}(t_n)) \in p^{\mathcal{M}}$.
2. $\not\models_{\mathcal{M}} \bot[\sigma]$, *i.e., not* $\models_{\mathcal{M}} \bot[\sigma]$.
3. $\models_{\mathcal{M}} \neg\varphi[\sigma]$ *iff* $\not\models_{\mathcal{M}} \varphi[\sigma]$.
4. $\models_{\mathcal{M}} \varphi_1 \wedge \varphi_2[\sigma]$ *iff* $\models_{\mathcal{M}} \varphi_1[\sigma]$ *and* $\models_{\mathcal{M}} \varphi_2[\sigma]$.
5. $\models_{\mathcal{M}} \varphi_1 \vee \varphi_2[\sigma]$ *iff* $\models_{\mathcal{M}} \varphi_1[\sigma]$ *or* $\models_{\mathcal{M}} \varphi_2[\sigma]$.
6. $\models_{\mathcal{M}} \varphi_1 \to \varphi_2[\sigma]$ *iff* $\models_{\mathcal{M}} \varphi_2[\sigma]$ *whenever* $\models_{\mathcal{M}} \varphi_1[\sigma]$.
7. $\models_{\mathcal{M}} \forall x{:}\varphi[\sigma]$ *iff* $\models_{\mathcal{M}} \varphi[\sigma[a/x]]$ *for all* $a \in |\mathcal{M}|$.
8. $\models_{\mathcal{M}} \exists x{:}\varphi[\sigma]$ *iff* $\models_{\mathcal{M}} \varphi[\sigma[a/x]]$ *for some* $a \in |\mathcal{M}|$.

Compare the definition above to the Definition 3. Restricted to propositional logic, a model correspond to a truth assignment and vice versa.

Definition 37 *Let Σ be a set of formulas, and φ be a formula.*

1. *φ is called valid in the model $\mathcal{M}$ ($\models_{\mathcal{M}} \varphi$) iff $\models_{\mathcal{M}} \varphi[\sigma]$ for all environments σ.*
2. *φ is called valid ($\models \varphi$) iff $\models_{\mathcal{M}} \varphi$ for all models $\mathcal{M}$.*
3. *φ is called satisfiable iff there is a model $\mathcal{M}$ and an environment so that $\models_{\mathcal{M}} \varphi[\sigma]$.*
4. *φ follows from Σ in $\mathcal{M}$ ($\Sigma \models_{\mathcal{M}} \varphi$) iff for all environments σ, whenever $\models_{\mathcal{M}} \psi[\sigma]$ for all $\psi \in \Sigma$, then $\models_{\mathcal{M}} \varphi[\sigma]$.*
5. *φ follows from Σ ($\Sigma \models \varphi$) iff $\Sigma \models_{\mathcal{M}} \varphi$ for all models $\mathcal{M}$.*

If a formula φ or a set of formulas Σ is valid in a model $\mathcal{M}$ we will call $\mathcal{M}$ a model of φ or Σ, respectively.

Let us consider an example.

Example 38 *Consider the language and the formulas of Example 29 again. The first model is the set of natural number* $\mathbb{N}$ *and the obvious interpretation of the function, constant and predicate symbols, e.g.,* $s^{\mathbb{N}}(x) = x + 1$*. In this case both formulas are valid. The first formula* $\forall x{:}\forall y{:}(s(x) = s(y) \rightarrow x = y)$ *simply says that successor is injective, and the second formula* $\forall x{:}(\neg(x = 1) \rightarrow \exists y{:}x = s(y))$ *says that every element except* 1 *has a predecessor.*

Now, we want to consider several different models. In all cases we will interpret the symbol $=$ *by equality. First, consider the model* A *with universe* $\{1^A\}$*. The function symbol* s *is interpreted by the identity function. This model can be visualized by the following graph:*

$$\begin{tikzcd} 1^A \arrow[loop above, "s^A"] \end{tikzcd}$$

The identity function is injective and all elements have a predecessor so that both formulas are again satisfied.

For the second model consider again the natural numbers. This time we interpret s *by the function* $n \mapsto 2n$*. This function is injective but all odd numbers are not in the image of that function.*

Last but not least, consider the models B *and* C *given by the graphs*

$$\begin{tikzcd} 1^B \arrow[r, "s^B"] & b \arrow[loop above, "s^B"] & & 1^C \arrow[r, "s^C"] & c_1 \arrow[loop above, "s^C"] \\ & & & c_0 \arrow[ur, "s^C"'] & \end{tikzcd}$$

Here s^B *is not injective but every elements except* 1 *has a predecessor. In the model* C *both formulas are not true.*

As in propositional logic we have the obvious relationship between satisfiability and validity.

Lemma 39 *Let* $\varphi \in \mathrm{FOL}$ *be a formula. Then* φ *is satisfiable iff* $\neg\varphi$ *is not valid.*

Proof. $\Rightarrow$: Assume φ is satisfiable. Then there is a model $\mathcal{M}$ and an environment σ with $\models_{\mathcal{M}} \varphi[\sigma]$. We conclude $\not\models_{\mathcal{M}} \neg\varphi[\sigma]$, and, hence, $\neg\varphi$ is not valid.

$\Leftarrow$: If $\neg\varphi$ is not valid then there is a model $\mathcal{M}$ and an environment σ with $\not\models_{\mathcal{M}} \neg\varphi[\sigma]$. We conclude $\models_{\mathcal{M}} \varphi[\sigma]$, and, hence, φ is satisfiable. $\square$

Similar to propositional logic we are just interested in variables that occur free.

Lemma 40 *Let* $t \in \mathrm{Term}$ *be a term,* $\varphi \in \mathrm{FOL}$ *be a formula, and* $\mathcal{M}$ *be a model.*

1. *If the environments* σ_1 *and* σ_2 *coincide on all variables of* t*, then* $\bar{\sigma}_1(t) = \bar{\sigma}_2(t)$.
2. *If the environments* σ_1 *and* σ_2 *coincide on all free variables of* φ*, then* $\models_{\mathcal{M}} \varphi[\sigma_1]$ *iff* $\models_{\mathcal{M}} \varphi[\sigma_2]$.

Proof. Both proofs are done by induction.

1. If $t = x$ is a variable we get
$$\bar{\sigma}_1(t) = \sigma_1(x) = \sigma_2(x) = \bar{\sigma}_2(t).$$
If t is of the form $f(t_1, \ldots, t_n)$ with a n-ary function symbol f and terms $t_1, \ldots, t_n$ we conclude
$$\begin{aligned} \bar{\sigma}_1(t) &= f^{\mathcal{M}}(\bar{\sigma}_1(t_1), \ldots, \bar{\sigma}_1(t_n)) \\ &= f^{\mathcal{M}}(\bar{\sigma}_2(t_1), \ldots, \bar{\sigma}_2(t_n)) && \text{by the induction hypothesis} \\ &= \bar{\sigma}_2(t). \end{aligned}$$
2. If $\varphi = p(t_1, \ldots, t_n)$ is an atomic formula then every variable in each term is free in φ so that we conclude $\bar{\sigma}_1(t_i) = \bar{\sigma}_2(t_i)$ for $i \in \{1, \ldots, n\}$ using 1., which immediately implies the assertion.

   The case φ is one of the formulas $\perp$, $\neg\varphi'$, $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$ or $\varphi_1 \to \varphi_2$ are straight forward applications of the induction hypothesis.

   Assume $\varphi = Qx{:}\varphi'$ with $Q \in \{\forall, \exists\}$. The free variables of φ' are the free variables of φ and the variable x. Consequently, the environments $\sigma_1[a/x]$ and $\sigma_2[a/x]$ for an arbitrary $a \in |\mathcal{M}|$ coincide on all free variables in φ'. We conclude
$$\begin{aligned} \models_{\mathcal{M}} \varphi[\sigma_1] &\Leftrightarrow \models_{\mathcal{M}} \varphi'[\sigma_1[a/x]] && \text{for all/some } a \in |\mathcal{M}| \\ &\Leftrightarrow \models_{\mathcal{M}} \varphi'[\sigma_2[a/x]] && \text{for all/some } a \in |\mathcal{M}| \\ &\Leftrightarrow \models_{\mathcal{M}} \varphi[\sigma_2], \end{aligned}$$
   where the second equivalence is an application of the induction hypothesis. □

The next lemma relates the two notion of substitution and updating an environment. First we want to illustrate it by an example.

Example 41 *Consider again the formula* $\forall y{:}(y|x \wedge y \neq 1 \rightarrow x = y)$ *using in the standard model* $\mathbb{N}$ *of the natural numbers, i.e., this formula states that* x *is prime. Now, consider the term* $2+3$*. On the one hand we could substitute* $2+3$ *for* x *in the formula, giving* $\forall y{:}(y|(2+3) \wedge y \neq 1 \rightarrow 2+3 = y)$*, and check its validity for a given environment* σ*. The formula is true for any environment since it does not contain any free variable. On the other hand, we could first compute the value* $\bar{\sigma}(2+3) = \mathbf{5}$ *(in order to distinguish terms and natural numbers we use bold symbols for numbers). This time we compute the validity of the original formula with the environment* $\sigma[\mathbf{5}/x]$*. Once again this result in true.*

Lemma 42 *Let* $x \in X$ *be a variable,* $t, t' \in \mathrm{Term}$ *be terms,* $\varphi \in \mathrm{FOL}$ *be a formula, and* $\mathcal{M}$ *be a model.*

1. $\bar{\sigma}(t'[t/x]) = \overline{\sigma[\bar{\sigma}(t)/x]}(t')$.
2. $\models_{\mathcal{M}} \varphi[t/x][\sigma]$ *iff* $\models_{\mathcal{M}} \varphi[\sigma[\bar{\sigma}(t)/x]]$.

Proof. Both assertions are shown by induction.

1. If $t' = y$ we distinguish two cases. If $x = y$ we get

$$\bar{\sigma}(t'[t/x]) = \bar{\sigma}(t) = \overline{\sigma[\bar{\sigma}(t)/x]}(x) = \overline{\sigma[\bar{\sigma}(t)/x]}(t').$$

If $x \neq y$ the environments σ and $\overline{\sigma[\bar{\sigma}(t)/x]}$ coincide on all variables in t'. We use Lemma 40(1) and conclude

$$\bar{\sigma}(t'[t/x]) = \bar{\sigma}(y) = \overline{\sigma[\bar{\sigma}(t)/x]}(y) = \overline{\sigma[\bar{\sigma}(t)/x]}(t').$$

If $t' = f(t_1, \ldots, t_n)$ we immediately get

$$\begin{aligned}
&\bar{\sigma}(t'[t/x]) \\
&= f^{\mathcal{M}}(\bar{\sigma}(t_1[t/x]), \ldots, \bar{\sigma}(t_n[t/x])) && \text{substitution} \\
&= f^{\mathcal{M}}(\overline{\sigma[\bar{\sigma}(t)/x]}(t_1), \ldots, \overline{\sigma[\bar{\sigma}(t)/x]}(t_n)) && \text{induction hypothesis} \\
&= \overline{\sigma[\bar{\sigma}(t)/x]}(t')
\end{aligned}$$

2. If $\varphi = p(t_1, \ldots, t_n)$ we conclude

$$
\begin{array}{ll}
\models_{\mathcal{M}} \varphi[t/x][\sigma] & \\
\Leftrightarrow \models_{\mathcal{M}} p(t_1[t/x], \ldots, t_n[t/x])[\sigma] & \text{substitution} \\
\Leftrightarrow (\bar{\sigma}(t_1[t/x]), \ldots, \bar{\sigma}(t_n[t/x])) \in p^{\mathcal{M}} & \\
\Leftrightarrow (\overline{\sigma[\bar{\sigma}(t)/x]}(t_1), \ldots, \overline{\sigma[\bar{\sigma}(t)/x]}(t_n)) \in p^{\mathcal{M}} & 1. \\
\Leftrightarrow \models_{\mathcal{M}} p(t_1, \ldots, t_n)[\sigma[\bar{\sigma}(t)/x]] & \\
\Leftrightarrow \models_{\mathcal{M}} \varphi[\sigma[\bar{\sigma}(t)/x]]. &
\end{array}
$$

The case φ is one of the formulas $\bot$, $\neg\varphi'$, $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$ or $\varphi_1 \to \varphi_2$ are straight forward applications of the induction hypothesis.

Assume $\varphi = Qy{:}\varphi'$ with $Q \in \{\forall, \exists\}$. In the case $x = y$ the variable x does not occur free in φ so that $\varphi[t/x] = \varphi$. By Lemma 40(2) we get $\models_{\mathcal{M}} \varphi[\sigma]$ iff $\models_{\mathcal{M}} \varphi[\sigma[\bar{\sigma}(t)/x]$, and, hence, the assertion. Assume $x \neq y$. Then we conclude

$$
\begin{array}{lll}
\models_{\mathcal{M}} \varphi[t/x][\sigma] & \Leftrightarrow \models_{\mathcal{M}} Qy{:}\varphi'[t/x][\sigma] & \\
& \Leftrightarrow \models_{\mathcal{M}} \varphi'[t/x][\sigma[a/y]] & \text{for all/some } a \in |\mathcal{M}| \\
& \Leftrightarrow \models_{\mathcal{M}} \varphi'[\sigma[a/y][\overline{\sigma[a/y]}(t)/x]] & \text{for all/some } a \in |\mathcal{M}| \\
& \Leftrightarrow \models_{\mathcal{M}} \varphi'[\sigma[a/y][\bar{\sigma}(t)/x]] & \text{for all/some } a \in |\mathcal{M}| \\
& & \text{by Lemma 40(2) since } t \\
& & \text{is free for } x \text{ in } Qy{:}\varphi \\
& \Leftrightarrow \models_{\mathcal{M}} \varphi'[\sigma[\bar{\sigma}(t)/x][a/y]] & \text{for all/some } a \in |\mathcal{M}| \\
& & \text{since } x \neq y \\
& \Leftrightarrow \models_{\mathcal{M}} Qy{:}\varphi'[\sigma[\bar{\sigma}(t)/x]] & \\
& \Leftrightarrow \models_{\mathcal{M}} \varphi[\sigma[\bar{\sigma}(t)/x]], &
\end{array}
$$

where the third equivalence is an application of the induction hypothesis. $\square$

2.3 Natural Deduction

We extend natural deduction for propositional logic by new rules for the new constructions, i.e., universal and existential quantification. Again, we will have introduction and elimination rules in both cases. We want to discuss the rules in detail.

For all elimination: If we know that $\forall x{:}\varphi$ is true, then it is legal to conclude that φ for x being an arbitrary element. In other terms, we are allowed to conclude that $\varphi[t/x]$ is true for an arbitrary term t.

$$\frac{\forall x{:}\varphi}{\varphi[t/x]}\ \forall\mathrm{E}$$

For all introduction: In order to show that a formula $\forall x{:}\varphi$ is true one may simply show φ, i.e., we simply assume that x is an arbitrary element. For this being legal we must assure that the variable x is not already used elsewhere (as a free variable), i.e., that it is a 'fresh/new' variable. We get the rule

$$\frac{\varphi}{\forall x{:}\varphi}\ \forall\mathrm{I} \quad \text{if } x \text{ does not occur free in any premises of this subtree}$$

This rule has condition, which has to be satisfied before we are allowed to apply this rule. Notice that this condition refers to the premises of the subtree, i.e., just to those assumptions that are not yet discarded.

Exists introduction: This rule is very simple. If we were able to derive $\varphi[t/x]$ we have already got a witness t of the existential version of the formula. The rule simply is

$$\frac{\varphi[t/x]}{\exists x{:}\varphi}\ \exists\mathrm{I}$$

Exists elimination: To understand this rule it is helpful to consider the case of a finite model, i.e., a model $\{a_1, \ldots, a_n\}$. In this case an existential formula $\exists x{:}\varphi$ is true if it is true for one of the elements $a_1, \ldots, a_n$. Assume that the terms $t_1, \ldots, t_n$ denote the elements, i.e., $\bar{\sigma}(t_i) = a_i$, then $\exists x{:}\varphi$ is true if $\varphi[t_1/x] \vee \cdots \vee \varphi[t_n/x]$ is true. Consequently, we get a rule similar to $\vee$ elimination. The formula χ must be independent of x, and x be local to that subtree motivating the variable condition of this rule.

$$\frac{\exists x{:}\varphi \qquad \begin{matrix}[\varphi] \\ \vdots \\ \chi\end{matrix}}{\chi}\ \exists\mathrm{E} \quad \begin{matrix}\text{if } x \text{ does not occur free in } \chi \text{ and in any} \\ \text{premises of the right subtree accept } \varphi\end{matrix}$$

As in the case of $\forall$I the variable condition refers to the premises of the right subtree, i.e., just to those assumptions that are not yet discarded.

As a first example we want to verify that bounded variables can be renamed.

Lemma 43 *Let $\varphi \in \mathrm{FOL}$ be a formula not containing a free occurrence of the variable y. The we have*

1. $\vdash \forall x{:}\varphi \leftrightarrow \forall y{:}\varphi[y/x]$.
2. $\vdash \exists x{:}\varphi \leftrightarrow \exists y{:}\varphi[y/x]$.

Proof. In both cases it is sufficient to show '$\rightarrow$'. The other implication is similar.

1.

$$\dfrac{\dfrac{\dfrac{[\forall x{:}\varphi]^1}{\varphi[y/x]}\ \forall\mathrm{E}}{\forall y{:}\varphi[y/x]}\ \forall\mathrm{I}}{\forall x{:}\varphi \rightarrow \forall y{:}\varphi[y/x]}\ {\rightarrow}\mathrm{I}^1$$

Notice that the variable condition of $\forall$I is satisfied since φ, and, hence, $\forall x{:}\varphi$ does not contain a free occurrence of y.

2.

$$\dfrac{\dfrac{[\exists x{:}\varphi]^2 \qquad \dfrac{[\varphi]^1}{\exists y{:}\varphi[y/x]}\ \exists\mathrm{I}}{\exists y{:}\varphi[y/x]}\ \exists\mathrm{E}^1}{\exists x{:}\varphi \rightarrow \exists y{:}\varphi[y/x]}\ {\rightarrow}\mathrm{I}^2$$

Notice that the $\exists$I is of the required form since $\varphi[y/x][x/y]$ is just φ (recall y does not occur free in φ). Furthermore, the variable condition of $\exists$E is satisfied since x does not occur free in $\exists y{:}\varphi[y/x]$. It occurs free in φ, which does not violate the condition.

This completes the proof. $\square$

We will provide further derivation when we consider normal forms of formulas in first-order logic.

Theorem 44 (Soundness) *Let $\varphi_1, \ldots, \varphi_n$ and ψ be formulas. If $\varphi_1, \ldots, \varphi_n \vdash \psi$, then $\varphi_1, \ldots, \varphi_n \models \psi$ holds.*

Proof. The proof is done by induction on the derivation $\varphi_1, \ldots, \varphi_n \vdash \psi$. The base case and the case corresponding to the introduction and elimination rules of $\neg, \wedge, \vee$ and $\rightarrow$ are similar to those in propositional logic, and, therefore, omitted. The remaining cases are:

$\forall$**I** : In this case $\psi = \forall x{:}\psi'$, and we have a derivation $\varphi_1, \ldots, \varphi_n \vdash \psi'$. The variable condition implies that x does not occur free in any of $\varphi_1, \ldots, \varphi_n$. Assume $\mathcal{M}$ is a model and σ an environment with $\models_{\mathcal{M}} \varphi_i[\sigma]$ for $i \in \{1, \ldots, n\}$. By Lemma 40(2) we have $\models_{\mathcal{M}} \varphi_i[\sigma[a/x]]$ for all $a \in |\mathcal{M}|$ and $i \in \{1, \ldots, n\}$. By the induction hypothesis we conclude $\models_{\mathcal{M}} \psi'[\sigma[a/x]]$ for all $a \in |\mathcal{M}|$, and, hence, $\models_{\mathcal{M}} \psi[\sigma]$.

$\forall$**E** : In this case $\psi = \psi'[t/x]$, and we have a derivation $\varphi_1, \ldots, \varphi_n \vdash \forall x{:}\psi'$. Assume $\mathcal{M}$ is a model and σ an environment with $\models_{\mathcal{M}} \varphi_i[\sigma]$ for $i \in \{1, \ldots, n\}$. By the induction hypothesis we conclude $\models_{\mathcal{M}} \forall x{:}\psi'[\sigma]$, and, in particular, $\models_{\mathcal{M}} \psi'[\sigma[\bar{\sigma}(t)/x]]$. From Lemma 42(2) we get $\models_{\mathcal{M}} \psi'[t/x][\sigma]$.

$\exists$**I** : In this case $\psi = \exists x{:}\psi'$, and we have a derivation $\varphi_1, \ldots, \varphi_n \vdash \psi'[t/x]$. Assume $\mathcal{M}$ is a model and σ an environment with $\models_{\mathcal{M}} \varphi_i[\sigma]$ for $i \in \{1, \ldots, n\}$. By the induction hypothesis we conclude $\models_{\mathcal{M}} \psi'[t/x][\sigma]$. Lemma 42(2) implies $\models_{\mathcal{M}} \psi'[\sigma[\bar{\sigma}(t)/x]]$, and, hence, $\models_{\mathcal{M}} \psi[\sigma]$.

$\exists$**E** : In this case we have derivations $\varphi_1, \ldots, \varphi_n \vdash \exists x{:}\psi'$ and $\Sigma, \psi' \vdash \psi$ with $\Sigma \subseteq \{\varphi_1, \ldots, \varphi_n\}$. The variable condition implies that x does not occur free in ψ and in any formula of Σ. Assume $\mathcal{M}$ is a model and σ an environment with $\models_{\mathcal{M}} \varphi_i[\sigma]$ for $i \in \{1, \ldots, n\}$. By the induction hypothesis we get $\models_{\mathcal{M}} \exists x{:}\psi'[\sigma]$. We conclude that there is an $a \in |\mathcal{M}|$ with $\models_{\mathcal{M}} \psi'[\sigma[a/x]]$. Since x does not occur free in the formulas in Σ we get $\models_{\mathcal{M}} \varphi[\sigma[a/x]]$ for all $\varphi \in \Sigma$ by Lemma 42(2). From the induction hypothesis we derive $\models_{\mathcal{M}} \psi[\sigma[a/x]]$. Again by Lemma 42(2) we conclude $\models_{\mathcal{M}} \psi[\sigma]$ since x does not occur free in ψ. $\square$

First-order languages quite often include a predicate symbol for equality $=$. The main difference between the symbol $=$ and other predicate symbols is that $=$ usually has a predefined interpretation, i.e., $=^{\mathcal{M}} = \{(a, a) \mid a \in |\mathcal{M}|\}$ for all models. We refer to this variant as first-order logic with equality. An extended version of natural deduction provides an introduction and elimination rule for =:

$$\frac{}{t = t} = \mathrm{I} \qquad \frac{t_1 = t_2 \quad \varphi[t_1/x]}{\varphi[t_2/x]} = \mathrm{E}$$

Notice that =I does not have an assumption, i.e., this rule is actually an axiom.

Our next goal is to show completeness of the calculus. First, we want to get rid of premises in a proof.

Lemma 45 *Let $\varphi_1, \ldots, \varphi_n$ and ψ be formulas. Then we have:*

1. $\varphi_1, \ldots, \varphi_n \models \psi$ *iff* $\models \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$.
2. $\varphi_1, \ldots, \varphi_n \vdash \psi$ *iff* $\vdash \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$.

Proof. Similar to Lemma 14. □

The next step is to take care about the free variables in a formula.

Lemma 46 *Let $\varphi \in \mathrm{FOL}$ be a formula. Then*

1. $\models_{\mathcal{M}} \varphi$ *iff* $\models_{\mathcal{M}} \forall x{:}\varphi$.
2. $\models \varphi$ *iff* $\models \forall x{:}\varphi$.
3. $\vdash \varphi$ *iff* $\vdash \forall x{:}\varphi$.

Proof.

1. $\Rightarrow$: Let σ be an arbitrary environment. By the assumption we have $\models_{\mathcal{M}} \varphi[\sigma[a/x]]$ for all $a \in |\mathcal{M}|$, and, hence, $\models_{\mathcal{M}} \forall x{:}\varphi[\sigma]$.

   $\Leftarrow$: Let σ be an arbitrary environment. Then $\sigma = \sigma[\sigma(x)/x]$. By the assumption we have $\models_{\mathcal{M}} \forall x{:}\varphi[\sigma]$, and, hence, $\models_{\mathcal{M}} \varphi[\sigma[\sigma(x)/x]]$. We conclude $\models_{\mathcal{M}} \varphi[\sigma]$.

2. This follows immediately from (1).

3. $\Rightarrow$: To the derivation $\vdash \varphi$ we apply the rule $\forall$I to get a derivation $\vdash \forall x{:}\varphi$. The variable condition is satisfied since the derivation has no premises.

   $\Leftarrow$: This follows by applying the rule $\forall$E. □

A formula that does not contain any free occurrence of a variable is called closed. Since every formula just contains finitely many free variables we may close a formula by adding universal quantifiers. e.g., if φ is a formula and has free variables $x_1, \ldots, x_n$ then $\forall x_1, \ldots, x_n{:}\varphi$ is a closed formula. We will denote this formula by $\forall\varphi$ and call it the closure of φ.

Lemma 47 *The following statements are equivalent:*

1. *The calculus of natural deduction is complete, i.e.,* $\varphi_1, \ldots, \varphi_n \models \psi$ *implies* $\varphi_1, \ldots, \varphi_n \vdash \psi$ *for all formulas* $\varphi_1, \ldots, \varphi_n$ *and* ψ*.*

2. $\models \varphi$ *implies* $\vdash \varphi$ *for all closed formulas* φ*.*

Proof.

1. ⇒ 2.: This implication is trivial.

2. ⇒ 1.: Assume $\varphi_1, \ldots, \varphi_n \models \psi$ and let $\varphi = \varphi_1 \to (\varphi_2 \to (\varphi_3 \to (\ldots(\varphi_n \to \psi)\ldots)))$. By Lemma 45(1) we have $\models \varphi$. Now, we apply Lemma 46(2) as often as we have free variables to conclude $\models \forall\varphi$. (2) implies $\vdash \forall\varphi$, and, hence $\vdash \varphi$ using Lemma 46(3). Finally, Lemma 45(2) shows $\varphi_1, \ldots, \varphi_n \vdash \psi$. □

We are going to prove (2) instead of (1).

Definition 48 *A set of closed formulas T is called a theory. A theory is called consistent iff $T \nvdash \bot$. It is called inconsistent iff it is not consistent, i.e., if $T \vdash \bot$.*

In the next lemma we want to relate derivations with the consistency of a theory.

Lemma 49 *Let T be a theory, and φ be a closed formula. Then $T \vdash \varphi$ iff $T \cup \{\neg\varphi\}$ is inconsistent.*

Proof. ⇒: Using $\neg\varphi$ and $\neg$E we get a derivation $T \cup \{\neg\varphi\} \vdash \bot$, i.e., $T \cup \{\neg\varphi\}$ is inconsistent.

⇐: This time we have a derivation $T \cup \{\neg\varphi\} \vdash \bot$. Using the rule PBC we get a derivation $T \vdash \varphi$. □

In the next lemma we provide the version of the completeness theorem we going to prove. Notice that (1) actually implies completeness.

Lemma 50 *The following statements are equivalent:*

1. *$T \models \varphi$ implies $T \vdash \varphi$ for all closed formulas φ and theories T.*

2. *Every consistent theory has a model.*

Proof.

1. $\Rightarrow$ 2.**:** If T is consistent, i.e., $T \not\vdash \bot$, then we have $T \not\models \bot$ by (1). This implies that there is a model $\mathcal{M}$ so that $\models_{\mathcal{M}} \psi$ for all $\psi \in T$ and $\not\models_{\mathcal{M}} \bot$. Consequently, $\mathcal{M}$ is a model of T.

2. $\Rightarrow$ 1.**:** Assume $T \not\vdash \varphi$. Then we have to show that $T \not\models \varphi$. By Lemma 49 the theory $T \cup \{\neg\varphi\}$ is consistent. From (2) we conclude that $T \cup \{\neg\varphi\}$ has a model $\mathcal{M}$, i.e., $\models_{\mathcal{M}} \psi$ for all $\psi \in T$ and $\not\models_{\mathcal{M}} \varphi$, i.e., $T \not\models \varphi$. $\square$

In the following we are going to prove 50(2). Our proof will be for the variant first-order logic with equality. Regular first-order logic is just a special case, of course.

We are facing the problem that we have to construct a model for a given theory. The key idea is to basically use the syntactic material itself, i.e., the universe is formed by the variable-free (or closed) terms.

Lemma 51 *Let T be a theory. The relation $\sim$ on closed terms defined by $t_1 \sim t_2$ iff $T \vdash t_1 = t_2$ is an equivalence relation with $t_1 \sim t_1', \ldots, t_n \sim t_n'$ implies*

1. *$f(t_1, \ldots, t_n) \sim f(t_1', \ldots, t_n')$ for all n-ary function symbols, and*

2. *$T \vdash p(t_1, \ldots, t_n)$ iff $T \vdash p(t_1', \ldots, t_n')$ for all n-ary predicate symbols p.*

Proof. The axiom =I shows that $\sim$ is reflexive. Assume $t_1 \sim t_2$ and $t_2 \sim t_3$. Then there are derivations $T \vdash t_1 = t_2$ and $T \vdash t_2 = t_3$. We get

$$\dfrac{\begin{matrix}\vdots \\ t_2 = t_3\end{matrix} \quad \begin{matrix}\vdots \\ t_1 = t_2\end{matrix}}{t_1 = t_3} \; =\mathrm{E}$$

and, hence, $t_1 \sim t_3$, i.e., $\sim$ is transitive. Notice that the formula φ of the rule =E is $t_1 = x$ in the example above. Assume $t_1 \sim t_2$, i.e., there is a derivation $T \vdash t_1 = t_2$. We get

$$\dfrac{\begin{matrix}\vdots \\ t_1 = t_2\end{matrix} \quad \dfrac{}{t_1 = t_1} \; =\mathrm{I}}{t_2 = t_1} \; =\mathrm{E}$$

and, hence, $t_2 \sim t_1$, i.e., $\sim$ is symmetric (This time we have used φ is $x = t_1$.

Now, assume $t_1 \sim t_1', \ldots, t_n \sim t_n'$, i.e., there are derivations $T \vdash t_1 = t_1', \ldots, T \vdash t_n = t_n'$. We get

$$\dfrac{\begin{array}{c}\vdots\\ t_1 = t_1'\end{array} \qquad \dfrac{}{f(t_1,t_2,\ldots,t_n) = f(t_1,t_2,\ldots,t_n)}\,{=}\mathrm{I}}{f(t_1,t_2,\ldots,t_n) = f(t_1',t_2,\ldots,t_n)}\,{=}\mathrm{E}$$

$n-1$ additional applications of the rule =E shows $T \vdash f(t_1,\ldots,t_n) = f(t_1',\ldots,t_n')$. Property (2) is shown analogously. □

Due to the last lemma the following structure is well-defined.

Definition 52 *Let T be a theory. Then the Henkin-model $\mathcal{H}$ of T is defined by*

1. $|\mathcal{H}| := \{[t] \mid t \in \mathrm{Term}$ *and t is closed*$\}$, *where $[t]$ denotes the equivalence class of the term t with respect to $\sim$.*
2. $f^{\mathcal{H}}([t_1],\ldots,[t_n]) := [f(t_1,\ldots,t_n)]$.
3. $([t_1],\ldots,[t_n]) \in p^{\mathcal{H}}$ *iff* $T \vdash p(t_1,\ldots,t_n)$.

Notice that in the Henkin-model we have $\bar{\sigma}(t) = [t]$ for all closed terms t (independent of σ).

Unfortunately, the model above is not necessarily a model of the theory. It might not even be a model because it is possible that the language does not have any closed terms. But if the theory (and the language) is sufficiently strong enough the Henkin-model is indeed a model of the theory.

Definition 53 *A theory T is called*

1. *complete iff $T \vdash \varphi$ or $T \vdash \neg\varphi$ for all closed formulas φ.*
2. *a Henkin-theory iff for every closed formula $\exists x{:}\varphi$ there is a constant symbol c so that $T \vdash \exists x{:}\varphi \to \varphi[c/x]$.*

As promised earlier we have the following lemma.

Lemma 54 *If T is consistent and a complete Henkin-theory then $\mathcal{H}$ is a model of T.*

Proof. First of all, the universe is not empty since there is a constant c with $T \vdash \exists x{:}x = x \to c = c$ because T is a Henkin-theory.

In order to show the property $\varphi \in T$ implies $\models_{\mathcal{H}} \varphi$ we are going to prove a stronger property. We are going to show

$$(*) \qquad \models_{\mathcal{H}} \varphi[\sigma[[t_1]/x_1]\dots[[t_n]/x_n]] \iff T \vdash \varphi[t_1/x_1]\dots[t_n/x_n]$$

for all formulas φ with free variables $x_1, \dots, x_n$, closed terms $t_1, \dots, t_n$ and environments σ. In the proof we are going to use the abbreviations

$$\begin{array}{rl} \overrightarrow{t} & \text{for} \quad (t_1, \dots, t_n), \\ \overrightarrow{t/x} & \text{for} \quad [t_1/x_1]\dots[t_n/x_n] \end{array}$$

and, similarly, $\overrightarrow{[t]}$, $\overrightarrow{[t]/x}$ and $\overrightarrow{\bar{\sigma}(t)/x}$. With those conventions (*) reads

$$\models_{\mathcal{H}} \varphi[\sigma\overrightarrow{[[t]/x]}] \iff T \vdash \varphi\overrightarrow{[t/x]}.$$

This is shown by induction.

$\varphi = p(s_1, \dots, s_m)$**:** First of all, we have

$$\begin{aligned} \overline{\sigma\overrightarrow{[[t]/x]}}(s_i) &= \overline{\sigma\overrightarrow{[\bar{\sigma}(t)/x]}}(s_i) \\ &= \bar{\sigma}(s_i\overrightarrow{[t/x]}) && \text{Lemma 42(1)} \\ &= [s_i\overrightarrow{[t/x]}] \end{aligned}$$

for $i \in \{1, \dots, m\}$. We conclude

$$\begin{aligned} &\models_{\mathcal{H}} p(s_1, \dots, s_m)[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow (\overline{\sigma\overrightarrow{[[t]/x]}}(s_1)), \dots, \overline{\sigma\overrightarrow{[[t]/x]}}(s_i)) \in p^{\mathcal{H}} \\ &\Leftrightarrow ([s_1\overrightarrow{[t/x]}], \dots, [s_m\overrightarrow{[t/x]}]) \in p^{\mathcal{H}} && \text{above} \\ &\Leftrightarrow T \vdash p(s_1\overrightarrow{[t/x]}, \dots, s_m\overrightarrow{[t/x]}) \\ &\Leftrightarrow T \vdash p(s_1, \dots, s_m)\overrightarrow{[t/x]}. \end{aligned}$$

$\varphi = \bot$**:** In this case we have $\not\models_{\mathcal{H}} \bot$ and $T \not\vdash \bot$ since T is consistent.

$\varphi = \neg\varphi'$: We immediately conclude

$$\begin{aligned} \models_{\mathcal{H}} \neg\varphi'[\sigma\overrightarrow{[[t]/x]}] &\Leftrightarrow \not\models_{\mathcal{H}} \varphi'[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow T \nvdash \varphi'\overrightarrow{[t/x]} && \text{induction hypothesis} \\ &\Leftrightarrow T \vdash (\neg\varphi')\overrightarrow{[t/x]}. && T \text{ complete} \end{aligned}$$

$\varphi = \varphi_1 \wedge \varphi_2$: In this case we have

$$\begin{aligned} &\models_{\mathcal{H}} \varphi_1 \wedge \varphi_2[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow \models_{\mathcal{H}} \varphi_1[\sigma\overrightarrow{[[t]/x]}] \text{ and } \models_{\mathcal{H}} \varphi_2[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow T \vdash \varphi_1\overrightarrow{[t/x]} \text{ and } T \vdash \varphi_2\overrightarrow{[t/x]} && \text{induction hypothesis} \\ &\Leftrightarrow T \vdash (\varphi_1 \wedge \varphi_2)\overrightarrow{[t/x]}. \end{aligned}$$

$\varphi = \varphi_1 \vee \varphi_2$ and $\varphi = \varphi_1 \to \varphi_2$: Similar to the previous case.

$\varphi = \exists y{:}\varphi'$: First of all, we have

$$\begin{aligned} &\models_{\mathcal{H}} \exists y{:}\varphi'[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow \models_{\mathcal{H}} \varphi'[\sigma\overrightarrow{[[t]/x]}[[s]/y]] && \text{for some } [s] \in |\mathcal{H}| \\ &\Leftrightarrow T \vdash \varphi'\overrightarrow{[t/x]}[s/y]]. && \text{for some closed term } s \\ & && \text{by the induction hypothesis} \end{aligned}$$

It remains to show that the last property is equivalent to $T \vdash (\exists y{:}\varphi')\overrightarrow{[t/x]}$. The implication $\Rightarrow$ follows by using the rule $\exists$I. Conversely, assume $T \vdash (\exists y{:}\varphi')\overrightarrow{[t/x]}$. Since T is a Henkin-theory there is a constant c with $T \vdash (\exists y{:}\varphi')\overrightarrow{[t/x]} \to \varphi'\overrightarrow{[t/x]}[c/y]$. We conclude $T \vdash \varphi'\overrightarrow{[t/x]}[c/y]$.

$\varphi = \forall y{:}\varphi'$: Similar to the previous case we get

$$\begin{aligned} &\models_{\mathcal{H}} \forall y{:}\varphi'[\sigma\overrightarrow{[[t]/x]}] \\ &\Leftrightarrow \models_{\mathcal{H}} \varphi'[\sigma\overrightarrow{[[t]/x]}[[s]/y]] && \text{for all } [s] \in |\mathcal{H}| \\ &\Leftrightarrow T \vdash \varphi'\overrightarrow{[t/x]}[s/y]]. && \text{for all closed term } s \\ & && \text{by the induction hypothesis} \end{aligned}$$

and it remains to be shown that the last property is equivalent to $T \vdash (\forall y{:}\varphi')\overrightarrow{[t/x]}$. The implication $\Leftarrow$ follows by using the rule $\forall$E. Conversely, assume $T \vdash \varphi'\overrightarrow{[t/x]}[s/y]]$ for all closed terms s. Since T is a Henkin-theory we have $T \vdash (\exists y{:}\neg\varphi')\overrightarrow{[t/x]} \to (\neg\varphi')\overrightarrow{[t/x]}[c/y]$ for a constant symbol c. Let ψ denote $\varphi'\overrightarrow{[t/x]}$ and consider the derivation

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{\psi[c/y]} \qquad \dfrac{\dfrac{\vdots}{\exists y{:}\neg\psi \to \neg\psi[c/y]} \quad [\exists y{:}\neg\psi]^1}{\neg\psi[c/y]}\;\to\mathrm{E}}{\bot}\;\neg\mathrm{E}}{\neg\exists y{:}\neg\psi}\;\neg\mathrm{I}^1 \qquad \dfrac{[\neg\psi]^2}{\exists y{:}\neg\psi}\;\exists\mathrm{I}}{\dfrac{\bot}{\psi}\;\mathrm{PBC}^2}\;\neg\mathrm{E}}{\forall y{:}\psi}\;\forall\mathrm{I}$$

The variable condition at the application of $\forall$I is satisfied since all premises of the subtree are elements of T, i.e., closed formulas. This completes the proof. $\square$

It remains to show that every consistent theory can be extended to a consistent and complete Henkin-theory. First, we want to show that just adding new constant symbols does not have any effect on the consistency of a theory.

In order to distinguish different languages we denote by $L(T)$ the language of T, i.e., $L(T) = (F, P)$ with F the set of function symbols and P the set of predicate symbols. We say that φ is a formula in the language $L(T)$ iff φ just contains symbols from F and P.

Lemma 55 *Let T be a theory, φ be a formula in the language $L(T)$, and C be a set of constant symbol with $F \cap C = \emptyset$. Then we have for all $c_1, \ldots, c_n \in C$ and variables $x_1, \ldots, x_n$*

$$T \vdash \varphi \qquad \Longleftrightarrow \qquad T \vdash \varphi[c_1/x_1] \ldots [c_n/x_n].$$

Proof. '$\Rightarrow$': Assume we have $T \vdash \varphi$. By using Lemma 46(3) n-times we get a derivation $T \vdash \forall x_1, \ldots x_n{:}\varphi$. We conclude $T \vdash \varphi[c_1/x_1] \ldots [c_n/x_n]$ by applying the rule $\forall$E n-times.

'$\Leftarrow$': We are going to use a similar notion as in the proof of Lemma 54, and we prove the following more general property for all formulas $\psi_1, \ldots, \psi_m$ and φ: Let $c_1, \ldots, c_n$ be the new constant symbols that occur in a derivation $T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}\} \vdash \varphi\overrightarrow{[c/x]}$. Then there is a derivation $T \cup \{\psi_1, \ldots, \psi_m\} \vdash \varphi$. We are going to prove this property by induction on the structure of the give derivation.

If the derivation is just an assumption, the formula $\varphi\overrightarrow{[c/x]}$ is either in T or equal to $\psi_i\overrightarrow{[c/x]}$ for an $i \in \{1, \ldots, m\}$. In the first case the formula does not contain any of the constant c since $F \cap C = \emptyset$ so that $\varphi\overrightarrow{[c/x]}$ is actually equal to φ. In the later case we conclude that $\varphi = \psi_i$ since both formulas do not contain any of the new constant symbols. Consequently, $\varphi = \psi_i$ is the required derivation.

$\wedge$I: We have $\varphi = \varphi_1 \wedge \varphi_2$, and we have derivations

$$T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}\} \vdash \varphi_1\overrightarrow{[c/x]},$$
$$T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}\} \vdash \varphi_2\overrightarrow{[c/x]}.$$

By the induction hypothesis we get derivations $T \cup \{\psi_1, \ldots, \psi_m\} \vdash \varphi_1$ and $T \cup \{\psi_1, \ldots, \psi_m\} \vdash \varphi_2$, which we combine using $\wedge$I to a derivation $T \cup \{\psi_1, \ldots, \psi_m\} \vdash \varphi$.

$\wedge$E1, $\wedge$E2, $\vee$I1, $\vee$I2, $\rightarrow$E, $\neg$E, $\forall$I, $\forall$E, $\exists$I are similar to the previous case.

$\vee$E: In this case we have derivations

$$T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}\} \vdash \varphi_1 \vee \varphi_2,$$
$$T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}, \varphi_1\} \vdash \varphi\overrightarrow{[c/x]},$$
$$T \cup \{\psi_1\overrightarrow{[c/x]}, \ldots, \psi_m\overrightarrow{[c/x]}, \varphi_2\} \vdash \varphi\overrightarrow{[c/x]}$$

for some formulas φ_1 and φ_2 (in the extended language, i.e., they may contain elements from C). Therefore, $\varphi_1 = \varphi_1'\overrightarrow{[c/x]}$ and $\varphi_1 = \varphi_1'\overrightarrow{[c/x]}$ by the assumption on the new constant occurring in the derivation. By the induction hypothesis we get derivations

$$T \cup \{\psi_1, \ldots, \psi_m \vdash \varphi_1' \vee \varphi_2',$$
$$T \cup \{\psi_1, \ldots, \psi_m, \varphi_1'\} \vdash \varphi,$$
$$T \cup \{\psi_1, \ldots, \psi_m, \varphi_2'\} \vdash \varphi,$$

which we combine using $\vee$E.

$\rightarrow$**I,** $\neg$**I, PBC,** $\exists$**E** are similar to the previous case. $\square$

In particular, the previous lemma implies that T is consistent iff T is consistent with respect to language enriched by new constant symbols.

Definition 56 *Let T be a theory in the language $L(T)$. We define the following languages and theories recursively:*

1. *$L_0 := L(T)$ and $T_0 := T$.*
2. *Let $C_{n+1} := \{c_{\exists x:\varphi} \mid \exists x{:}\varphi$ a closed formula in the language $L_n\}$ be a set of new constant symbols, i.e., $C_{n+1} \cap F_n = \emptyset$. Then $L_{n+1} := L_n \cup C_{n+1}$ and $T_{n+1} := T_n \cup \{\exists x{:}\varphi \rightarrow \varphi[c_{\exists x:\varphi}/x] \mid c_{\exists x:\varphi} \in C_{n+1}\}$.*
3. *$L_H := \bigcup\limits_{i\geq 0} L_i$, and $T_H := \bigcup\limits_{i\geq 0} T_i$.*

Lemma 57 *If T is a consistent theory, then T_H is a consistent Henkin-theory.*

Proof. First, we show by induction that every T_n is consistent. For $n = 0$ this is trivial. Assume there is a derivation $T_{n+1} \vdash \bot$. Then there are m formulas $\psi_i = \exists x_i{:}\varphi_i \rightarrow \varphi_i[c_i/x_i]$ with $i \in \{1, \ldots, m\}$ so that the derivation above is actually a derivation $T_n \cup \{\psi_1, \ldots, \psi_m\} \vdash \bot$. We assume that the variables $x_1, \ldots, x_m$ are different, otherwise we rename certain variables. By Lemma 45(2) we get a derivation $T_n \vdash \psi_1 \rightarrow (\psi_2 \rightarrow \ldots(\psi_m \rightarrow \bot)\ldots)$. Notice that ψ_i is of the form $\psi_i'[c_i/x_i]$ with $\psi_i' = \exists x_i{:}\varphi_i \rightarrow \varphi_i$ a formula in the language L_n so that the previous statement can be written as $T_n \vdash \psi_1' \rightarrow (\psi_2' \rightarrow \ldots(\psi_m' \rightarrow \bot)\ldots)[c_1/x_1]\cdots[c_n/x_n]$. Lemma 55 implies that there is a derivation $T_n \vdash \psi_1' \rightarrow (\psi_2' \rightarrow \ldots(\psi_m' \rightarrow \bot)\ldots)$ in the language L_n, and, hence, $T_n \cup \{\psi_1', \ldots, \psi_m'\} \vdash \bot$ using Lemma 45(2) again. The following steps are repeated m times:

By Lemma 45(2) we get a derivation $T_n \cup \{\psi_1', \ldots, \psi_{m-1}'\} \vdash \psi_m' \rightarrow \bot$, and by Lemma 46(3) $T_n \cup \{\psi_1', \ldots, \psi_{m-1}'\} \vdash \forall x_m{:}(\psi_m' \rightarrow \bot)$. Consider the derivation of $\exists x_m{:}\psi_m'$, and the combination of that derivation and the derivation above given in figure below. This shows that there is a derivation $T_n \cup \{\psi_1', \ldots, \psi_{m-1}\} \vdash \bot$.

After m repetition we end up with a derivation $T_n \vdash \bot$, a contradiction to the induction hypothesis that T_n is consistent.

$$
\dfrac{
\begin{array}{c}\text{Lemma 15(4)}\\ \vdots \\ \exists x_m{:}\varphi_m \vee \neg\exists x_m{:}\varphi_m\end{array}
\qquad
\dfrac{[\exists x_m{:}\varphi_m]^3 \qquad \dfrac{\dfrac{[\varphi_m]^1}{\exists x_m{:}\varphi_m \to \varphi_m}\,{\to}\mathrm{I}}{\exists x_m{:}(\exists x_m{:}\varphi_m \to \varphi_m)}\,\exists\mathrm{I}}{\exists x_m{:}(\exists x_m{:}\varphi_m \to \varphi_m)}\,\exists\mathrm{E}^1
\qquad
\dfrac{\dfrac{\dfrac{\dfrac{[\neg\exists x_m{:}\varphi_m]^3 \quad [\exists x_m{:}\varphi_m]^2}{\bot}\,\neg\mathrm{E}}{\varphi_m}\,\mathrm{PBC}}{\exists x_m{:}\varphi_m \to \varphi_m}\,{\to}\mathrm{I}^2}{\exists x_m{:}(\exists x_m{:}\varphi_m \to \varphi_m)}\,\exists\mathrm{I}
}{\exists x_m{:}(\exists x_m{:}\varphi_m \to \varphi_m)}\,\vee\mathrm{E}^3
$$

$$
\dfrac{
\begin{array}{c}\text{above}\\ \vdots \\ \exists x_m{:}\psi'_m\end{array}
\qquad
\dfrac{\dfrac{\begin{array}{c}T_n \cup \{\psi'_1, \ldots, \psi_{m-1}\}\\ \vdots \\ \forall x_m{:}(\psi'_m \to \bot)\end{array}}{\psi'_m \to \bot}\,\forall\mathrm{E} \qquad [\psi'_m]^1}{\bot}\,{\to}\mathrm{E}
}{\bot}\,\exists\mathrm{E}^1
$$

Now, assume T_H is not consistent. Since every derivation just uses finitely many premises and every formula uses just finitely many symbols this derivation is a derivation $T_n \vdash \bot$ for some n, a contradiction.

It remains to show that T_H is a Henkin-theory. Assume that $\exists x{:}\varphi$ is a closed formula in L_H. Since the formula just contains finitely many symbols there is an n so that $\exists x{:}\varphi$ is a closed formula in the language L_n. The theory T_{n+1} contains the formula $\exists x{:}\varphi \rightarrow \varphi[c/x]$ for a constant symbol c so that $T_H \vdash \exists x{:}\varphi \rightarrow \varphi[c/x]$ follows immediately. $\square$

For the next step we assume that the closed formulas of the language are enumerated, and we denote by φ_n the n-th closed formula. This does not cause any problems since the set of formulas is enumerable.

Definition 58 *Let T be a theory, and define the following theories recursively:*

1. $T_0 := T$.
2. $T_{n+1} := \begin{cases} T_n \cup \{\varphi_n\} & \text{if } T_n \cup \{\varphi_n\} \text{ is consistent,} \\ T_n \cup \{\neg\varphi_n\} & \text{if } T_n \cup \{\varphi_n\} \text{ is inconsistent.} \end{cases}$
3. $T^c := \bigcup_{i \geq 0} T_i$.

Lemma 59 *If T is a consistent Henkin-theory, then T^c is a consistent and complete Henkin-theory.*

Proof. First, we want to show that each T_n is consistent. The case $n = 0$ is trivial. Assume there T_{n+1} is inconsistent. Then by the construction of T_{n+1} the theory T_n is inconsistent, a contradiction to the induction hypothesis.

Now, assume T^c is inconsistent. Then there is a derivation $T^c \vdash \bot$. Since every derivation uses just finitely many premises this derivation is actually a derivation $T_n \vdash \bot$ for some n, a contradiction.

T^c is Henkin-theory because $T = T_0$ is, and the language was not modified.

Finally, for every closed formula φ_n we have $\varphi \in T_n$ or $\neg\varphi_n \in T_n$ so that T^c is complete. $\square$

We are now ready to prove the main theorem of this chapter.

Theorem 60 *Every consistent theory has a model.*

Proof. Let T be a consistent theory. Then the theory T_H^c, precisely $(T_H)^c$, is a consistent and complete Henkin-theory. By Lemma 54 this theory has a model $\mathcal{H}$. Let $\mathcal{H}'$ denote the model derived from $\mathcal{H}$ by restricting $\mathcal{H}$ to the language of T, i.e., removing the interpretation of those symbols that are not in the language $L(T)$.

Let $\varphi \in T$. Then $\varphi \in T_H^c$, and, hence, we have $\models_{\mathcal{H}} \varphi$. Since φ is a formula in the language $L(T)$ we conclude $\models_{\mathcal{H}'} \varphi$. $\square$

An important consequence of the previous theorem is the following:

Theorem 61 (Compactness) *Let T be a theory. Then T has a model iff every finite subset of T has a model.*

Proof. $'\Rightarrow'$ is trivial. For the converse implication assume T does not have a model. By Theorem 60 we conclude that T is inconsistent, i.e., there is a derivation $T \vdash \bot$. This derivation just uses finitely many premises from T. If T' denotes that set, we have $T' \vdash \bot$, i.e., there is a finite subset of T that is inconsistent. $\square$

Theorem 62 (Compactness (2nd version)) *Let T be a theory, and φ be a closed formula. Then φ is true in all models of T iff there is a finite subset S of T so that φ holds in all models of S.*

Proof. Assume S is a finite subset of T with a model $\mathcal{M}$ in which φ is false. Since φ is closed $\neg\varphi$ is true in $|mathvcalM$ verifying that $S \cup \{\neg\varphi\}$ has a model. If the above holds for every finite subset of T then we conclude by Theorem 61 that $T \cup \{\neg\varphi\}$ has a model. This is a contradiction to the assumption that φ is true in all models of T. $\square$

We want to illustrate the importance of the (two versions of the) compactness theorem by two examples.

Example 63 *In this example we want to construct a model that is different from the natural numbers, meaning not isomorphic, but satisfies exactly the same formulas.*

Consider the natural number $\mathbb{N}$ and a suitable language therefor that contains at least the constant symbol 1, the function symbol $+$ and the binary predicate $<$ with their usual interpretation. Let

$$Th(\mathbb{N}) := \{\varphi \mid \varphi \text{ closed and } \models_{\mathbb{N}} \varphi\},$$

i.e., the set of all closed formulas that are true in the natural numbers (also called the theory of $\mathbb{N}$*). Notice that* $Th(\mathbb{N})$ *is a complete theory since every closed formula is either true or false in this model.*

In order to construct a different model of $Th(\mathbb{N})$ *let* $\underline{n}$ *denote the term* $1+1+\cdots+1$ *with* n *occurrences of* 1*. Consequently, the value of* $\underline{n}$ *is the natural number* n*. Furthermore, we extend the language by a new constant symbol* c*, and we consider the theory*

$$T := Th(\mathbb{N}) \cup \{c > \underline{n} \mid n \in \mathbb{N}\}.$$

We want to show that this theory has a model using the compactness theorem. Therefore, let $S \subseteq T$ *be a finite subset of* T*. Since* S *is finite* S *is contained in* $T_m := Th(\mathbb{N}) \cup \{c > \underline{n} \mid n \leq m\}$ *for an* $m \in \mathbb{N}$*. The natural number together with* $m+1$ *being the interpretation of* c *is obviously a model of* T_m*, and, hence, a model of* S*.*

By removing the interpretation of c *from a model of* T *we get a model of* $Th(\mathbb{N})$*. This model still contains an element that is bigger than every natural, i.e., is not isomorphic to* $\mathbb{N}$*.*

Example 64 *In this example we consider the language of groups, i.e., a constant symbol* 1*, a binary function symbol* $\cdot$ *and a unary function symbol* $^{-1}$ *written as an exponent. The theory* G *of groups consists of the following formulas:*

$$\begin{aligned}
&\forall x, y, z\colon x \cdot (y \cdot z) = (x \cdot y) \cdot z,\\
&\forall x\colon 1 \cdot x = x,\\
&\forall x\colon x^{-1} \cdot x = 1.
\end{aligned}$$

The models of this theory are called groups. Furthermore, a group is called torsionfree iff the following formulas are valid:

$$\begin{aligned}
\varphi_2 = \quad & \forall x{:}(x \neq 1 \rightarrow x \cdot x \neq 1),\\
\varphi_3 = \quad & \forall x{:}(x \neq 1 \rightarrow (x \cdot x) \cdot x \neq 1),\\
\vdots \quad & \\
\varphi_n = \quad & \forall x{:}(x \neq 1 \rightarrow \underbrace{(\ldots(x \cdot x) \cdot \ldots) \cdot x}_{n-times} \neq 1),\\
\vdots \quad &
\end{aligned}$$

We want to show that there is no finite set of formulas characterizing the torsionfree groups, i.e., there is no finite set of closed formulas T so that $\mathcal{G}$ is a model of $G \cup \{\varphi_2, \varphi_3, \ldots\}$ iff $\mathcal{G}$ is a model of $G \cup T$. Notice that it is sufficient to show that there is no formula that characterizes the torsionfree groups since a model satisfies all formulas of a finite set $T = \{\psi_1, \ldots, \psi_n\}$ iff it satisfies the formula $\psi = \psi_1 \wedge \ldots \wedge \psi_n$.

Assume there is a formula ψ with the desired property. Then ψ is true in all torsionfree groups, i.e., true in all models of $G \cup \{\varphi_2, \varphi_3, \ldots\}$. By Theorem 62 there is an $m \in \mathbb{N}$ so that ψ is true in all models of the theory $G \cup \{\varphi_2, \varphi_3, \ldots, \varphi_m\}$. Let $p > m$ be a prime number, and consider the group $\mathbb{Z}_p$. This group satisfies $\varphi_2, \varphi_3, \ldots, \varphi_m$ but is not torsionfree.

2.4 Normal forms of formulas

In this section we want to study normal forms of first-order formulas. The normal forms discussed here can be combined with the disjunctive or conjunctive normal form introduced for predicate logic. First, we provide some derivations needed later.

Lemma 65 *Let $\varphi, \varphi_1, \varphi_2, \varphi_3 \in$ FOL be formulas so that x does not occur free in φ_3. Then we have:*

1. $\vdash \forall x{:}\forall y{:}\varphi \leftrightarrow \forall y{:}\forall x{:}\varphi$.
2. $\vdash \forall x{:}\forall y{:}\varphi \leftrightarrow \forall y{:}\forall x{:}\varphi$.
3. $\vdash \neg\forall x{:}\varphi \leftrightarrow \exists x{:}\neg\varphi$.
4. $\vdash \neg\exists x{:}\varphi \leftrightarrow \forall x{:}\neg\varphi$.
5. $\vdash \forall x{:}(\varphi_1 \wedge \varphi_3) \leftrightarrow \forall x{:}\varphi_1 \wedge \varphi_3$.
6. $\vdash \forall x{:}(\varphi_1 \wedge \varphi_2) \leftrightarrow \forall x{:}\varphi_1 \wedge \forall x{:}\varphi_2$.
7. $\vdash \forall x{:}(\varphi_1 \vee \varphi_3) \leftrightarrow \forall x{:}\varphi_1 \vee \varphi_3$.
8. $\vdash \exists x{:}(\varphi_1 \vee \varphi_3) \leftrightarrow \exists x{:}\varphi_1 \vee \varphi_3$.
9. $\vdash \exists x{:}(\varphi_1 \vee \varphi_2) \leftrightarrow \exists x{:}\varphi_1 \vee \exists x{:}\varphi_2$.
10. $\vdash \exists x{:}(\varphi_1 \wedge \varphi_3) \leftrightarrow \exists x{:}\varphi_1 \wedge \varphi_3$.

Proof. In the following derivation we will always omit the application of $\rightarrow$I and $\leftrightarrow$I.

1. We just prove $\rightarrow$ since the converse implication follows analogously:

$$\dfrac{\dfrac{\dfrac{\dfrac{\forall x{:}\forall y{:}\varphi}{\forall y{:}\varphi}\,\forall\mathrm{E}}{\varphi}\,\forall\mathrm{E}}{\forall x{:}\varphi}\,\forall\mathrm{I}}{\forall y{:}\forall x{:}\varphi}\,\forall\mathrm{I}$$

   The variable condition in the two applications is satisfied since $\forall x{:}\forall y{:}\varphi$ does not contain x or y freely.

2. Again, we just prove $\rightarrow$:

$$\dfrac{\exists x{:}\exists y{:}\varphi \qquad \dfrac{[\exists y{:}\varphi]^2 \qquad \dfrac{\dfrac{[\varphi]^1}{\exists x{:}\varphi}\,\exists\mathrm{I}}{\exists y{:}\exists x{:}\varphi}\,\exists\mathrm{I}}{\exists y{:}\exists x{:}\varphi}\,\exists\mathrm{E}^1}{\exists y{:}\exists x{:}\varphi}\,\exists\mathrm{E}^2$$

   The variable condition in the two applications of $\exists$E is satisfied since $\exists y{:}\exists x{:}\varphi$ does not contain x or y freely.

3.

$$\dfrac{\dfrac{\neg\forall x{:}\varphi \qquad \dfrac{\dfrac{\dfrac{[\neg\exists x{:}\neg\varphi]^2 \qquad \dfrac{[\neg\varphi]^1}{\exists x{:}\neg\varphi}\,\exists\mathrm{I}}{\bot}\,\neg\mathrm{E}}{\varphi}\,\mathrm{PBC}^1}{\forall x{:}\varphi}\,\forall\mathrm{I}}{\bot}\,\neg\mathrm{E}}{\exists x{:}\neg\varphi}\,\mathrm{PBC}^2 \qquad \dfrac{\dfrac{\exists x{:}\neg\varphi \qquad \dfrac{[\neg\varphi]^1 \qquad \dfrac{[\forall x{:}\varphi]^2}{\varphi}\,\forall\mathrm{E}}{\bot}\,\neg\mathrm{E}}{\bot}\,\exists\mathrm{E}^1}{\neg\forall x{:}\varphi}\,\neg\mathrm{I}^2$$

   The variable conditions of $\forall$I in the first derivation and of $\exists$E in the second derivation are satisfied since x does neither occur free in $\neg\exists x{:}\neg\varphi$ nor in $\bot$ and $\forall x{:}\varphi$.

4.

$$\dfrac{\dfrac{\dfrac{\neg\exists x{:}\varphi \qquad \dfrac{[\varphi]^1}{\exists x{:}\varphi}\,\exists\mathrm{I}}{\bot}\,\neg\mathrm{E}}{\neg\varphi}\,\neg\mathrm{I}^1}{\forall x{:}\neg\varphi}\,\forall\mathrm{I} \qquad \dfrac{\dfrac{[\exists x{:}\varphi]^2 \qquad \dfrac{\dfrac{\forall x{:}\neg\varphi}{\neg\varphi}\,\forall\mathrm{E} \qquad [\varphi]^1}{\bot}\,\neg\mathrm{E}}{\bot}\,\exists\mathrm{E}^1}{\neg\exists x{:}\varphi}\,\neg\mathrm{I}^2$$

The variable conditions of $\forall$I in the first derivation (provided by Shahid Mahmood) and of $\exists$E in the second derivation are satisfied since x does neither occur free in $\neg\exists x{:}\varphi$ nor in $\perp$ and $\forall x{:}\neg\varphi$.

5.
$$\dfrac{\dfrac{\dfrac{\dfrac{\forall x{:}(\varphi_1 \wedge \varphi_3)}{\varphi_1 \wedge \varphi_3}\,\forall\mathrm{E}}{\varphi_1}\,\wedge\mathrm{E1}}{\forall x{:}\varphi_1}\,\forall\mathrm{I} \qquad \dfrac{\dfrac{\forall x{:}(\varphi_1 \wedge \varphi_3)}{\varphi_1 \wedge \varphi_3}\,\forall\mathrm{E}}{\varphi_3}\,\wedge\mathrm{E2}}{\forall x{:}\varphi_1 \wedge \varphi_3}\,\wedge\mathrm{I} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\forall x{:}\varphi_1 \wedge \varphi_3}{\forall x{:}\varphi_1}\,\wedge\mathrm{E1}}{\varphi_1}\,\forall\mathrm{E} \qquad \dfrac{\forall x{:}\varphi_1 \wedge \varphi_3}{\varphi_3}\,\wedge\mathrm{E2}}{\varphi_1 \wedge \varphi_3}\,\wedge\mathrm{I}}{\forall x{:}(\varphi_1 \wedge \varphi_3)}\,\forall\mathrm{I}$$

The variable conditions of $\forall$I in the first derivation and the second derivation are satisfied since x does neither occur free in $\forall x{:}(\varphi_1 \wedge \varphi_3)$ nor in $\forall x{:}\varphi_1 \wedge \varphi_3$ (assumption on φ_3).

6. This derivation is similar to the previous one by adding an application of $\forall$I, respectively of $\forall$E, in the second branch of both parts.

7. For the implication $\rightarrow$ we first provide a derivation $\forall x{:}(\varphi_1 \vee \varphi_3), \neg\varphi_1 \vdash \forall x{:}\varphi_1 \vee \varphi_3$:

$$\dfrac{\dfrac{\forall x{:}(\varphi_1 \vee \varphi_3)}{\varphi_1 \vee \varphi_3}\,\forall\mathrm{E} \qquad \dfrac{\dfrac{\neg\varphi_1 \quad [\varphi_1]^1}{\perp}\,\neg\mathrm{E}}{\forall x{:}\varphi_1 \vee \varphi_3}\,\mathrm{PBC} \qquad \dfrac{[\varphi_3]^1}{\forall x{:}\varphi_1 \vee \varphi_3}\,\vee\mathrm{I2}}{\forall x{:}\varphi_1 \vee \varphi_3}\,\vee\mathrm{E}^1$$

Using the derivation above we get $\forall x{:}(\varphi_1 \vee \varphi_3), \neg\forall x{:}\varphi_1 \vdash \forall x{:}\varphi_1 \vee \varphi_3$ as follows:

$$\dfrac{\dfrac{\begin{array}{c}(3)\\ \vdots\\ \neg\forall x{:}\varphi_1 \rightarrow \exists x{:}\neg\varphi_1\end{array} \quad \neg\forall x{:}\varphi_1}{\exists x{:}\neg\varphi_1}\,\rightarrow\mathrm{E} \qquad \begin{array}{c}\forall x{:}(\varphi_1 \vee \varphi_3) \cdots [\neg\varphi_1]^1\\ \vdots\\ \forall x{:}\varphi_1 \vee \varphi_3\end{array}}{\forall x{:}\varphi_1 \vee \varphi_3}\,\exists\mathrm{E}^1$$

The variable condition for $\exists$E is satisfied since x does neither occur free in $\forall x{:}(\varphi_1 \vee \varphi_3)$ nor in $\forall x{:}\varphi_1 \vee \varphi_3$ (assumption on φ_3). Finally, using the derivation above we get

$$\dfrac{\begin{array}{c}\text{Lemma 15(4)}\\ \vdots\\ \forall x{:}\varphi_1 \vee \neg\forall x{:}\varphi_1\end{array} \qquad \dfrac{[\forall x{:}\varphi_1]^1}{\forall x{:}\varphi_1 \vee \varphi_3}\,\vee\mathrm{I1} \qquad \begin{array}{c}\forall x{:}(\varphi_1 \vee \varphi_3) \ldots [\neg\forall x{:}\varphi_1]^1\\ \vdots\\ \forall x{:}\varphi_1 \vee \varphi_3\end{array}}{\forall x{:}\varphi_1 \vee \varphi_3}\,\vee\mathrm{E}^1$$

The converse implication follows from

$$\dfrac{\dfrac{\forall x{:}\varphi_1 \lor \varphi_3 \qquad \dfrac{\dfrac{[\forall x{:}\varphi_1]^1}{\varphi_1}\,\forall\mathrm{E}}{\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}1 \qquad \dfrac{[\varphi_3]^1}{\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}2}{\varphi_1 \lor \varphi_3}\,\lor\mathrm{E}^1}{\forall x{:}(\varphi_1 \lor \varphi_3)}\,\forall\mathrm{I}$$

The variable condition of $\forall$I is satisfied since x does not occur free in $\forall x{:}\varphi_1 \lor \varphi_3$ (assumption on φ_3).

8.

$$\dfrac{\exists x{:}(\varphi_1 \lor \varphi_3) \qquad \dfrac{[\varphi_1 \lor \varphi_3]^2 \qquad \dfrac{\dfrac{[\varphi_1]^2}{\exists x{:}\varphi_1}\,\exists\mathrm{I}}{\exists x{:}\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}1 \qquad \dfrac{[\varphi_3]^2}{\exists x{:}\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}2}{\exists x{:}\varphi_1 \lor \varphi_3}\,\lor\mathrm{E}^1}{\exists x{:}\varphi_1 \lor \varphi_3}\,\exists\mathrm{E}^2$$

The variable condition of $\exists$E is satisfied since x does not occur free in $\exists x{:}\varphi_1 \lor \varphi_3$ (assumption on φ_3).

$$\dfrac{\exists x{:}\varphi_1 \lor \varphi_3 \qquad \dfrac{[\exists x{:}\varphi_1]^2 \qquad \dfrac{\dfrac{[\varphi_1]^1}{\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}1}{\exists x{:}(\varphi_1 \lor \varphi_3)}\,\exists\mathrm{I}}{\exists x{:}(\varphi_1 \lor \varphi_3)}\,\exists\mathrm{E}^1 \qquad \dfrac{\dfrac{[\varphi_3]^2}{\varphi_1 \lor \varphi_3}\,\lor\mathrm{I}2}{\exists x{:}(\varphi_1 \lor \varphi_3)}\,\exists\mathrm{I}}{\exists x{:}(\varphi_1 \lor \varphi_3)}\,\lor\mathrm{E}^2$$

The variable condition of $\exists$E is satisfied since x does not occur free in $\exists x{:}(\varphi_1 \lor \varphi_3)$.

9. This derivation is similar to the previous one by adding an application of $\exists$I, respectively of $\exists$E, in the right most branch of both parts.

10.

$$\dfrac{\exists x{:}(\varphi_1 \land \varphi_3) \qquad \dfrac{\dfrac{\dfrac{[\varphi_1 \land \varphi_3]^1}{\varphi_1}\,\land\mathrm{E}1}{\exists x{:}\varphi_1}\,\exists\mathrm{I} \qquad \dfrac{[\varphi_1 \land \varphi_3]^1}{\varphi_3}\,\land\mathrm{E}2}{\exists x{:}\varphi_1 \land \varphi_3}\,\land\mathrm{I}}{\exists x{:}\varphi_1 \land \varphi_3}\,\exists\mathrm{E}^1$$

The variable condition of $\exists$E is satisfied since x does not occur free in $\exists x{:}\varphi_1 \wedge \varphi_3$ (assumption on φ_3).

$$\dfrac{\dfrac{\exists x{:}\varphi_1 \wedge \varphi_3}{\exists x{:}\varphi_1}\wedge\mathrm{E1} \qquad \dfrac{\dfrac{[\varphi_1]^1 \quad \dfrac{\exists x{:}\varphi_1 \wedge \varphi_3}{\varphi_3}\wedge\mathrm{E2}}{\varphi_1 \wedge \varphi_3}\wedge\mathrm{I}}{\exists x{:}(\varphi_1 \wedge \varphi_3)}\exists\mathrm{I}}{\exists x{:}(\varphi_1 \wedge \varphi_3)}\exists\mathrm{E}^1$$

The variable condition of $\exists$E is satisfied since x does neither occur free in $\exists x{:}(\varphi_1 \wedge \varphi_3)$ nor in $\exists x{:}\varphi_1 \wedge \varphi_3$ (assumption on φ_3). $\square$

We want to illustrate the first normal form by an example.

Example 66 *Consider the formula $\forall x{:}p(x) \vee (\exists y{:}q(y) \wedge \forall x{:}r(x))$. Using the equivalences from the previous lemma we are able to move all quantifiers to the beginning of the formula.*

$$\begin{aligned}
\forall x{:}p(x) \vee (\exists y{:}q(y) \wedge \forall x{:}r(x)) &\leftrightarrow \forall x{:}(p(x) \vee (\exists y{:}q(y) \wedge \forall x{:}r(x))) \\
&\leftrightarrow \forall x{:}(p(x) \vee \exists y{:}(q(y) \wedge \forall x{:}r(x))) \\
&\leftrightarrow \forall x{:}\exists y{:}(p(x) \vee (q(y) \wedge \forall x{:}r(x))) \\
&\leftrightarrow \forall x{:}\exists y{:}(p(x) \vee \forall x{:}(q(y) \wedge r(x))) \\
&\leftrightarrow \forall x{:}\exists y{:}\forall z{:}(p(x) \vee (q(y) \wedge r(z)))
\end{aligned}$$

Notice that in Step 5 we had to rename to bounded variable x since it is free in $p(x)$. Furthermore, the matrix (the quantorfree part) of the last formula above is in disjunctive normal form.

Definition 67 *A formula $\varphi \in \mathrm{FOL}$ is called a prenex formula iff it is of the form $Q_1x_1{:}Q_2x_2{:}\cdots Q_nx_n{:}\varphi'$ where $Q_i \in \{\forall, \exists\}$ for $i \in \{1, \ldots, n\}$ and φ' quantorfree. φ' is called the matrix of φ.*

Theorem 68 *For every formula $\varphi \in \mathrm{FOL}$ there exists an equivalent prenex formula ψ.*

Proof. The proof uses induction on the structure of φ.

$\varphi = p(x_1, \ldots, x_n)$ or $\varphi = \perp$: In this case φ is already a prenex formulas.

$\varphi = \varphi' \wedge \varphi''$: By the induction hypothesis there are prenex formulas $\psi' = Q'_1 x'_1 {:} \cdots Q'_n x'_n {:} \chi'$ and $\psi'' = Q''_1 x''_1 {:} \cdots Q''_m x''_n {:} \chi''$. We may assume, w.l.o.g., that all variables $x'_1, \ldots, x'_n, x''_1, \ldots, x''_m$ are different (otherwise rename them using Lemma 43). Then φ is equivalent to

$$Q'_1 x'_1 {:} \cdots Q'_n x'_n {:} Q''_1 x''_1 {:} \cdots Q''_m x''_n {:} (\chi' \wedge \chi'')$$

by Lemma 65(5,7,8,10).

$\varphi = \varphi' \vee \varphi''$ or $\varphi = \varphi' \to \varphi''$: Similar to the previous case.

$\varphi = Qx{:}\varphi'$: By the induction hypothesis φ' is equivalent to a prenex formula ψ' so that φ is equivalent to the prenex formula $Qx{:}\psi'$. □

Example 66 also shows that the prenex normal form is not necessarily unique. Starting with the existential quantifier (instead of the left-most universal quantifier) leads to a different formula, namely $\exists y{:}\forall x{:}\forall z{:}(p(x) \vee (q(y) \wedge r(z)))$. However, both normal form are equivalent, of course.

As in the example the prenex normal form can be combined with either the conjunctive or disjunctive normal form of the matrix.

Definition 69 *A prenex formula $\varphi = Q_1 x_1 {:} \cdots Q_n x_n {:} \varphi'$ is called in universal (existential) skolem normal form iff $Q_i = \forall$ $(= \exists)$ for all $i \in \{1, \ldots, n\}$.*

The notion of a skolem normal form seems to be too restrictive, i.e., not every formula might have an equivalent skolem normal form. This is true if we fix the language. If we allow extra function symbols one is able to find always an equivalent skolem normal form in the following sense:

Theorem 70 *For every prenex formula φ in a language L there are formulas $\varphi_\forall$ and $\varphi_\exists$ in an extended language L^+ so that:*

1. $\vdash_{L^+} \varphi_\exists$ *iff* $\vdash_L \varphi$.
2. $\vdash_{L^+} \neg\varphi_\forall$ *iff* $\vdash_L \neg\varphi$.

Chapter 3

Modal Logic

We want to concentrate on propositional modal logics. As the previous sentence already suggests we are talking about a whole class of different logics distinguished by certain properties of their intended class of models. For example, temporal logic focuses on past and future time, whereas dynamic logic focuses on the behavior of programs. All of those logics have a common kernel. They enrich propositional logic by new operators, $\Box$ 'box' and $\Diamond$ 'diamond', that give access to a restricted version of quantification. Depending on the context (or the intended interpretation) those operators have different reading. We want to sketch three of them:

1. The formula $\Diamond\varphi$ can be read as 'It is possibly the case that φ', and $\Box\varphi$ as 'Necessarily φ'. Under this reading typical axiom schemas such as $\Box\varphi \to \Diamond\varphi$ and $\varphi \to \Diamond\varphi$ become 'Whatever is necessary, is possible' and 'What is, is possible'.

2. In this reading (epistemic logic) the operators express knowledge. $\Box\varphi$ reads as 'The agent knows that φ'.

3. The last version reads $\Box\varphi$ as 'φ is provable'. This reading is especially used in provability logics for Peano arithmetics.

3.1 Syntax

For simplicity we are going to introduce a basic modal language using one set of unitary modal operators, $\Box$ and $\Diamond$. An extension to multiple sets of operators, not necessarily unary, is possible. We will study such an example, propositional dynamic logic, in the next chapter.

Definition 71 *The set of* Mod *of modal formulas is defined as the set* Prop *of propositional formulas extended by the following construction: If* $\varphi \in \mathrm{Mod}$*, then* $\Box\varphi \in \mathrm{Mod}$ *and* $\Diamond\varphi \in \mathrm{Mod}$*.*

We adopt all convention and precedences from propositional logic.

3.2 Semantics

The intended interpretation of the modal operators is a quantification restricted to elements related by a certain relation. Notice that the elements are not explicitly available in the language, i.e., there are no constant symbol, no individual variables or even terms. The elements and the corresponding quantification is implicit in the langauge.

For example, in temporal logic we are talking about points in time that are in an obvious relationship. A suitable interpretation of $\Diamond\varphi$ is 'φ will be true at a future time', or in term of propositional logic $\exists t{:}t_C \leq t \wedge \varphi(t)$ where t_c is the current time and $\leq$ is the time relationship.

Definition 72 *A frame* $\mathcal{F} = (W, R)$ *is a pair such that:*

1. *W is a non-empty set, called the universe.*
2. *R is a binary relation on W, i.e., $R \subseteq W \times W$.*

We will use the usual notation Rxy to denote $(x, y) \in R$, i.e., that x and y are in relation R.

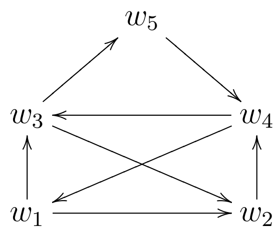
The elements of W are called points, states, worlds, times and situations depending on the intended interpretation of the modal logic.

A frame can be visualized as graph.

Example 73 *Consider the set* $W = \{w_1, w_2, w_3, w_4, w_5\}$ *and the relation*

$$R = \{(w_1, w_2), (w_1, w_3), (w_2, w_4), (w_3, w_2), (w_3, w_5), (w_4, w_1), (w_4, w_3), (w_5, w_4)\}$$

This frame can be visualized by the following graph where each edge corresponds to a pair in the relation R.



A model adds an interpretation of the propositional variables to a frame. A propositional variable has to be interpreted as a property (or predicate) on the elements of the universe.

Definition 74 *A model is a pair $\mathcal{M} = (\mathcal{F}, v)$ where $\mathcal{F}$ is a frame and $v :$ Prop $\rightarrow \mathcal{P}(W)$ is a function, called valuation, assigning a subset of W to each propositional variable.*

Now, we are ready to define the validity of modular formulas.

Definition 75 *Let $\mathcal{M}$ be a model, and $w \in W$ be a state. The satisfaction relation $\mathcal{M}, w \models \varphi$ is recursively defined by:*

1. $\mathcal{M}, w \models p$ *iff* $w \in v(p)$.

2. $\mathcal{M}, w \not\models \perp$.

3. $\mathcal{M}, w \models \neg\varphi$ *iff* $\mathcal{M}, w \not\models \varphi$.

4. $\mathcal{M}, w \models \varphi_1 \wedge \varphi_2$ *iff* $\mathcal{M}, w \models \varphi_1$ *and* $\mathcal{M}, w \models \varphi_2$.

5. $\mathcal{M}, w \models \varphi_1 \vee \varphi_2$ *iff* $\mathcal{M}, w \models \varphi_1$ *or* $\mathcal{M}, w \models \varphi_2$.

6. $\mathcal{M}, w \models \varphi_1 \rightarrow \varphi_2$ *iff* $\mathcal{M}, w \models \varphi_2$ *whenever* $\mathcal{M}, w \models \varphi_1$.

7. $\mathcal{M}, w \models \Diamond\varphi$ *iff there is a* $w' \in W$ *with* Rww' *and* $\mathcal{M}, w' \models \varphi$.

8. $\mathcal{M}, w \models \Box\varphi$ *iff for all* $w' \in W$ *with* Rww' *we have* $\mathcal{M}, w' \models \varphi$.

We say φ is true (or satisfied) in $\mathcal{M}$ at w iff $\mathcal{M}, w \models \varphi$. φ is called true in $\mathcal{M}$ (written $\mathcal{M} \models \varphi$) iff $\mathcal{M}, w \models \varphi$ for all $w \in W$, and φ is called valid (or true, or a tautology) iff it is true in all models (written $\models \varphi$).

We want to illustrate the previous definition by an example.

Example 76 *Consider the frame from Example 73, and choose the valuation function* $v(p) = \{w_2, w_3\}$ *and* $v(q) = \{w_3, w_5\}$.

Then the formula $\Diamond q \to \Box p$ *is true at* w_1 *since there is a state reachable from* w_1 *(via* R*) in* q *(namely* w_3*) and all states that is reachable from* w_1 *are in* p*. The same formula is not true at* w_3.

As a second example, consider the formula $p \to \Diamond\neg p$*. This formula is true at all states since from* w_2 *as well as from* w_3 *there is an edge ending outside of* p.

3.3 Deductive System

We want to switch to a different kind of system, so called Hilbert-style deductive system. This kind of system is characterized by providing a huge set of axioms, or better, axiom schemas, and few deduction rules. A characterizing property of those systems is that one does not make temporary assumptions. Every formula in the derivation is a valid formula (not pending on any assumption).

Definition 77 *A* K*-derivation is either an axiom or application of a rule to a* K*-derivations. The axioms are given by all instances of the following schemas:*

(Prop) *All instances of propositional tautologies.*

(K) $\Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)$.

(Dual) $\Diamond\varphi \leftrightarrow \neg\Box\neg\varphi$.

The rules are:

(MP) $\dfrac{\varphi \to \psi \quad \varphi}{\psi}\ MP$

(Gen) $\dfrac{\varphi}{\Box\varphi}\ Gen$

We write $\vdash \varphi$ *iff there is a derivation of* φ.

By an instance of a schema we refer to formulas obtained by uniformly replacing φ, ψ etc. by concrete modular formulas. For example, $\Box p \vee \neg\Box p$ is an axiom of the above calculus since it is an instance of $\varphi \vee \neg\varphi$, which is a propositional tautology.

As defined above (Prop) is an infinite, but decidable, list of schemas. It can be replaced by a finite list. For example, the following axiom schemas constitutes a suitable set:

1. $\varphi \to (\psi \to \varphi)$.
2. $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$.
3. $(\neg\psi \to \neg\varphi) \to (\varphi \to \psi)$.
4. $\bot \leftrightarrow \neg(\varphi \to \varphi)$.
5. $\varphi \vee \psi \leftrightarrow (\neg\varphi \to \psi)$.
6. $\varphi \wedge \psi \leftrightarrow \neg(\varphi \to \neg\psi)$.

It is easier to reason about a Hilbert-style system, e.g., proving soundness and completeness, than it is to reason about a natural deduction system. But they are hard to use as the following derivation of $\Box p \wedge \Box q \to \Box(p \wedge q)$ suggests:

Instead of using the tree notation, derivations in Hilbert-style calculus are often presented as a list of formulas where each line is either an axiom or the result of applying a rule to previous lines.

1	$p \to (q \to q \wedge p)$	(Prop)
2	$\Box(p \to (q \to q \wedge p))$	Gen 1
3	$\Box(p \to (q \to q \wedge p)) \to (\Box p \to \Box(q \to q \wedge p))$	K
4	$\Box p \to \Box(q \to q \wedge p)$	MP 2,3
5	$(\Box p \to \Box(q \to q \wedge p)) \to ((\Box(q \to q \wedge p)$ $\to (\Box q \to \Box(q \wedge p))) \to (\Box p \to (\Box q \to \Box(p \wedge q))))$	(Prop)
6	$\Box(q \to q \wedge p) \to (\Box q \to \Box(q \wedge p))$	K
7	$(\Box(q \to q \wedge p) \to (\Box q \to \Box(q \wedge p))) \to (\Box p \to (\Box q \to \Box(p \wedge q)))$	MP 4,5
8	$\Box p \to (\Box q \to \Box(p \wedge q))$	MP 6,7
9	$\Box p \to (\Box q \to \Box(p \wedge q)) \to (\Box p \wedge \Box q \to \Box(p \wedge q))$	(Prop)
10	$\Box p \wedge \Box q \to \Box(p \wedge q)$	MP 8,9

There is a common pitfall that is very easy to fall into when switching from natural deduction to a Hilbert-style system. We are not allowed to freely make assumptions. The following sequences is not legal:

1	p	Assumption
2	$\Box p$	Gen 1
3	$p \to \Box p$	Discharge assumption

Every line in Hilbert-style proof must be a valid formula. This is not the case for line 1. In fact the formula, $p \to \Box p$ is not even valid. Take, for example, the frame $a \longrightarrow b$ and let $v(p) = \{a\}$.

Theorem 78 (Soundness) *If* $\varphi \in \mathrm{Mod}$*, then* $\vdash \varphi$ *implies* $\models \varphi$*.*

Proof. The proof is done by induction on the derivation. We just prove the soundness of the Axioms K and Dual and that the rule GenEverything else was already shown in Chapter 2.

Assume $\mathcal{M}$ is a model and $w \in W$ with $\mathcal{M}, w \models \Box(\varphi \to \psi)$. We have to show that $\mathcal{M}, w \models \Box\varphi \to \Box\psi$ also holds. Assume that $\mathcal{M}, w \models \Box\varphi$. Then we have $\mathcal{M}, w' \models \varphi \to \psi$ and $\mathcal{M}, w' \models \varphi$ for all $w' \in R$ with Rww' using both assumptions. This implies $\mathcal{M}, w' \models \psi$ for all $w' \in R$ with Rww', and, hence, $\mathcal{M}, w \models \Box\psi$.

Axiom Dual is straight forward application of Lemma 65(3) and (4), and, therefore, left as exercise.

Finally, assume that $\models \varphi$, i.e., $\mathcal{M}, w \models \varphi$ for all models $\mathcal{M}$ and all $w \in W$. Now, let $\mathcal{M}$ be a model and $w \in W$ be an element. We have to show that $\mathcal{M}, w \models \Box\varphi$. Therefore, let $w' \in W$ be an element with Rww'. The assumption on φ implies $\mathcal{M}, w' \models \varphi$, and, hence, $\mathcal{M}, w \models \Box\varphi$. $\Box$

A completeness theorem can also be proved. Again, this is done by constructing a model for every consistent theory. We omit this proof and just state the result.

Theorem 79 (Completeness) *If* $\varphi \in \mathrm{Mod}$*, then* $\models \varphi$ *implies* $\vdash \varphi$*.*

Depending on extra properties of the relation R in a frame one may define stronger modal logics. Such a property is related to a certain extra axiom in the corresponding modal logic. The following list shows some commonly used axioms and their traditional names:

(4)	$\Diamond\Diamond\varphi \to \Diamond\varphi$
(T)	$\varphi \to \Diamond\varphi$
(B)	$\varphi \to \Box\Diamond\varphi$
(D)	$\Box\varphi \to \Diamond\varphi$

There is a convention naming the logics by the axioms used, e.g., KT4 is the modal logic generated by adding Axiom (T) and (4). Sometimes abbreviation or historical names are used, thus, one uses T, S4, B and S5 instead of KT, KT4, KB, KT4B.

The following lists several modal logic and the class of frames for which they are sound and complete:

K	all frames
K4	transitive frames
T (KT)	reflexive frames
B (KB)	symmetric frames
KD	right-unbounded frames
S4 (KT4)	reflexive, transitive frames
S5 (KT4B)	frames whose relation is an equivalence relation

3.4 Decidability

For simplicity we reduce the language of modal logic to the operators $\neg, \vee$ and $\Diamond$, i.e., we replace the other operators by their equivalent construction in the reduced language:

$$\begin{aligned} \varphi_1 \wedge \varphi_2 &= \neg(\neg\varphi_1 \vee \neg\varphi_2) \\ \varphi_1 \rightarrow \varphi_2 &= \neg\varphi_1 \vee \varphi_2 \\ \Box\varphi &= \neg\Diamond\neg\varphi \end{aligned}$$

Definition 80 *A set $\Sigma \subseteq \mathrm{Mod}$ of formulas is called subformula closed iff the following holds:*

1. *If $\triangle\varphi \in \Sigma$ for $\triangle \in \{\neg, \Diamond\}$, then $\varphi \in \Sigma$.*
2. *If $\varphi_1 \vee \varphi_2 \in \Sigma$, then $\varphi_1 \in \Sigma$ and $\varphi_2 \in \Sigma$.*

A subformula closed set of formulas naturally induces an equivalence relation on any model.

Definition 81 *Let $\mathcal{M}$ be a model, and Σ be a subformula closed set of formulas. The binary relation $\equiv_\Sigma \subset W \times W$ on the set of states is defined by:*

$$v \equiv_\Sigma w \ \text{ iff for all } \varphi \in \Sigma : (\mathcal{M}, v \models \varphi \iff \mathcal{M}, w \models \varphi).$$

Lemma 82 *Let $\mathcal{M}$ be a model, and Σ be a subformula closed set of formulas. Then $\equiv_\Sigma$ is an equivalence relation.*

Proof. All three properties, reflexivity, transitivity and symmetry, follow immediately from the corresponding property of $\iff$. $\square$

The induced equivalence relation $\equiv_\Sigma$ can be used to define a new model based on the equivalence classes.

Definition 83 *Let $\mathcal{M}$ be a model, and Σ be a subformula closed set of formulas. Then the model $\mathcal{M}^f_\Sigma$ is defined by:*

1. $W^f_\Sigma = \{[w]_\Sigma \mid w \in W\}$ *where $[w]_\Sigma$ denotes the equivalence class of w with respect to $\equiv_\Sigma$.*
2. $R^f[w]_\Sigma[v]_\Sigma$ *iff there is a $w' \in [w]_\Sigma$ and a $v' \in [v]_\Sigma$ with Rwv.*
3. $v^f_\Sigma(p) = \{[w]_\Sigma \mid w \in v(p)\}$ *for all propositional variables.*

$\mathcal{M}^f_\Sigma$ is called the filtration of $\mathcal{M}$ through Σ.

If it is clear from the context we will omit the index Σ.

Notice that the notion of a filtration is more general. For simplicity we have chosen the smallest filtration in the definition above. This choice is sufficient for the basic modal logic. If one considers additional properties such as transitivity the more general notion is required.

We want to illustrate the situation so far by an example.

Example 84 *Consider the model $\mathcal{M} = (\mathbb{N}, R, v)$ on the set of natural numbers with $R = \{(0,1),(0,2),(1,3),(2,3)\} \cup \{(n, n+1) \mid n \geq 3\}$ and $v(p) = \mathbb{N} \setminus \{0\}$. The frame is visualized in the following graph:*

$$\begin{tikzcd}
& 2 \arrow[dr] & & & \\
0 \arrow[ur] \arrow[dr] & & 3 \arrow[r] & 4 \arrow[r] & \cdots \\
& 1 \arrow[ur] & & &
\end{tikzcd}$$

Furthermore, consider the subformula closed set $\{\Diamond p, p\}$. In this model we have the following:

$$\begin{array}{l}
\mathcal{M}, n \models \Diamond p \text{ for all } n \in \mathbb{N}. \\
\mathcal{M}, 0 \not\models p. \\
\mathcal{M}, n \models p \text{ for all } n \in \mathbb{N} \setminus \{0\}
\end{array}$$

Consequently, we have exactly two equivalence classes $[0] = \{0\}$ *and* $[1] = \mathbb{N} \setminus \{0\}$*. Since we have* $R01$ *and* $R12$ *we get* $R^f[0][1]$ *and* $R^f[1][1]$*. Last but not least* $v^f(p) = \{[n] \mid n \in vp\} = \{[1]\}$*. The frame of the fitration is visualized by the following graph:*

$$\begin{tikzcd} {[0]} \arrow[r] & {[1]} \arrow[loop right] \end{tikzcd}$$

An important property of a filtration is that the state space is finite.

Theorem 85 *Let* $\mathcal{M}$ *be a model,* Σ *be a subformula closed set of formulas, and* $\mathcal{M}^f$ *be the filtration of* $\mathcal{M}$ *through* Σ*. Then* W^f *has at most* 2^n *elements where* n *is the size of* Σ*.*

Proof. The elements in W^f are the equivalence classes of $\equiv_\Sigma$. Define a function $g : W^f \to \mathcal{P}(\Sigma)$ by $g([w]) := \{\varphi \in \Sigma \mid \mathcal{M}, w \models \varphi\}$.

First, we have to show that g is well-defined. Assume $w \equiv v$. By the definition of $\equiv$ we have $\mathcal{M}, w \models \varphi$ iff $\mathcal{M}, v \models \varphi$ for all $\varphi \in \Sigma$. We conclude $\{\varphi \in \Sigma \mid \mathcal{M}, w \models \varphi\} = \{\varphi \in \Sigma \mid \mathcal{M}, v \models \varphi\}$.

We want to show that g is injective, which immediately implies the assertion. Assume $g([w]) = g([v])$. Then $\{\varphi \in \Sigma \mid \mathcal{M}, w \models \varphi\} = \{\varphi \in \Sigma \mid \mathcal{M}, v \models \varphi\}$ follows, and, hence, $w \equiv v$. $\Box$

It remains to be shown that the filtration satisfies essentially, i.e., in terms of the given set Σ of formulas, the same formulas.

Theorem 86 (Filtration Theorem) *Let* $\mathcal{M}$ *be a model,* Σ *be a subformula closed set of formulas, and* $\mathcal{M}^f$ *be the filtration of* $\mathcal{M}$ *through* Σ*. Then* $\mathcal{M}, w \models \varphi$ *iff* $\mathcal{M}^f, [w] \models \varphi$ *for all* $\varphi \in \Sigma$*.*

Proof. The proof is done by induction on φ. The property of Σ being subformula closed allows us to apply the induction hypothesis.

$\varphi = p$**:** The assertion follows immediately from the definition of v^f.

$\varphi = \bot$**:** This formula is not true in any model.

$\varphi = \neg\psi$**:** We compute

$$\begin{aligned} \mathcal{M}^f, [w] \models \neg\psi &\iff \mathcal{M}^f, [w] \not\models \psi && \text{definition } \models \\ &\iff \mathcal{M}, w \not\models \psi && \text{induction hypothesis} \\ &\iff \mathcal{M}, w \models \neg\psi && \text{definition } \models. \end{aligned}$$

$\varphi = \psi_1 \vee \psi_2$: Similar to the previous case.

$\varphi = \Diamond\psi$: Assume $\mathcal{M}, w \models \Diamond\psi$. Then there is a v with Rwv and $\mathcal{M}, v \models \psi$. The definition of R^f implies $R^f[w][v]$, and the induction hypothesis shows $\mathcal{M}^f, [v] \models \psi$. We conclude $\mathcal{M}^f[w] \models \Diamond\psi$.

Conversely, suppose $\mathcal{M}^f, [w] \models \Diamond\psi$. Then there is a $[v]$ with $R^f[w][v]$ and $\mathcal{M}^f, [v] \models \psi$. From the definition of R^f we conclude that there are elements $w' \in [w]$ and $v' \in [v]$ with $Rw'v'$. Furthermore, by the induction hypothesis we have $\mathcal{M}, v \models \psi$. Since $v' \equiv v$ we get $\mathcal{M}, v' \models \psi$ so that $\mathcal{M}, w' \models \Diamond\psi$ follows. We conclude $\mathcal{M}, w \models \Diamond\psi$ since $w \equiv w'$.$\Box$

Putting everything together we get the following theorem:

Theorem 87 (Finite Model Property) *Let $\varphi \in \mathrm{Mod}$ be a formula. Then φ is satisfiable iff φ is satisfiable on a finite model containing at most 2^n elements where n is the number of subformulas of φ.*

Proof. Assume φ is satisfiable, i.e., there is a model $\mathcal{M}$ and a state w with $\mathcal{M}, w \models \varphi$. Let Σ be the set of all φ and all of its subformulas. Then in the filtration we have $\mathcal{M}, [w] \models \varphi$, i.e., φ is satisfiable on a finite model with the required size. $\Box$

The last theorem shows that basic modal logic is decidable. As mentioned above, if we consider additional properties on a frame such as transitivity we may have to use a more general notion of a filtration. The reason is that the smallest filtration may not satisfy this additional property. We want to illustrate this by an example.

Example 88 *Consider the model $\mathcal{M} = (\{a, b, c, d\}, \{(a, b), (c, d)\}, v)$ with $v(p) = \{b, c\}$.*

$$\begin{tikzcd} b & c \arrow[d] \\ a \arrow[u] & d \end{tikzcd}$$

This frame is transitive. Now consider the subformula closed set $\{\Diamond p, p\}$. In this model we have:

$$\mathcal{M}, x \models p \textit{ for } x \in \{b, c\}.$$
$$\mathcal{M}, x \not\models p \textit{ for } x \in \{a, d\}.$$
$$\mathcal{M}, a \models \Diamond p.$$
$$\mathcal{M}, x \not\models \Diamond p \textit{ for } x \in \{b, c, d\}.$$

Consequently, we have three equivalence classes $[a] = \{a\}$*,* $[b] = \{b, c\}$ *and* $[d] = \{d\}$*. The relation* R^f *of the smallest filtration is visualized by:*

$$[a] \longrightarrow [b] \longrightarrow [d]$$

This frame is not transitive.

Chapter 4

Propositional Dynamic Logic

In this chapter we want to study a specific modal logic introduced to reason about programs - Propositional Dynamic Logic (PDL). This kind of logic describes the interaction between programs, their execution and propositions that are independent of the domain of computation, i.e., independently whether the program works on integers of lists of images. Therefore, programs in this logic do not contain a notion of an assignment. Instead, the program is built up from primitive statements that are interpreted by arbitrary binary relations on an abstract set of state. Using PDL one is able to express and to prove properties of the dynamic or flow of the program, i.e., the sequence of atomic operations. But the language of PDL is not rich enough to state properties of the result of that computation. The latter can be done in the first-order extension of PDL called Dynamic Logic (DL). However, reasoning about the sequence of computation is useful in many application. For example, code optimization in a compiler is usually based on general program transformations not effecting the sequence of atomic statements. Correctness of such a transformation can be expressed (and be proven) in PDL.

4.1 Syntax

PDL is a modal logic with countable many $\Box$ and $\Diamond$ operations. In fact, there is a $\Box$ and a $\Diamond$ for each program. Furthermore, there is a construction (?) converting a formula φ into a program $\varphi?$ - the test whether φ is true.

Definition 89 *Let P be a set of propositional variables, and $\prod$ be a set of* *atomic programs. The set* PDL *of formulas of PDL and the set* Prog *of programs are defined by:*

1. *Each propositional variable is a formula, i.e.,* $P \subseteq \mathrm{PDL}$.
2. $\bot \in \mathrm{PDL}$.
3. *If* $\varphi_1, \varphi_2 \in \mathrm{PDL}$, *then* $\neg\varphi_1, \varphi_1 \wedge \varphi_2, \varphi_1 \vee \varphi_2, \varphi_1 \to \varphi_2 \in \mathrm{PDL}$.
4. *If* $\varphi \in \mathrm{PDL}$ *and* $\pi \in \mathrm{Prog}$, *then* $[\pi]\varphi, \langle\pi\rangle\varphi \in \mathrm{PDL}$.
5. *Each atomic program is a program, i.e.,* $\prod \subseteq \mathrm{Prog}$.
6. *If* $\pi_1, \pi_2 \in \mathrm{Prog}$, *then* $\pi_1^*, \pi_1; pi_2, \pi_1 \cup \pi_2 \in \mathrm{Prog}$.
7. *If* $\varphi \in \mathrm{PDL}$, *then* $\varphi? \in \mathrm{Prog}$.

The programs have the following intuitive meanings:

$\pi_1; \pi_2$	Sequential composition, first π_1 then π_2.
$\pi_1 \cup \pi_2$	Nondeterministic choice, either π_1 or π_2.
π^*	Iteration, execute π a nondeterministically chosen finite number of times.
$\varphi?$	Test, proceed if φ is true, fail otherwise.

The program constructions may seem unconventional. They were chosen because of their mathematical simplicity. Familiar program constructions can be expressed in terms of those PDL programs. Some examples are:

skip	$\equiv (\neg\bot)?$
if φ **then** π_1 **else** π_2	$\equiv \varphi?; \pi_1 \cup (\neg\varphi)?; \pi_2$
while φ **do** π	$\equiv (\varphi?; \pi)^*; (\neg\varphi)?$
repeat π **until** φ	$\equiv \pi; ((\neg\varphi)?; \pi)^*; \varphi?$

The two modal operators have the following intuitive meanings:

$[\pi]\varphi$	"Every computation of π that terminates leads to a state satisfying φ."
$\langle\pi\rangle\varphi$	"There is a computation of π that terminates in a state satisfying φ."

Consequently, $\langle\pi\rangle\varphi$ implies that π terminates, whereas $[\pi]\varphi$ does not.

4.2 Semantics

PDL is a modal logic with an infinite collection of modal operators, one pair for each PDL program. Consequently, a frame has to provide a binary relation for each such pair, i.e., for each program. This relation is the computational behavior of the program and defined recursively starting from atomic programs.

Since programs and formulas are defined mutually recursive we are not longer able to separate frames and models.

Definition 90 *A PDL model* $\mathcal{M} = (W, \{R_\pi \mid \pi \in \mathrm{Prog}\}, v)$ *is a triple with a non-empty set* W*, a function* $v : P \to \mathbb{P}(W)$ *and a set of binary relations satisfying:*

1. $R_\alpha \subseteq W \times W$ *is an arbitrary binary relation on* W *for each atomic program* $\alpha \in \prod$.

2. $R_{\pi_1;\pi_2} = R_{\pi_1}; R_{\pi_2} = \{(u, w) \mid \exists v{:}R_{\pi_1}uv \wedge R_{\pi_2}vw\}$.

3. $R_{\pi_1 \cup \pi_2} = R_{\pi_1} \cup R_{\pi_2}$.

4. $R_{\pi^*} = R_\pi^* = \bigcup_{i \geq 0} R_\pi^i$ *(reflexive transitive closure).*

5. $R_{\varphi?} = \{(w, w) \mid \mathcal{M}, w \models \varphi\}$.

The validity relationship $\mathcal{M}, w \models \varphi$ *is defined as before.*

Example 91 *In the case of a PDL model we may visualize a frame by a labelled graph. For example, the model given by*

$$\begin{tikzcd} 1 \arrow[rr, "\alpha"] & & 2 \arrow[dl, "\beta"] & \\ & 3 \arrow[ul, bend right, "\beta"'] \arrow[ul, bend left, "\alpha"] \arrow[rr, "\alpha"] & & 4 \end{tikzcd}$$

and $v(p) = \{1, 2\}, v(q) = \{4\}$ *assigns the relation* $\{(1, 2), (3, 1), (3, 4)\}$ *to* α *and* $\{(2, 3), (3, 1)\}$ *to* β*. We want to compute the relation associated with the program*

$$(\textbf{while } p \textbf{ do } (\alpha \cup \beta)); \alpha,$$

i.e., the relation of $(p?;(\alpha \cup \beta))^*;(\neg p)?;\alpha$. *First, we compute the iterations of* $p?;(\alpha \cup \beta)$

$$\bigcup_{i=0}^{0}(p?;(\alpha \cup \beta))^i = \{(1,1),(2,2),(3,3),(4,4)\}$$
$$\bigcup_{i=0}^{1}(p?;(\alpha \cup \beta))^i = \{(1,1),(1,2),(2,2),(2,3),(3,3),(4,4)\}$$
$$\bigcup_{i=0}^{2}(p?;(\alpha \cup \beta))^i = \{(1,1),(1,2),(1,3),(2,2),(2,3),(3,3),(4,4)\}$$
$$\bigcup_{i=0}^{3}(p?;(\alpha \cup \beta))^i = \{(1,1),(1,2),(1,3),(2,2),(2,3),(3,3),(4,4)\}$$

In this case a finite iteration is sufficient. Consequently, the relation of $(p?;(\alpha \cup \beta))^*;(\neg p)?$ *is* $\{(1,3),(2,3),(3,3),(4,4)\}$ *and of the whole program* $\{(1,4),(2,4),(3,4)\}$. *By the definition of the modal operator we get*

$$\mathcal{M} \models [(\textbf{while } p \textbf{ do } (\alpha \cup \beta));\alpha]q$$
$$\mathcal{M} \not\models \langle(\textbf{while } p \textbf{ do } (\alpha \cup \beta));\alpha\rangle q$$