CHAPTER 3

PROPOSITIONAL LANGUAGES

1 Introduction

We define here a general notion of a propositional language. We show how to obtain, as specific cases, various languages for propositional classical logic and some non-classical logics.

We assume that any propositional language contains a countably infinite set of variables VAR, which elements will be denoted by $a, b, c, ...$ with indices if necessary.

All propositional languages share the general way their *sets of formulas* are formed.

What distinguishes one propositional language from the other is the choice of its **set of propositional connectives.**

We adopt a notation

$$\mathcal{L}_{CON},$$

where CON stands for the set of connectives for a propositional language with the set CON of logical connectives.

For example, the language

$$\mathcal{L}_{\{\neg\}}$$

denotes a propositional language with only one connective $\neg$. The language

$$\mathcal{L}_{\{\neg,\Rightarrow\}}$$

denotes that the language has only two connectives $\neg$ and $\Rightarrow$ adopted as propositional connectives.

Remember: any formal language deals with symbols only and is also called a symbolic language.

Theoretically one can use any symbols to denote propositional connectives. But there are some preferences, as those connectives do have intuitive meaning. The formal meaning, i.e. the assignment logical values to the formulas is going to be defined separately. It is formally called and **a semantics** for the given language

One language can have many semantics. Different logics can share the same language.

For example the language

$$\mathcal{L}_{\{\neg, \cap, \cup, \Rightarrow\}}$$

is used here as a propositional language of classical, intuitionistic logic, some many valued logics, and is extended to the language of modal logics.

It is possible for several languages to share the same semantics.
The classical propositional logic is the best example of such situation. It is due to functional dependency of classical logical connectives, discussed in the previous chapter. We will show formally later that the languages:

$$\mathcal{L}_{\{\neg\Rightarrow\}}, \mathcal{L}_{\{\neg\cap\}}, \mathcal{L}_{\{\neg\cup\}}, \mathcal{L}_{\{\neg, \cap, \cup, \Rightarrow\}}, \mathcal{L}_{\{\neg, \cap, \cup, \Rightarrow, \Leftrightarrow\}},$$

and even two languages with only one binary propositional connectives, denoted usually by $\uparrow$ and $\downarrow$, respectively, i.e languages

$$\mathcal{L}_{\{\uparrow\}}, \mathcal{L}_{\{\downarrow\}}$$

all share the same semantics characteristic for classical propositional logic.

The connectives have well established common names, even if their semantic can differ.

We use names, as in Chapter 2, *negation, conjunction, disjunction* and *implication* for $\neg, \cap, \cup, \Rightarrow$, respectively. The connective $\uparrow$ is called *alternative negation* and $A \uparrow B$ reads: *not both A and B*. The connective $\downarrow$ is called *joint negation* and $A \downarrow B$ reads: *neither A nor B*.

Other most common propositional connectives are probably modal connectives of *possibility* and *necessity* .

Standard modal symbols are $\Box$ for *necessity* and $\Diamond$ for *possibility.*

We will also use symbols **C** and **I** for modal connectives of possibility and necessity, respectively.

A formula $\mathbf{C}A$, or $\Diamond A$ reads: *it is possible that A* or *A is possible* and a formula $\mathbf{I}A$, or $\Box A$ reads: *it is necessary that A* or *A is necessary.*

The motivation for notation **C** and **I** arises from topological interpretation of modal S4 and S5 logics. **C** becomes equivalent to *a set closure operation*, hence $\mathbf{C}A$ means a closure of the set A, and **I** becomes equivalent to*a set interior operation* and $\mathbf{I}A$ denotes an interior of the set A.

The symbols $\Diamond$, **C** and $\Box$, **I** are not the only symbols used for modal connectives. Other notations include N for necessity and P for possibility. There are also a variety of modal logics created by computer scientists all with their own set of symbols and motivations for their use. The modal logic language extends the classical logic, i.e. we adopt the language

$$\mathcal{L}_{\{\neg, \mathbf{C}, \mathbf{I}, \cup\}}, \mathcal{L}_{\{\neg, \cap, \cup, \Rightarrow\}} \text{ or } \mathcal{L}_{\{\Box, \Diamond, \neg, \cap, \cup, \Rightarrow\}}.$$

The knowledge logics use together with classical connectives a new *knowledge connective* denoted by K. The formula KA reads: *it is known that* A or A *is known.* The language of a knowledge logic is denoted by

$$\mathcal{L}_{\{\neg,\ K,\ \cup\}}, \mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow\}}.$$

The autoepistemic logics use, for example *believe connectives*, often denoted by B. The formula BA reads: *it is believed that* A . They also extend the classical logic and hence their language is

$$\mathcal{L}_{\{\neg,\ B,\ \cup\}}, \mathcal{L}_{\{\neg,\cap,\cup,\Rightarrow\}}.$$

Temporal logics use, together with classical propositional connectives, the temporal connectives . For example one can use connectives (operators, as they are often called) F, P, G, and H to denote the following intuitive readings. FA reads A *is true at some future time*, PA reads A *was true at some past time*, GA reads A *will be true at all future times*, and HA reads A *has always been true in the past.* In order to take account of this variation of truth-values over time, some formal semantics were created, and many more will be created.

It is possible to create connectives with more then one or two arguments, but we allow here only one and two argument connectives, as logics which will be discussed here use only those two kind of connectives.

2 Formal Propositional Languages

We adopt the following definition, common to all propositional languages considered in our propositional logics investigations.

Definition 2.1 *By* **a propositional language** *we understand a pair*

$$\mathcal{L} = (\mathcal{A}, \mathcal{F}),$$

where $\mathcal{A}, \mathcal{F}$ *are called the* **alphabet** *and* **a set of formulas**.

The alphabet $\mathcal{A}$, and the set of formulas $\mathcal{F}$ are defined as follows.

Definition 2.2 **Alphabet** *is a set*

$$\mathcal{A} = VAR \cup CON \cup PAR,$$

where VAR, CON, PAR are all disjoint sets and VAR, CON are non-empty sets.

VAR is a countably infinite set, called **a set of propositional variables***;*

we denote elements of VAR by $a, b, c, \ldots$ *etc, (with indices if necessary);*

the set $CON \neq \emptyset$ *is a finite set of* **propositional connectives***;*

$PAR \neq \emptyset$ *is a set of* **auxiliary symbols** *and we assume that* PAR *contains two elements* $(,)$ *called* **parentheses***, i.e.*

$$PAR = \{(,)\}.$$

The set PAR may be empty, for example of the case of Polish notation, but we assume here that it contains two parenthesis to make the reading of formulas more natural and uniform.

We assume that CON a non empty set, what means that *there is* a logical connective. We specify the set CON for specific cases (specific logics).

We denote the language $\mathcal{L}$ (definition 2.1) with the set of connectives CON by

$$\mathcal{L}_{CON}.$$

It is possible to consider languages with connectives which have more then one or two arguments, nevertheless we restrict ourselves to languages with one or two argument connectives only.

Definition 2.3 We assume *that the set* CON *of propositional connectives of the language (definition 2.1)*

$$\mathcal{L}_{CON}$$

is non-empty and contains unary and binary connectives, i.e.

$$CON = C_1 \cup C_2$$

where

C_1 *is a finite set (possibly empty) of* **unary connectives** *,*

C_2 *is a finite set (possibly empty) of* **binary connectives** *of the language* $\mathcal{L}$*.*

The set $\mathcal{F}$ of all formulas of a propositional language $\mathcal{L}_{CON}$ is built recursively from the signs of the alphabet $\mathcal{A}$, i.e.

$$\mathcal{F} \subseteq \mathcal{A}^*,$$

where $\mathcal{A}^*$ is the set of all finite sequences (strings) form from elements of $\mathcal{A}$.

Definition 2.4 *The set $\mathcal{F}$ of* **all formulas** *of the language $\mathcal{L}_{CON}$ is* **the smallest set** *built from the signs of the alphabet $\mathcal{A}$, i.e. $\mathcal{F} \subseteq \mathcal{A}^*$, such that the following conditions hold:*

(1) $VAR \subseteq \mathcal{F}$ *(atomic formulas),*

(2) *if $A \in \mathcal{F}$, $\triangledown \in C_1$ i.e $\triangledown$ is an one argument connective, then $\triangledown A \in \mathcal{F}$,*

(3) *if $A, B \in \mathcal{F}$, $\circ \in C_2$ i.e $\circ$ is a two argument connective, then $(A \circ B) \in \mathcal{F}$.*

Definition 2.5 *The elements of the set $VAR \subseteq \mathcal{F}$ are called* **atomic formulas***.*

The set $\mathcal{F}$ is often called also a set of all **well formed formulas** (wff) of the language.

Example 1
Consider

$$CON = C_1 = \{\neg\}.$$

It means that the language $\mathcal{L}$ contains only one, one -argument connective $\neg$. If we call $\neg$, *a negation*, then we can say that the formulas of this languages are propositional variables or a multiple negation s of of a propositional variable. I. e. for any $a \in$ VAR, $a \in \mathcal{F}$, $\neg a \in \mathcal{F}$, $\neg\neg a \in \mathcal{F}$, $\neg\neg\neg a \in \mathcal{F}$, ... etc.

Observe that the strings $(\neg a), \neg, \neg(\neg a), \neg(a)$ are not well formed formulas of our propositional language. I.e. $(\neg a) \notin \mathcal{F}, \neg \notin \mathcal{F}, \neg(\neg a) \notin \mathcal{F}$ and $\neg(a) \notin \mathcal{F}$.

Example 2
Consider now

$$CON = \{\neg\} \cup \{\Rightarrow\},$$

where $\neg \in C_1$ and $\Rightarrow \in C_2$. By definition 2.4 initial recursive step we get for any $a \in VAR$, $a \in \mathcal{F}$. By the recursive step and its repetition we get $\neg a \in \mathcal{F}$, $\neg\neg a \in \mathcal{F}$, $\neg\neg\neg a \in \mathcal{F}$, ... etc., i.e. all formulas from the previous example are in our new $\mathcal{F}$. But also $(a \Rightarrow a)$, $(a \Rightarrow b), \neg(a \Rightarrow b), (\neg a \Rightarrow b), \neg((a \Rightarrow a) \Rightarrow \neg(a \Rightarrow b))$.... etc. are all in $\mathcal{F}$ and infinitely many others.

Observe that $(\neg(a \Rightarrow b))), a \Rightarrow b, (a \Rightarrow)$ are not in $\mathcal{F}$.

Example 3
Take $C_1 = \{\neg, P, N\}$, $C_2 = \{\Rightarrow\}$ and $CON = C_1 \cup C_2$. If we understand P, N as a *possibility* and *necessity* connectives, the obtained language is called a *modal language*. The set of formulas $\mathcal{F}$ contains all formulas from the previous example , but also the expressions of the form: $Na, \neg Pa, P\neg a, (N\neg b \Rightarrow Pa), \neg P\neg a, ((N\neg b \Rightarrow Pa) \Rightarrow b)$, etc.

As we can see from above examples the propositional language is different for different sets of propositional connectives. We will see later that that different sets of connectives may (but may also not) define different *propositional logics.*

When the set of connectives for a given logic is fixed we use the plain $\mathcal{L}$ instead of the notation $\mathcal{L}_{CON}$.

Theoretically the choice of appropriate symbols for logical connectives depends really on a personal preferences of the authors (creators) of different logics, and one can find quite a variety of symbols in the literature, some of them discussed in the introduction.

We will introduce now the formal definitions of *a main connective*, *a sub-formula* and *degree* of a given formula.

Definition 2.6 (Main Connective) *Given a language $\mathcal{L}_{CON}$.*

For any $\bigtriangledown \in C_1$, $\circ \in C_2$,

$\bigtriangledown$ *is called* **a main connective** *of* $\bigtriangledown A \in \mathcal{F}$ *and*

$\circ$ *is* **a main connective** *of* $(B \circ C) \in \mathcal{F}$.

Observe that it follows directly from the definition of the set of formulas that for any formula $C \in \mathcal{F}$, exactly one of the following holds:

1. C is atomic,

2. there is a unique formula A and a unique unary connective $\bigtriangledown \in C_1$, such that C is of the form $\bigtriangledown A$,

3. there are unique formulas A and B and a unique binary connective $\circ \in C_2$, such that C is $(A \circ B)$.

We have hence proved the following lemma.

Lemma 2.1 *For any formula $C \in \mathcal{F}$, C is atomic or has a unique main connective.*

Example 4

1. The main connective of $(a \Rightarrow \neg Nb)$ is $\Rightarrow$.

2. The main connective of $N(a \Rightarrow \neg b)$ is N.

3. The main connective of $\neg(a \Rightarrow \neg b)$ is $\neg$

4. The main connective of $(\neg a \cup \neg(a \Rightarrow b))$ is $\cup$.

Definition 2.7 *We define a notion of direct* **a direct sub-formula** *as follows:*

1. *Atomic formulas have no direct sub-formulas.*

2. *A is a direct sub-formula of a formula $\bigtriangledown A$, where $\bigtriangledown$ is any unary connective.*

3. *A, B are direct sub-formulas of a formula $(A \circ B)$ where $\circ$ is any binary connective.*

Directly from the definition 2.7 we get the following.

Lemma 2.2 *For any formula C, C is atomic or has exactly one or two direct sub-formulas depending on its main connective being unary or binary, respectively.*

Example 5
The formula

$$(\neg a \cup \neg(a \Rightarrow b))$$

has exactly $\neg a$ and $\neg(a \Rightarrow b)$ as direct sub-formulas.

Definition 2.8 *We define a notion of a sub-formula of a given formula in two steps:*

For any formulas A and B, A is **a proper sub-formula** *of B if there is sequence of formulas, beginning with A, ending with B, and in which each term is a direct sub-formula of the next.*

A sub-formula *of a given formula A is any proper sub-formula of A, or A itself.*

Example 6
The formula $(\neg a \cup \neg(a \Rightarrow b))$ has $\neg a$ and $\neg(a \Rightarrow b)$ as direct sub-formula.

The formulas $\neg a$ and $\neg(a \Rightarrow b)$ have a and $(a \Rightarrow b)$ as their direct sub-formulas, respectively.

The formulas $\neg a$, $\neg(a \Rightarrow b)$, a and $(a \Rightarrow b)$ are all proper sub-formulas of the formula $(\neg a \cup \neg(a \Rightarrow b))$ itself.

Formulas (atomic formulas) a and b are direct sub-formulas of $(a \Rightarrow b)$.

Atomic formula b is a proper sub-formula of $(\neg a \cup \neg(a \Rightarrow b))$.

The set of all sub-formulas of

$$(\neg a \cup \neg(a \Rightarrow b))$$

consists of $(\neg a \cup \neg(a \Rightarrow b))$, $\neg a$, $\neg(a \Rightarrow b)$, $(a \Rightarrow b)$, a and b.

Definition 2.9 (Degree of a formula) *By* **a degree of a formula** *we mean the number of occurrences of logical connectives in the formula.*

Example 7

The degree of $(\neg a \cup \neg(a \Rightarrow b))$ is 4.

The degree of $\neg(a \Rightarrow b))$ is 2.

The degree of $\neg a$ is 1.

The degree of a is 0.

Note that the degree of any proper sub-formula of A must be one less than the degree of A. This is the central fact upon mathematical induction arguments are based.

Proofs of properties formulas are usually carried by mathematical induction on their degrees.

2.1 Languages with Propositional Constants

A propositional language $\mathcal{L} = (\mathcal{A}, \mathcal{F})$ is called a language with propositional constants, when we distinguish certain constants, like symbol of truth (T) or falsehood (F), or other symbols as elements of the alphabet. The propositional constants are zero-argument connectives. In this case the set CON of logical connectives contains a a finite (possibly empty) set of **zero argument connectives** C_0, called **propositional constants**, i.e.

$$CON = C_0 \cup C_1 \cup C_2.$$

The definition of the set $\mathcal{F}$ of **all formulas** of the language $\mathcal{L}$ contains now an additional recursive step and is as follows.

The set $\mathcal{F}$ of **all formulas** of the language $\mathcal{L}_{CON}$ is
the smallest set built from the signs of the alphabet $\mathcal{A}$, i.e. $\mathcal{F} \subseteq \mathcal{A}^*$, such that the following conditions hold:

(1) $VAR \subseteq \mathcal{F}$ (atomic formulas),

(2) $C_0 \subseteq \mathcal{F}$ (atomic formulas),

(3) if $A \in \mathcal{F}$, $\bigtriangledown \in C_1$ i.e $\bigtriangledown$ is an one argument connective, then $\bigtriangledown A \in \mathcal{F}$,

(4) if $A, B \in \mathcal{F}$, $\circ \in C_2$ i.e $\circ$ is a two argument connective, then $(A \circ B) \in \mathcal{F}$.

Example Let $\mathcal{L} = \mathcal{L}_{\{T,\neg,\cap\}}$, i.e. $C_0 = \{V\}$. Atomic formulas of $\mathcal{L}$ are all variables and the symbol T. The language admits formulas that involve the symbol T like $T, \neg T$, $(T \cap a)$, $(\neg a \cap \neg T), \neg(b \cap T)$, etc... We might interpret the symbol T as a symbol of truth (statement that is always true).

2.2 Exercises and Homework Problems

This set of problems deals with the formal description of propositional languages. We investigate the syntactic correctness of formulas of a given language. Here is a sample problem.

Exercise 1
Given a language $\mathcal{L} = \mathcal{L}_{\{\neg,\ C, I, \cup, \cap, \Rightarrow\}}$ and a set S of formulas:

$$S = \{\mathbf{C}\neg a \Rightarrow (a \cup b), (\mathbf{C}(\neg a \Rightarrow (a \cup b))), C\neg(a \Rightarrow (a \cup b))\}$$

Determine which of the formulas from S is, and which is not well formed formulas of $\mathcal{L}$.

If a formula is correct, determine its *main connective*. If it is not correct, write the corrected formula and then determine its *main connective*.

If a formula is correct, write what it says. If it is not correct, write the corrected formula and then write what it says.

Solution

1. The formula $\mathbf{C}\neg a \Rightarrow (a \cup b)$ is not a well formed formula.

The correct formula is

$$(\mathbf{C}\neg a \Rightarrow (a \cup b).)$$

The main connective is $\Rightarrow$.

The correct formula says: *If negation of a is possible, then we have a or b .*

Another correct formula is

$$\mathbf{C}(\neg a \Rightarrow (a \cup b)).$$

The main connective is $\mathbf{C}$.

The correct formula says: *It is possible that not a implies a or b .*

2. The formula $(\mathbf{C}(\neg a \Rightarrow (a \cup b)))$ is not a well formed formula.

The correct formula is

$$\mathbf{C}(\neg a \Rightarrow (a \cup b)).$$

The main connective is $\mathbf{C}$.

The formula $\mathbf{C}(\neg a \Rightarrow (a \cup b))$ says: *It is possible that not a implies a or b .*

3. The formula $\mathbf{C}\neg(a \Rightarrow (a \cup b))$ is a correct formula.

The main connective is $\mathbf{C}$.

The formula says: *the negation of the fact that a implies a or b is possible.*

Exercise 2
Given a set S of formulas:

$$S = \{((a \Rightarrow \neg b) \Rightarrow \neg a), \Box(\neg\Diamond a \Rightarrow \neg a)\}.$$

Define a formal language $\mathcal{L}$ to which to which all formulas in S belong, i.e. a language determined by the set S.

Solution

Any propositional language $\mathcal{L}$ is determined by its set of connectives. All connectives appearing in the formulas of the set S are: $\Rightarrow, \neg b, \Box$, and $\Diamond$. Hence the required language is $\mathcal{L}_{\{\Rightarrow, \neg b, \Box, \Diamond\}}$.

Exercise 3
Given formulas:

$\Diamond((a \cup \neg a) \cap b)$ and $\neg(a \Rightarrow (b \Rightarrow))$.

1. Determine their degree,

2. Write down all their sub-formulas.

Solution

1. The degree of $\Diamond((a \cup \neg a) \cap b)$ is 4, the degree of $\neg(a \Rightarrow (b \Rightarrow))$ is 3.

2. $\Diamond((a \cup \neg a) \cap b), ((a \cup \neg a) \cap b), (a \cup \neg a), \neg a, b, a$ are all sub-formulas of $\Diamond((a \cup \neg a) \cap b)$. a, b are atomic sub-formulas, and $\Diamond((a \cup \neg a) \cap b)$ is not a proper sub-formula.

$\neg(a \Rightarrow (b \Rightarrow c)), (a \Rightarrow (b \Rightarrow c)), (b \Rightarrow c), a, b, c$ are all sub-formulas of $\neg(a \Rightarrow (b \Rightarrow c))$, a, b, c are atomic sub-formulas, and $\neg(a \Rightarrow (b \Rightarrow c))$ is not a proper sub-formula.

Homework Problems

Problem 1

Consider the following formulas.

1. $((a \uparrow b) \uparrow (a \uparrow b) \uparrow a)$
2. $(a \Rightarrow \neg b) \Rightarrow \neg a$
3. $\Diamond(a \Rightarrow \neg b) \cup a$, $\Diamond(a \Rightarrow (\neg b \cup a)$, $\Diamond a \Rightarrow \neg b \cup a$
4. $(\Box\neg\Diamond a \Rightarrow \neg a)$, $\Box(\neg\Diamond a \Rightarrow \neg a)$, $\Box\neg\Diamond(a \Rightarrow \neg a)$
5. $((a \cup \neg K\neg a))$, $KaK(b \Rightarrow \neg a)$, $\neg K(a \cup \neg a)$
6. $(B(a \cap b) \Rightarrow Ka)$, $B((a \cap b) \Rightarrow Ka)$
7. $G(a \Rightarrow b) \Rightarrow Ga \Rightarrow Gb)$, $a \Rightarrow HFa$, $FFa \Rightarrow Fa$

For all formulas listed above do the following.

(a) Determine which of the formulas is, and which is not a well formed formula. Determine a formal language of $\mathcal{L}$ to which the formula or set of formulas belong.

(b) If a formula is correct, write what its *main connective* is. If it is not correct, write the corrected formula and then write its *main connective*. If there is more then one way to correct the formula, write all possible correct formulas.

(c) If a formula is correct, write what it says. If it is not correct, write the corrected formula and then write what it says.

Problem 2

For each of the formulas listed below, determine its degree and write down its all proper non-atomic sub-formulas.

1. $(a \Rightarrow ((\neg b \Rightarrow (\neg a \cup c)) \Rightarrow \neg a))$
2. $\Diamond((a \cap \neg a) \Rightarrow (a \cap b))$
3. $\Box\neg\Diamond(a \Rightarrow \neg a)$
4. $\Diamond(\Diamond a \Rightarrow (\neg b \cup \Diamond a))$
5. $(\neg(a \cap b) \cup a)$