Chapter 3: Propositional Languages

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 the following :

All propositional languages contain a set of variables VAR, which elements are denoted by

$$a, b, c, ...$$

with indices, if necessary.

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

We distinguish 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.

We use a notation

$$\mathcal{L}$$

when the set of connectives is fixed.

For example, the language

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

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

The language

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

denotes that a language with two connectives $\neg$ and $\Rightarrow$ adopted as propositional connectives.

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

Symbols for connectives do have intuitive meaning.

Semantics is a formal meaning of the connectives and is defined separately.

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 as a propositional language of **classical** and **intuitionistic** logics, some **many- valued** logics, and **is extended** to the language of **modal** logics.

Several languages can share the same semantics.

The classical propositional logic is the best example of such situation.

Due to functional dependency of classical logical connectives the languages:

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

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

all share the same semantics characteristic for classical propositional logic.

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

We use names 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 modal connectives of **possibility** and **necessity** .

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

We will also use symbols $\mathbf{C}$ and $\mathbf{I}$ for modal connectives of possibility and necessity, respectively.

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

The motivation for notation $\mathbf{C}$ and $\mathbf{I}$ arises from topological interpretation of modal S4 and S5 logics.

In topology $\mathbf{C}$ is a symbol for *a set closure operation*, hence $\mathbf{C}A$ means *a closure* of the set A

$\mathbf{I}$ is a symbol for *a set interior operation* and $\mathbf{I}A$ denotes an interior of the set A.

Modal logics *extend* the classical logic.

A modal logic language is for example

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

Knowledge logics also extend the classical logic by adding a new *knowledge connective* denoted by K. T

A formula KA reads: *it is known that A* or *A is known*.

A language of a knowledge logic is for example

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

Autoepistemic logics extend classical logic by adding *a believe connective*, often denoted by B.

The formula BA reads: *it is believed that A* .

A language of an Autoepistemic logic is for example

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

Temporal logics also extend classical logic by adding temporal connectives.

Some of temporal connectives are: F, P, G, and H.

Intuitive meanings are:
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*.

A language of an Temporal logic is for example

$$\mathcal{L}_{\{\ F,P,G,H,\ \neg,\cap,\cup,\Rightarrow\}}.$$

It is possible to create connectives with more then one or two arguments.

We consider here languages with only **one** or **two argument** connectives.

Propositional Languages
Formal definitions

A propositional language is a pair

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

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

Alphabet is a set

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

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 $a, b, c, ...$ etc, (with indices if necessary).

The set $CON \neq \emptyset$ is a **finite set** of **logical connectives**.

We assume that CON is a **non empty set**, i.e. we assume that **there is** a logical connective.

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

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

PAR is a set of **auxiliary symbols**.

This set may be empty (for example in case of Polish Notation).

We assume here that it contains two parenthesis, i.e.

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

We also assume that the set CON of logical connectives of the language

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

contains only unary and binary connectives.

We write :

$$CON = C_1 \cup C_2$$

C_1 is called the set of all **unary connectives**
,

C_2 is called the set of all **binary connectives** of the language $\mathcal{L}_{CON}$.

The set $\mathcal{F}$ of **all formulas** of a propositional language $\mathcal{L}_{CON}$ is build recursively from the elements of the alphabet $\mathcal{A}$ as follows.

$\mathcal{A}$ is **the smallest set** built from the elements of the alphabet, such that:

(1) $VAR \subseteq \mathcal{F}$

(2) If $A \in \mathcal{F}$, $\bigtriangledown \in C_1$, then $\bigtriangledown A \in \mathcal{F}$.

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

Propositional variables are formulas and they are called **atomic formulas**.

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

A is called its **direct sub-formula** of $\bigtriangledown A$.

$\circ$ is called **a main connective** of the formula $(A \circ B) \in \mathcal{F}$.

A, B are called **direct sub-formulas** of $(A \circ B)$.

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

1. **Main connective** of $(a \Rightarrow \neg Nb)$ is $\Rightarrow$.

$a, \neg Nb$ are **direct sub-formulas.**

2. **Main connective** of $N(a \Rightarrow \neg b)$ is N.

$(a \Rightarrow \neg b)$ is the **direct sub-formula.**

3. **Main connective** of $\neg(a \Rightarrow \neg b)$ is $\neg$.

$(a \Rightarrow \neg b)$ is the **direct sub-formula**.

4. **Main connective** of $(\neg a \cup \neg(a \Rightarrow b))$ is $\cup$.

$\neg a$, $\neg(a \Rightarrow b)$ are **direct sub-formulas.**

Sub-formula definition is defined in two steps:

Step 1 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.

Step 2 A **sub-formula** of a given formula A is any proper sub-formula of A, or A itself.

The formula $(\neg a \cup \neg(a \Rightarrow b))$ has direct sub-formulas: $\neg a$ and $\neg(a \Rightarrow b)$.

The direct sub-formulas of $\neg a$ and $\neg(a \Rightarrow b)$ are a and $(a \Rightarrow b)$, respectively.

The direct sub-formulas of a, $(a \Rightarrow b)$, are a, b.

END of the process.

The set of all **proper sub-formulas** of $(\neg a \cup \neg(a \Rightarrow b))$ is

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

The set of all **sub-formulas** of $(\neg a \cup \neg(a \Rightarrow b))$ is

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

A degree of a formula is number of occurrences of logical connectives in the formula.

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.

Observation: the degree of any proper subformula of A must be one less than the degree of A.

This is the central fact upon which mathematical induction arguments are based.

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

Exercise 1

Consider a language $\mathcal{L} = \mathcal{L}_{\{\neg, \ \Diamond, \Box, \cup, \cap, \Rightarrow\}}$ and a set S of formulas:

$$S = \{\Diamond\neg a \Rightarrow (a \cup b), (\Diamond(\neg a \Rightarrow (a \cup b))),$$

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

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

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

is not a well formed formula.

The correct formula is

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

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

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

The main connective is $\Diamond$.

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

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 all formulas in S belong, i.e. a language determined by the set S.

Solution

All connectives appearing in the formulas in S are: $\Rightarrow, \neg b, \Box$ and $\Diamond$.

The language is $\mathcal{L}_{\{\Rightarrow, \neg b, \Box, \Diamond\}}$.

Exercise 3

For a given formula:

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

Determine its degree.

Write down all its sub-formulas.

Solution

The degree is 4.

All sub-formulas are:

$$\Diamond((a \cup \neg a) \cap b), ((a \cup \neg a) \cap b),$$

$$(a \cup \neg a), \neg a, b, a.$$