# Symbolic Artificial Intelligence
# A soft introduction to propositional logic for AI

Pierre Mercuriali

**Abstract**

The purpose of this document is to introduce in a gentle manner some (not all!) important concepts in logic and symbolic artificial intelligence, in particular propositional logic.

## Contents



# 1 Introduction

## 1.1 Course contents

Symbols allow us to talk and reason about things that are not here. If I want to tell you that I think hagelslag, which are small chocolate bits commonly eaten on bread and butter in the Netherlands, is tasty, I do not need an actual pack of hagelslag to do so, provided you know what the symbols

- `hagelslag`,

- `is`, and
- `tasty`

refer to, and that you know what happens when I combine those symbols into the sentence

`hagelslag is tasty`.

I can simply use the symbol `hagelslag`, written with letters, or pronounced with phonemes, to refer to real hagelslag, or the general concept of hagelslag. I can also use a drawing of a hagelslag, or a photo of a pack of hagelslag.

There is thus a *communicative* function to this symbol: I am thinking about something, and I want to think about the same thing I am thinking of, so I use a specific symbol to do so, hoping that we share the common understanding of the symbol, or, in other words, that we interpret the symbol in the same way.

Furthermore, I can also *reason* about the knowledge that I have of hagelslag, myself, and, in general, the world around me. For example, I know that hagelslag is a tasty thing. I also know that I happen to like tasty thing. So, can you, or myself, deduce that I like hagelslag?

One way to approach artificial intelligence (AI) is to make it so the AI, be it a robot, a chatbot, a computer or a global acentric artificial intelligence set out to destroy Humanity, can do all those things we have hinted at just now.

1. We can represent our knowledge about the world using symbols,
2. we can use those symbols to communicate with each other, to share this knowledge about the world, as we are doing right now, and
3. we can manipulate those symbols internally, with our brains, to deduce and discover new knowledge about the world.

In this document about symbolic AI, we will discover how robots and humans can do these things, drawing examples from history, literature, and scientific research.

First, in Sections 2 and 3 we will explore the notion of symbols and properly define it. Second, in Section 4, we will discover what it takes to represent knowledge about the world using symbols, and to manipulate that knowledge, using logic. Specifically, we will use the logic language called Propositional Logic (PL) to express and, later, deduce knowledge about the world using knowledge we already have. Third, in Sections 5, we will talk about truth and reasoning about truth, and mention some fun and insightful pitfalls that AI (and us) can fall into and that symbolic reasoning aims at overcoming. Fourth, in Section 6, we will briefly mention other logics, slightly different from PL, and some motivations to use them.

### 1.2 Learning objectives

In Table 1, we summarized the various learning objectives (LOs) that you will reach during this course. In Appendix A, you will find more details on the LOs and examples of exercises you may be asked to solve.

> ☞ More on that...
>
> An information box such as this one accompanies the text, usually at the end of the sections, to indicate resources and themes to look up.

## 2 What are symbols?

### 2.1 Signs

We shan't go too deeply into the intricacies of the philosophical theories of symbols and referenes, and the study of symbols, also called *semiotics*, but if you are curious, I have provided references throughout the text. Symbols, also called signs by some authors, allow us to talk and reason about things that are not there, things that are abstract, and to convey information, complex or simple.

In cities and roads all over the world, signs are all over the place as... signs. When you see a (red) octogonal sign, you might know that it means 'STOP', even if the word STOP or the actual action of stopping is neither written, nor shown. Sometimes, signs are more explicit indeed: instead of an abstract green round light that indicates that a pedestrian can now cross the street, the sign is a glowing light in the shape of a person walking.

| Section/video | Section | Learning Objectives | Relevant terms |
|---|---|---|---|
| Presentation of the subject | 1 | LO1: knowing about the learning objectives | |
| What are symbols? | 2 | LO2: being able to give examples of signs in the world around you<br>LO2': knowing the etymology of the word 'symbol' and relating it to its modern use | symbol, sign, explainable AI |
| Why use symbols and logic to represent knowledge about the world? | 3 | LO3: explaining some ambiguities in natural language or robotics and explaining how logic can solve them<br>LO3': giving at least two motivations for the use of logic | ambiguity, syntax, semantics, grounding<br>formal reasoning, inference, deduction |
| How to build formulas in propositional logic? | 4 | LO4: recognising and building a well-formed formula in PL<br>LO4': building a formula to model a real problem<br>LO4": reading a formula and translating it in natural language | recursively defined, syntax, operator names and symbols<br>modelling, variable grounding |
| Interpretation(s) and truth(s) | 5 | LO5: determining if a formula is valid/satisfiable or not by building its truth table | validity, satisfiability, truth table, interpretation |
| Limits of PL. Are there other logics? | 6 | LO6: knowing about other types of truth values<br>LO6': knowing about the limits of PL and what other logics to use for what purposes | multi-valued logic: fuzzy logic<br>first order logic, modal logic, temporal logic, fuzzy logic, description logics |

Table 1: Learning Objectives (LOs) of the course.

## 2.2 Etymology: symbols

Etymologically[1], the word 'symbol' itself comes from the ancient Greek 'σύμβολον' (sumbolon) meaning

> 'to compare', 'to put things together'.

The word itself comes from the words 'sum' ('together') and 'ballo' ('I throw, I put'). In Ancient Greece, this word had a very concrete meaning. During a commercial deal for example, two contracting parties would break a piece of pottery in two. Each party would keep one of the pieces, which would serve as a piece of identity when they were matched together, as a key and a lock, or Cinderella's foot to its delicate yet dangerous glass slipper. In Latin and in the Roman world, later, the meaning evolved to a more general

> 'mark or sign as proof of recognition',

or even

> 'a proof of identity',

in the similar way as an identity card nowadays may refer to the actual person, the physical, real you (or its legal equivalent – how many representation layers deep are we in?). Later again, the meaning evolved to

> 'a natural fact or object evoking by its form or its nature an association of ideas with something abstract or absent'

- and there we recover the idea I have presented at the beginning: representing the abstract, or something that is not here. In English, the specific meaning for 'something which stands for something else' appears in Edmund Spenser's *Faerie Queene*[2] (1590). In this part of the story, one of the heroes, Sir Guyon, knight of Temperance (the Christian virtue similar to the modern value of *self-control* and resistance to temptation) comes across a woman named Amavia in the act of suicide, for her lover has been poisoned by the witch Acrasia. She stabs herself in the chest and dies in the knight's arms, and he is left caring for her child. He attempts to clean the baby but his mother's blood will not come off. The blood acts as a *symbol* of his mother's innocence and chastity, a reminder of his mother's death, and a call to revenge.

| | |
|---|---|
| From thence it comes, that this babes bloudy hand | Thus it comes that this baby's bloody hand |
| May not be clensd with water of this well: | May not be cleaned with water of this [magical] spring: |
| Ne certes Sir striue you it to withstand, | Neither assuredly sir, strive you to withstand, |
| But let them still be bloudy, as befell, | But let them always be bloody, as it befell, |
| That they his mothers innocence may tell, | So that they may tell his mother's innocence, |
| As she bequeathd in her last testament; | As she bequeathed in her last testament; |
| That as a sacred Symbole it may dwell | So that it may remain as a sacred *symbol*, |
| In her sonnes flesh, to minde reuengement, | In her son's flesh, to bring revenge to mind, |
| And be for all chast Dames an endlesse moniment. | And be for all chaste dames an endless monument. |

Edmund Spenser's *Faerie Queene*, Book II, Canto II (Translation/adaptation P.M.)

In fact, and as you might have already noticed, what I am using right now to communicate with you can be, and have been, considered symbols. When I use words in English or Dutch, I use them to refer to meanings that are abstract or not, with the assumption that we are in rough agreement over the meaning of enough words that we can communicate efficiently.

## 2.3 Communication

We will go deeper into the links between symbols and meanings in Section 5. For now, I would like to motivate further the use of symbols and logic in A.I. The theologian and philosopher Augustine of Hippo (354 - 430) who lived in current-day Algeria, wrote, of signs:

> "There is no reason for signifying, i.e., for giving signs except to convey into another's mind what the sign-giver has in his own mind."
> Augustine, De doctr. chr. II 3, 1963, 34: 17–20

[1] https://www.etymonline.com/search?q=symbol

[2] http://www.luminarium.org/renascence-editions/fqintro.html; see also https://www.gutenberg.org/cache/epub/6930/pg6930.txt

If we can agree, with robots, on the meaning(s) of some signs, then we can communicate with the robots, express our intentions to them, but also understand them and what they are doing with those signs, what reasoning they are doing by manipulating those signs. This is part of a new trend in A.I. called Explainable A.I.: humans find it important to know what, and how the robot reaches a certain result. This is particularly important when handling and understanding failures, to be able to fix or prevent them.

☞ More on that...

- Medieval semiotics at https://plato.stanford.edu/entries/semiotics-medieval/. The SEP (Stanford Encyclopedia of Philosophy) contains excellent articles on many areas relevant to logic, language, intelligence, and A.I., to only cite a few philosophical themes.
- Explainable A.I. (XAI)
- Trustworthy A.I.

# 3 Why use symbols and logic to represent knowledge about the world?

In the context of the course, logic refers roughly to a language, that is a set of sentences, alongside a way to manipulate those sentences and relate them together. We will talk about two main motivations for logic:

1. eliminating ambiguities, and
2. studying formal structures of reasoning.

When we use natural language such as French, English, Dutch, Chinese, Arabic..., we sometimes stumble upon ambiguities, that I will illustrate with the gloomy example of brazen heads. In early modern period myths, that is, from circa 1400 to 1800, brazen heads are humanoid heads cast in bronze (brazen, in other words) that were attributed magical or unexplained powers of prediction and truth-telling.

## 3.1 Ambiguity

Gerbert of Aurillac (c.946 - 12 May 1003) was a French-born scholar who later became Pope Sylvester II. He was an accomplished scientist who built the first mechanical clock and a hydraulic-powered organ. He studied and wrote on arithmetic, geometry, astronomy, music, and also grammar, logic, and rhetoric. Amongst the legends surrounding the character, it is said that he build one of those *brazen heads*, a physical device based on astrology that could tell him the truth, after stealing a spell book from a Saracen philosopher in Spain, and making a pact with the devil to become Pope. The device will ultimately cause his own demise. I have slightly altered and simplified the following version of the myth: please refer to the info box at the end of this section for the source of the original by William of Malmesbury.

Gerbert thus, after he has obtained the spell book, casts the head of the statue. The statue has only a few, albeit very powerful, properties.

1. It does not speak unless spoken to (not quite the most powerful property, though useful when Gerbert wants to sleep).
2. It pronounces the truth by answering either YES or NO.

Gerbert thus asks the statue:

> 'Will I become Pope?'

and the statue to answer:

> 'YES.'

Gerbert is satisfied, and asks another question:

> 'Will I die before I sing the mass at Jerusalem?'

| Logical sentence | Brazen head answer |
|---|---|
| Gerbert_will_be_Pope | YES |
| IF Gerbert_sings_mass AND Gerbert_at_Jerusalem THEN Gerbert_dies | NO |

Table 2: Gerbert's dialogue, a bit more formal.

and the statue to answer:

> 'NO.'

Gerbert does not think much about this, and merely decides never to visit Jerusalem, referring to the city in the Middle East. As he understands, if he never sings the mass at Jerusalem, then he will not die! He becomes Pope Sylvester II, and fulfills the first prophecy of the brazen head. Content with his duties, he sings the mass in various churches in Rome. However, in Rome, there is a certain church called Saint Mary of Jerusalem, that is commonly called by the people 'Jerusalem'. Gerbert unknowingly sings the mass there and, as he prepares, he suddenly becomes violently ill. Losing his mind to the sickness, he becomes so certain of his approaching death that he calls for his followers to dismember him and spread his remains across the city of Rome as penance for his unholy crimes.

There is clearly a problem there. Gerbert certainly did not want to die, and yet was reassured by the brazen head's answers. To analyse the miscommunication problem, let us rewrite the dialogue in a more formal way in Table 2.

One of the problems lies in what is called the *grounding of the variable* `Gerbert_at_Jerusalem`, that is, how it relates to the real world. Gerbert tries to make this variable FALSE by *not* singing at Jerusalem, in which case he will not die no matter where he sings – an important duty for the Pope. But he unfortunately fails to realise that 'Jerusalem' also refers to a church in Rome. The brazen head did not lie, and Gerbert's reasoning is sound: but the brazen head was referring to the small church in Rome called Jerusalem, or, perhaps, any place called Jerusalem. How would you have solved this problem, if you had been in Gerbert's place (and more cautious than he)?

## 3.2 Studying formal structures of reasoning

Studying precisely what happened in Gerbert's story allowed us to exhibit where the problem is and one of Gerbert's mistakes. Now, there is another motivation for using formal logic in A.I. and to study human reasoning in a formal way. Look at these very similar sentences.

1. Gerbert dies if he sings the mass at Jerusalem. Gerbert sings the mass at Jerusalem. Therefore, Gerbert dies.
2. Socrates is a man. All men are mortal. Therefore, Socrates is mortal.
3. Plato is a philosopher. If someone is a philosopher, then they are strong! Therefore, Plato is strong.
4. Socrates is famous. A famous person is also well-known. Therefore, Socrates is well-known.

All those sentences follow the same structure, and involve the same kind of reasoning, as hinted by the symbol (word) 'therefore'. This structure is present regardless of the other symbols that we use: 'man', 'philosopher', 'Plato', 'strong'... Aristotle, the Greek philosopher, started to conceive the idea of studying these general 'reasoning structures', that he called *syllogisms*. This is what formal logic consist of: the study of these reasoning structures. Philosophers and logicians such as Leibniz, Frege, Peano and Russell investigated truth and the reality of the world using artificial formal languages, such as logic. In the next section, we will define such a formal artificial language, called *propositional logic*.

| Connective | How to read it | Name | Example |
|---|---|---|---|
| `¬A` | 'not A' | negation | Socrates is not young. |
| `A ∧ B` | 'A and B' | conjunction | Carol Karp and Susanne Langer are logicians. |
| `A ∨ B` | 'A or B' | disjunction | 'Are you a philosopher or a logician?' 'Yes.' |
| `A ⇒ B` | 'A implies B', or 'if A, then B' | implication | If Gerbert sings, then he dies. |
| `A ⇔ B` | 'A if and only if B', or 'A is equivalent to B' | equivalence | The brazen head is a head made of bronze (something is a brazen head if and only if it is a head, and it is made of bronze). |

Table 3: Usual connectives in propositional logic.

☞ More on that...

- The folly of humans dealing with absolutes (such as the absolute truth of the brazen head) is a recurrent theme in literature and stories about artificial intelligence. When the brazen heads do not fail outright to work because of the human qualities of the maker (such as Bacon falling asleep, exhausted, at a crucial point in the construction of his own brazen head), they lead to their makers' demise, because of the maker's errors, as with Gerbert's, or even out of the head's own malevolent intent. See, for example, LaGrandeur's The Persistent Peril of the Artificial Slave.
- Loglan and its successor Lojban are conlangs (constructed languages) that aim at eliminating ambiguities in natural language. `https://omniglot.com/writing/lojban.htm`, `https://mw.lojban.org/index.php?title=Lojban&setlang=en-US`.
- Garden path sentences, donkey sentences, paraproskodia, are some language effects that rely on ambiguities.
- William of Malmesbury wrote about Gerbert in his Chronicle of the Kings of England. Exercise caution however: he may have been biased against Gerbert, as noted in `https://www.icysedgwick.com/brazen-head/` (a great article on brazen heads).
- Why did I mention philosophers being strong? Socrates, in one of Xenophon's dialogues (Memorabilia 3.12), mentions the importance of taking care of one's own body, for the city – because, according to him, one needs a fit body to defend one's city, and go to war on its behalf.

# 4 How to build formulas in propositional logic?

In natural languages like English, and without going too much into details regarding complex grammatical rules, you construct sentences using words. Words can be seen as building blocks, and there are rules that tell you how to put those words together, commonly referred to as grammatical rules. For example, before a noun, you can find an article, as in "the cat", or "a robot". In logic, this is much the same thing: you have at your disposal building blocks, that are symbols, and rules to combine them. In this section, I will show you how to build sentences, also called formulas, of propositional logic.

First, consider a set of symbols, also called *logical connectives*, or *logical operators*

$$Op = \{\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow, (, )\}$$

and read as described in Table 3.

I personally always confuse 'disjunction' and 'conjunction', just like right and left. One way to remember which is which is to remember that *con*, in Latin, means 'with', 'together', which is a cognate (related in meaning) of 'and'.

The parentheses have a syntactic function only: they can make the reading of formulas explicit, like in mathematics. Without any rule on the precedence of operators + and ×, the mathematical expression $1 + 2 \times 3$ is ambiguous, but $(1 + 2) \times 3$ is not. It can be evaluated, starting from what is inside the deepest parentheses (if I am not mistaken, it is equal to 9). Similarly, the logical formula $A \wedge B \vee C$ is ambiguous, but $(A \wedge B) \vee C$ is not. In propositional logic, the operators $\Rightarrow$ and $\Leftrightarrow$ have precedence over $\wedge$ and $\vee$.

Now, consider a set of symbols $Var$, that is a set of names such as 'woman', 'Plato', 'strong', 'AdaLovelace', or even 'X', 'Y', 'Z', 'A', 'B'. We call them *variables*. Finally, consider another set of *constant* symbols (also simply called *constants*) $\{T, F\}$, read as 'True' and 'False', respectively. The sets of connectives $Op$, of variables $Var$, and the constants $\{T, F\}$ are pairwise disjoints. The elements of $Var$ are also called the *propositional variables*.

Now, we have all the *building blocks* we need to build our formulas! Given the sets $Op$, $Var$, and $\{T, F\}$, the *set of propositional formulas* is inductively (also called recursively) defined as follows.

- The constants T and F are *atomic* formulas. They are called atomic because they are the smallest components of formulas.
- All propositional variables are also atomic formulas.
- If $f$ and $g$ are formulas, atomic or not, then $\neg f$, $f \wedge g$, $f \vee g$, $f \Rightarrow g$, $f \Leftrightarrow g$, and $(f)$ are also formulas.

For example, the expression

$$\texttt{Gerbert\_sings} \Rightarrow \texttt{Gerbert\_dies}$$

is a formula in PL, where $\{\texttt{Gerbert\_sings}, \texttt{Gerbert\_dies}\}$ is the set of propositional variables that appear in the expression. The expression

$$\texttt{Gerbert\_sings\_mass} \wedge \texttt{Gerbert\_at\_Jerusalem} \Rightarrow \texttt{Gerbert\_dies}$$

is also a formula in PL. As an exercise, try to formulate what both of these formulas mean in natural language (English, Dutch, French, etc.).

Note that several answers to the previous exercise are possible. There is not an exact correspondence between logic and natural language, or between logic and the world. This is related to the problem of *modelling*: finding a close enough approximation of real life that we can get meaningful results or predictions out of it.

Finally, the expression

$$(\texttt{Gerbert\_sings\_mass} \wedge) \texttt{Gerbert\_at\_Jerusalem} \Rightarrow \neg\neg \texttt{Gerbert\_dies}$$

is *not* a formula of propositional logic. Can you see why?

☞ More on that...

- Chapter 7, 8, and 10 from Russell SJ, Norvig P. Artificial Intelligence: A Modern Approach. (Fourth edition, 2021) provide a thorough introduction of propositional logic,
- as well as Delftse Foundations of Computation, Hugtenburg & Yorke-Smith, TU Delft Open 2019, textbooks.open.tudelft.nl, ISBN 978-94-6186-952-4.

# 5 Interpretation(s) and truth(s)

So far, in PL, we have been looking at formulas made up of symbols, built following a small set of construction rules in a mechanical manner. But how can we relate our formulas to whatever situation, real or fictional, we want to describe using logic?

## 5.1 Interpretations

Consider the case of Gerbert singing. At any moment, we consider (this is already a modelling decision!) that Gerbert is either singing, or is not singing. We can capture this modelling decision by using the propositional variable

$$\texttt{Gerbert\_sings}.$$

In itself, it does not say much about the state of the world: this is just a symbol. We have seen however that symbols carry meaning. In this case, there are two obvious meanings to the symbol `Gerbert_sings` : either Gerbert

| A | B | (A) | ¬A | A ∧ B | A ∨ B | A ⇒ B | A ⇔ B |
|---|---|---|---|---|---|---|---|
| T | T | T | F | T | T | T | T |
| T | F | T | F | F | T | F | F |
| F | T | F | T | F | T | T | F |
| F | F | F | T | F | F | T | T |

Table 4: Definition of the semantic behaviour of our connectives.

is singing, in which case we say that the variable `Gerbert_sings` is TRUE, or Gerbert is not singing, in which case the variable `Gerbert_sings` is FALSE.

In propositional logic, we can formally define the notion of truth, meaning, semantic content of a formula, through what we call an *interpretation* of a formula. Interpreting is very close to what you do during communicating: you are trying to understand what your friend is saying by giving meaning to the phonemes they say and their body movements!

In PL, there are two truth values; T, for 'true', and F, for 'false'. Notice there that I am using the same symbol for the constants T and F in the formulas. This is an abuse of notation, but since there is no ambiguity, this is acceptable!

An *interpretation* $\mathcal{I}$ is a mapping (or function) $\mathcal{I} : Var \to \{T, F\}$ that assigns, to every propositional variable, a truth value: either $T$ or $F$.

In the previous example, if `pope` is a variable that works as a stand-in symbol to express the fact that Gerbert is Pope, we can say if it is true or not using the interpretation. If in our situation Gerbert is not yet Pope, then we will write

$$\mathcal{I}(\texttt{pope}) = F$$

thereby linking symbols to the corresponding situation. Note that to each different world situation may correspond a different interpretation (in this case, $\mathcal{I}$ corresponds to a situation where Gerbert is not yet Pope).

It does not end here! We can also interpret *entire formulas*; not just atomic formulas. Formulas that are not atomic contain connectives, so we need to describe how the connectives behave semantically. We already have a natural understanding of how 'and' or 'not' behave, for example; thanks to an interpretation, we can formally define this natural understanding.

For example, to compute the truth value of the formula

$$\texttt{will\_be\_pope} \wedge \texttt{will\_die},$$

given an interpretation $\mathcal{I}$, we need to know the truth values $\mathcal{I}(\texttt{will\_be\_pope})$ and $\mathcal{I}(\texttt{will\_die})$ of the propositional variables `will_be_pope` and `will_die`, and we need to know how the connective $\wedge$ behaves semantically. Here, $\mathcal{I}(\wedge)$ is actually a *function* with two inputs. Suppose as in the story that $\mathcal{I}(\texttt{will\_be\_pope}) = T$ and $\mathcal{I}(\texttt{will\_die}) = T$. Then, the truth value of the formula is

$$\begin{aligned} \mathcal{I}(\texttt{will\_be\_pope} \wedge \texttt{will\_die}) &= \mathcal{I}(\wedge)(\mathcal{I}(\texttt{will\_be\_pope}), \mathcal{I}(\texttt{will\_die}))) \\ &= \mathcal{I}(\wedge)(T, T) \\ &= T, \end{aligned}$$

something that Gerbert failed to conclude.

The last step in the previous computation is by definition of the function $\mathcal{I}(\wedge)$. One way to define our basic connectives $\{\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow\}$ is to describe their behaviour exhaustively, that is, to range over every possible interpretation of the inputs and give their output. This is what is called building a *truth table*. For example, to define the negation $\neg$, we give the value of $\neg$A for every possible value of A: if A is (interpreted as) true, then $\neg$A is (interpreted as) false; if A is false, then $\neg$A is true.

In Table 4, I give the truth table of all our connectives. You can see on the first two columns that we are indeed ranging over every possible truth values for the variables A and B.

Remark that some interpretations do not correspond to the usual interpretation of the connectives in natural languages; In English, when we use the word 'or', we typically use it as an *exclusive or*, like in the sentence

'Do you want coconut `or` chocolate flavoured ice-cream?'

In logic, we typically instead use the *inclusive or*, where 'A or B' is interpreted as true as long as at least one of the variables is interpreted as true - much to the delight of smarty-pants students of logic who will answer 'yes' to such questions.

Note that it is perfectly possible to define your own connectives. Some currents of logic use the exclusive or extensively (especially in Boolean circuit theory); in this course, we use these specific definition because they are common in modern logic literature. We will assume that all the connectives in this documents are thus interpreted like in Table 4.

Now, let us interpret the value of another formula using the definitions of Table 4. Consider the propositional logic expression

$$p := (\texttt{gerbert\_sings} \wedge \texttt{gerbert\_at\_Jerusalem}) \Rightarrow \neg\texttt{gerbert\_dies}$$

and the variable assignment (the interpretation):

$$\mathcal{I} = \{\texttt{gerbert\_sings} : F, \texttt{gerbert\_at\_Jerusalem} : F, \texttt{gerbert\_dies} : T\}.$$

(The symbol ':=' indicates that I define $p$ as the symbol that stands for the formula).

We determine that

$$\begin{aligned}
\mathcal{I}(p) &= \mathcal{I}((\texttt{gerbert_sings} \wedge \texttt{gerbert_at_Jerusalem}) \Rightarrow \neg\texttt{gerbert_dies}) \\
&= (\mathcal{I}(\texttt{gerbert_sings})\mathcal{I}(\wedge)\mathcal{I}(\texttt{gerbert_at_Jerusalem}))\mathcal{I}(\Rightarrow)\mathcal{I}(\neg)\mathcal{I}(\texttt{gerbert_dies}) \\
&= (F\mathcal{I}(\wedge)F)\mathcal{I}(\Rightarrow)\mathcal{I}(\neg)T && \text{by definition of } \mathcal{I} \\
&= F\mathcal{I}(\Rightarrow)F && \text{by Table 4} \\
&= F\mathcal{I}(\Rightarrow)F && \text{by Table 4} \\
&= T.
\end{aligned}$$

Reflect upon this result. Is this surprising? What does the surprise come from? Provide an explanation of the formula $p$ in natural language.

**Remark on notation** There are several ways of writing down the functions $\mathcal{I}(a)$ when a is a logical connective. In the computation above, I have chosen to keep the infix notation like in $F\mathcal{I}(\wedge)T$ that mirrors the syntactic formulas in propositional logic, but it is perfectly acceptable to use a functional notation like in $\mathcal{I}(\wedge)(F, T)$; the choice boils down to what is more convenient, readable, and what prevents ambiguities. Some papers even use the Polish notation, like in $\mathcal{I}(\wedge)FT$, or the reverse Polish notation $FT\mathcal{I}(\wedge)$ (which is actually quite convenient once you get used to it - remember calculators?)!

## 5.2 Satisfiability

If a formula has a variable assignment that makes it true, then it is said to be *satisfiable*. A formula is *valid* if all variable assignments (i.e., all interpretation) make it true. It is also called a *tautology*. [3]

In general, it is computationally expensive, in the sense that it requires a lot of computing time and memory (for a certain definition of 'a lot' that is outside the scope of this course!) to figure out if a formula is satisfiable or not. 'SAT solving' refers to the research area that is interested in coming up with efficient algorithms and studying formally the general problem of finding if a formula is satisfiable.

Amazingly, very concrete problems can be modelled in formal logic as a SAT problem. For example,

- the travelling salesman problem: how to visit a graph (towns relied to each other by roads) in a minimal amount of time?
- software verification: when will a piece of code fail?
- air-traffic control,
- scheduling of sports tournaments...

[3] You might have heard this term in linguistics, where the meaning is very similar. There, it designates a redundant statement, something that 'says the same thing twice'. For example: a 'free gift', 'enough is enough'. See also: 'RAS syndrome'.

☞ More on that...

- Isn't it surprising that 'False implies False' is interpreted as True, and that, more generally, 'False implies anything' is always interpreted as True no matter the truth value of 'anything'? This is a feature of the 'material conditional' which is how we define the implication in this course. Some logicians have criticised this non-intuitive behaviour. See https://en.wikipedia.org/wiki/Material_conditional.
- Satisfiability problem (SAT - not to be confused with the US' 'Scholastic Aptitude Test')
- Constraint satisfaction problems
- Computational complexity
- Very Large Scale Integration (VLSI) for circuit synthesis: how to build logic circuits (as in computer chips, for example) that behave as we want them to behave?
- Boolean Functions: Theory, Algorithms, and Applications by Peter L. Hammer and Yves Crama is an excellent book that covers propositional logic in particular, but also circuit theory and problem solving in general, through the lens of Boolean functions - what we interpret our connectives as.

# 6 Limits of PL. Are there other logics?

## 6.1 Other connectives

Now, one important thing to remember is that most of the rules and definitions for connectives we have seen correspond to *one* possible definition for a logic, suitable for a certain purpose. Propositional logic is a language, much like musical notes on a partition or tablatures. Remark that it sheet music is more suited to sharing and communicating music than sharing and communicating cooking recipes!

Some constructions of PL omit the equivalence operator $\Leftrightarrow$, because it can be defined using the (material) implication $\Rightarrow$. Indeed, if you have taken a course in mathematical logic, you might know that $\mathtt{A} \Rightarrow \mathtt{B}$ and $\mathtt{B} \Rightarrow \mathtt{A}$ is equivalent to $\mathtt{A} \Leftrightarrow \mathtt{B}$ (to convince yourself, build the truth tables of both $\mathtt{A} \Rightarrow \mathtt{B} \wedge \mathtt{B} \Rightarrow \mathtt{A}$ and $\mathtt{A} \Leftrightarrow \mathtt{B}$). In fact, using one special operator called the Sheffer's stroke and the constants T and F, it is possible to build any kind of formula you want! That is, for every formula that you can make using the connectives from Tables 3 and 4, you can build a logically equivalent formula, using only this connective and the constants T and F! Of course, this makes formulas rather hard to read, so we have kept the traditional, classical connectives and, or, implies, not... because they are close to natural language and relatively easy to understand.

## 6.2 Complexity of modelling

Let us continue our discussion on the complexity of modelling. Imagine if Gerbert had wanted to describe his position in the city of Rome by writing down the street name, and the street number of the building he is currently at. In propositional logic, he would then use variable symbols such as

`Gerbert_at_number_3_of_street_Appia_Antica`,

and would do so for every number... and every street name. If he wanted to accurately model all his possible positions in Rome, even with only 10 street numbers and 100 street names (a conservative assumption!), he would already need 1000 different variables!

Furthermore, if he wanted to describe the position of his brazen head in Rome, he would need and additional 1000 variables, such as

`my_brazen_head_at_number_7_of_street_Via_dei_Fori_Imperiali`!

This quickly becomes impossible to manage, and propositional logic is, in fact, not suitable for these purposes. One solution is to use another kind of logic, called *predicate logic* or *first-order logic*, that permits the use of special *function symbols*. Functions symbols allow you to access properties of the objects that you manipulate, such as the position of a variable. Thus, Gerbert can now write formulas such as

$$\mathit{position}(\mathtt{Gerbert}),$$

and

$$\mathit{position}(\texttt{my\_brazen\_head}),$$

which only require three symbols, instead of the 2000 in propositional logic.

There is a trade-off, however. Giving such representative power to our logic makes it much, much more complicated to solve problems in the general case. People sometimes use what is called a *fragment* of first-order logic, that is a little bit more powerful than propositional logic, but not so much that it makes automated reasoning impossible for material reasons.

## 6.3 Other types of interpretations

Another way to access different logics and better model certain problems is to change the nature of the interpretation map.

So far, we have seen that $\mathcal{I}$ will assign, to any variable, a value: true, or false. This is quite natural when it comes to describing things that can either be true of false, such as

`gerbert_sings_the_mass`

or

`gerbert_dies`.

But some properties can be much harder to capture that way. Consider the kind of properties onto which, in English, we can use the adverb 'very', as in 'very evil' , 'very rich', 'very old', or 'very nice'. Compare with 'very singing the mass' or 'very dead' (which we can capture linguistically; but this is outside of the scope of this course)!

If we increase the number of truth values the interpretation can take, we can capture this effect. For example, in *multi-valued logic*, we might decide to interpret variables on a scale from 0 to 5, with 5 corresponding to someone who is very nice, and 0 corresponding to someone who is not very nice (very not nice); and 3 someone who is nice, but less nice than someone who has a score of 5...

*Fuzzy logic* takes it even further in the continuity by assigning real values to variables, for example between 0 and 1. Something that is without question nice will be interpreted as 1, and something that is without question *not* nice will be interpreted as 0; so far, nothing different from propositional logic. But something that has a score between 0 and 1 (strict) will be both nice and not nice to varying degrees. Amongst other things, it allows people to perform more subtle comparisons, here, regarding niceness. In systems and controls, it can model the probabilities of failure of systems and elements within a system, and can allow an A.I. to choose the best behaviour given a certain situation.

☞ More on that...

- Some discussion on the properties of things and the nature of the properties of things can be found in Aristotle's *Organon*.
- Sheffer stroke, Peirce's arrow.
- Another great article from the SEP: https://plato.stanford.edu/entries/logic-firstorder-emergence/, on the history of First-order logic. Search in the SEP for other types of logic:
- Fuzzy logic, multi-valued logic, temporal logic, description logics.
- Description logics, that are a particular class of logic with applications in symbolic A.I. and expert systems, provide a nice middle ground between not enough computational power for certain real life applications (as in PL) and so much that some computations become impossible (as in FOL). See A Description Logic Primer, for a thorough yet pleasant to read introduction https://arxiv.org/abs/1201.4089.
- (for advanced logicians) To give an idea of what strange things happen when we use logics that are 'too powerful'. https://en.wikipedia.org/wiki/G%C3%B6del%27s_incompleteness_theorems and https://blog.plover.com/math/Gdl-Smullyan.html.

# 7 General conclusion

Let us conclude and wrap up what we have learned during this series.

We have learned what *symbols* are. We have learned what *uses* symbols have: to communicate and eliminate ambiguities. We have learned what logic is. We have learned what *uses logic* has: We have studied a particular kind of logic: *propositional logic*. We have learned how to *build formulas* in propositional logic, and how to read them. We had only focused on the syntax: but we also considered the notion of *truth*, or *interpretation*, of a formula, to relate formulas to the world. Finally, we saw some examples of *other logics* and ways of representing truth, that are suitable for different problem.

I hope this document has given you a taste of symbolic logic in AI and has given you some resources to research it more, and I wish you good luck in your future logical endeavours!

☞ `What now?` ☕

Here are some questions that you may be asking yourself. They are out of the scope of this course, but the answers to them can be found in the books cited at the end of Section 4 and, for example, at `https://iep.utm.edu/prop-log/#H5`.

- Natural deduction. How do we make robots and AIs reason and discover new knowledge mechanically from what we already know?
- Are there rules to manipulate formulas? Can we simplify formulas? Can we combine formulas to produce new ones?
- Can we prove mathematical theorems with a computer?

# A Detailed presentation of some of the Learning Objectives

In this section I give examples of the skills that you will be able to do with regard to the learning objectives, and the type of exercises you should be able to solve.

**LO1: knowing about the learning objectives** Knowing about what you learn can be important. It allows you to self-reflect on what and how you learn, check your knowledge, and orient your studies towards concrete goals. For more information, see `https://educationendowmentfoundation.org.uk/education-evidence/teaching-learning-toolkit/metacognition-and-self-regulation`.

**LO2: being able to give examples of signs in the world around you** The obvious example of sign, or symbols, around you are the signs in the street themselves. Each of them has a certain meaning that tell you how to behave while driving, what to pay attention to, etc.

Another less obvious is the language that we use: words, phonemes, gestures and expressions in the case of sign language, act as symbols to carry meanings that we share.

**LO2': knowing the etymology of the word 'symbol' and relating it to its modern use** See Section 2 for the Greek etymology, and the modern use of the word.

**LO3: explaining some ambiguities in natural language or robotics and explaining how logic can solve them** In natural language, some ambiguities stem from ambiguous syntax:

> 'The robot sees the researcher with her camera.'

Whose camera is it? Is it the robot's or the researcher's?

Some other ambiguities stem from problems of interpretation or *variable grounding*. Gerbert asks a question to his brazen head about Jerusalem: the symbol 'Jerusalem', to him, refers exclusively to the city in the Middle East. However, to the brazen head, the symbol 'Jerusalem' also refers to the church in Rome that people call 'Jerusalem', hence the qui-pro-quo, the communication problem, that leads to Gerbert falling ill - despite Gerbert's reasoning being correct.

**LO3': giving at least two motivations for the use of logic** Communicating, eliminating ambiguities, studying formal reasoning, automatising all that.

**LO4: recognising and building a well-formed formula in PL** A 'well-formed' formula is an expression in PL that follows the rules of construction given in Section 4. One result that is not trivial but outside of the scope of this course is that if an expression does *not* follow these rules, then it is not an expression in PL. In other words, the rules allow you to build exactly all the formulas in PL (nothing more, nothing less).

**LO4': building a formula to model a real problem** This is a complicated question. Modelling needs to be close enough to the real world so that we can explain and predict the real world by simulating it with the model, the model acting as a proxy for the real world, but it needs to be simple enough so that we can get meaningful results. For example, if a robot driving on a road has a representation of the world that allows it to accurately recognise shapes, but if it takes, for every photo of the situation, 20 seconds to compute and recognise arriving cars and take a decision, this robot is basically useless!

In general, in propositional logic, you can use propositional variables to model the state of the components in the situation you want to model. For example, `can_wash` indicates that something can be washed (or not), such as the hands of Amavia's son in the *Faerie Queene*. You can also use formulas with implication to express general knowledge about the situation. For example, Gerbert is both a philosopher and a Pope: $\texttt{Gerbert} \Rightarrow \texttt{philosopher} \wedge \texttt{Pope}$.

This is not the only option to model problems! In Appendix B, I give examples of modelling various situations in propositional logic.

As a general guideline, try to use explicit variable names, i.e., `Gerbert`, `Amavia`, `clean_hands` instead of `G`, `A`, `c_h`.

**LO4": reading a formula and translating it in natural language** As we have seen, there can be several understandings of a formula. In general, explaining what the formula means is enough to solve ambiguities. This is a loose exercise, mainly to test your understanding of the logical connectives. For example, you might be asked to provide an explanation for the formula

$$\texttt{Amavia} \wedge \texttt{virtuous} \wedge \texttt{dead},$$

or the formula

$$\texttt{robot} \wedge \neg\texttt{can\_move},$$

which requires knowing what the connectives $\wedge$ and $\neg$ mean; refer to Table 3.

**LO5: determining if a formula is valid/satisfiable or not by building its truth table** Recall that a formula is satisfiable if there exist a variable assignment or, in other words, an interpretation that makes it true. You can follow this recipe, that I apply on the formula

$$f := (\neg\texttt{can\_wash} \wedge (\texttt{can\_wash} \Leftrightarrow \texttt{clean\_hands})) \Rightarrow \neg\texttt{clean\_hands}.$$

This formula describes Guyon's attempts at cleaning the hands of Amavia's son: it is not possible to wash the hands (expressed with $\neg$`can_wash`) and the hands are clean if and only if Guyon can wash them (expressed with ($\texttt{can\_wash} \Leftrightarrow \texttt{clean\_hands}$)). We then want to know if this knowledge implies that the hands are never clean (right part of the implication).

- Determine what the propositional variables are in your formula. In this case, the propositional variables in $f$ are `can_wash` and `clean_hands`.
- Build a truth table that contains all the possible combinations of values that your variables can take. With $n$ variables, you will need $2^n$ rows (can you figure out why?).
- Little by little, starting wherever you can, apply the rules of Table 4 to compute the values for each row of the truth table. In this case, start with determining the truth table of every propositional variables in the formula, then of ($\texttt{can\_wash} \Leftrightarrow \texttt{clean\_hands}$), then of $\neg$`can_wash`, then $\neg$`clean_hands`, then of $\neg\texttt{can\_wash} \wedge (\texttt{can\_wash} \Leftrightarrow \texttt{clean\_hands})$, and finally of $f$. I summarise this process in Figure 1.

(You can also just build a table as in Table 4).

As you can see, the truth table of $f$ is $TTTT$: no matter the variable assignment, $f$ is always true. It is a *valid* formula, also called a *tautology*. What can you conclude about the situation that the formula models?

| (¬ | can_wash | ∧ | (can_wash | ⇔ | clean_hands)) | ⇒ | ¬ | clean_hands |
|---|---|---|---|---|---|---|---|---|
| | $T$ | | $T$ | | $T$ | | | $T$ |
| | $T$ | | $T$ | | $F$ | | | $F$ |
| | $F$ | | $F$ | | $T$ | | | $T$ |
| | $F$ | | $F$ | | $F$ | | | $F$ |

| (¬ | can_wash | ∧ | (can_wash | ⇔ | clean_hands)) | ⇒ | ¬ | clean_hands |
|---|---|---|---|---|---|---|---|---|
| $F$ | $T$ | | $T$ | $T$ | $T$ | | $F$ | $T$ |
| $F$ | $T$ | | $T$ | $F$ | $F$ | | $T$ | $F$ |
| $T$ | $F$ | | $F$ | $F$ | $T$ | | $F$ | $T$ |
| $T$ | $F$ | | $F$ | $T$ | $F$ | | $T$ | $F$ |

| (¬ | can_wash | ∧ | (can_wash | ⇔ | clean_hands)) | ⇒ | ¬ | clean_hands |
|---|---|---|---|---|---|---|---|---|
| $F$ | $T$ | $F$ | $T$ | $T$ | $T$ | | $F$ | $T$ |
| $F$ | $T$ | $F$ | $T$ | $F$ | $F$ | | $T$ | $F$ |
| $T$ | $F$ | $F$ | $F$ | $F$ | $T$ | | $F$ | $T$ |
| $T$ | $F$ | $T$ | $F$ | $T$ | $F$ | | $T$ | $F$ |

| (¬ | can_wash | ∧ | (can_wash | ⇔ | clean_hands)) | ⇒ | ¬ | clean_hands |
|---|---|---|---|---|---|---|---|---|
| $F$ | $T$ | $F$ | $T$ | $T$ | $T$ | $T$ | $F$ | $T$ |
| $F$ | $T$ | $F$ | $T$ | $F$ | $F$ | $T$ | $T$ | $F$ |
| $T$ | $F$ | $F$ | $F$ | $F$ | $T$ | $T$ | $F$ | $T$ |
| $T$ | $F$ | $T$ | $F$ | $T$ | $F$ | $T$ | $T$ | $F$ |

The result!

Figure 1: Building a truth table step by step.

| A | B | ¬A | A ∧ B | A ∨ B |
|---|---|---|---|---|
| T | T | F | T | T |
| U | T | U | U | T |
| F | T | T | F | T |
| T | U | F | U | T |
| U | U | U | U | U |
| F | U | T | F | U |
| T | F | F | F | T |
| U | F | U | F | U |
| F | F | T | F | F |

Table 5: Definition of some connectives in Kleen and Priest logics.

**LO6: knowing about other types of truth values** In the digital, binary world, there are two truth values: True and False, 1 and 0 (in circuit logic), top and bottom (in algebra, sometimes), max and min (in calculus), etc. In *propositional logic*, the interpretation function is defined as a function $\mathcal{I} : Var \to \{T, F\}$ that assigns a truth value, either T or F, to every propositional variable in $Var$.

In *multi-valued logic*, it is possible to define interpretation functions with values in other sets or intervals. In *ternary logic* for example, $\mathcal{I} : Var \to \{T, U, F\}$ with $U$ a symbol that might stand for 'unknown' or 'don't care'. In *fuzzy logic* and *probabilistic logic*, you can even find $\mathcal{I} : Var \to [0, 1]$ directly in the continuous interval $[0, 1]$!

Of course, changing the interpretation function requires giving alternative definitions of the connectives, and the truth tables are a bit more complicated as a result. For example, in Kleene and Priest logics, the negation, conjunction, and disjunction are defined with the truth tables given in Table 5.

**LO6': knowing about the limits of PL and what other logics to use for what purposes** Sometimes, modelling real life situations with PL requires a huge number of variables, that makes it difficult to handle even with a computer. *First-order logic* is another logic that solves some of these issues, but at the cost of being so powerful that it may make reasoning impossible. A compromise between the two is, for example, the family of logics known as *description logics*.

# B Some examples of modelling a situation in propositional logic

In this section, I give some examples of propositional logic formulas that correspond to various situations inspired from literature. Note the conceptual differences between the elements I describe. Sometimes, I express knowledge about *specific* elements or actors in the situation, as I have done earlier with Gerbert or with Guyon; sometimes, I express *general* knowledge about the world, general rules, such as the fact that Jerusalem is a city in the Middle East (or is it?), usually with an implication like so:

$$\texttt{Jerusalem} \Rightarrow \texttt{city} \land \texttt{in\_the\_Middle\_East}.$$

## B.1 Asimov's three laws of robotics

Isaac Asimov (1920 – 1992) was Russian-born American writer and professor of biochemistry. He wrote a large amount of science-fiction stories[4] and popular science books, which had a certain influence over robotics (the term of which he coined) and artificial intelligence in general[5]. In particular, he explored the interactions between robots and humans in a non-conflictual manner, with robots that are not enemies and whose failures arise from logical errors instead of malevolent intent, as was common in American science-fiction at the time. In his short story *Runaround* (1942), he introduces a set of rules known as the Three Laws of Robotics which inspired the rest of his work and others' on robotics. The rules are as follows[6].

First Law: A robot may not injure a human being or, through inaction, allow a human being to come to harm.

Second Law: A robot must obey orders given it by human beings except where such orders would conflict with the First Law.

[4] See this extensive list at https://en.wikipedia.org/wiki/Isaac_Asimov_short_stories_bibliography!

[5] See, for example, https://www.latimes.com/archives/la-xpm-1992-04-08-vw-636-story.html; Marvin Minsky called himself a 'robot psychologist' in reference to Asimov's works.

[6] https://webhome.auburn.edu/~vestmon/robotics.html

Third Law: A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Most robots are supposed to follow these rules to a tie. Conflicts arise when they cannot, for various reasons.

As I mentioned in the introduction of the current section, and because these are very general rules that apply to all robots and many different situations, as opposed to a *specific* robot, I will use implications to express them.

**First Law** Note the importance of the parenthesis to indicate that the negation applies to the whole subformula that expresses the idea of something 'that injures humans or allows a human to come to harm'. Note also that I have not expressed the idea of 'inaction'. Do you think this is important? How would you have expressed it?

$$\texttt{first\_law} := \texttt{robot} \Rightarrow \neg(\texttt{injure\_human} \vee \texttt{allow\_human\_to\_come\_to\_harm})$$

**Second Law** Let the symbol `first_law` stand for the formula above.

$$\texttt{second\_law} := \texttt{robot} \Rightarrow \texttt{obey\_human\_orders} \wedge \texttt{first\_law}$$

**Third Law**

$$\texttt{third\_law} := \texttt{robot} \Rightarrow \texttt{protect\_own\_existence} \wedge \texttt{first\_law} \wedge \texttt{second\_law}$$

## B.2 Talos, the brazen man

In his epic poem *Argonautica*, Apollonius of Rhodes (first half of 3rd century BC) relates the story of the hero Jason and the Argonauts who set out to retrieve the mythical Golden Fleece from the remote Colchis (actual Georgia). Fleeing the Colchians after they stole the Fleece, they sail aboard their ship, the Argo, towards the island of Crete. The island is protected by a giant made of bronze, Talos[7].

> From that point they [the Argonauts] were to cross to Krete (Crete), the greatest island in the sea. But when they sought shelter in the haven of Dikte (Dicte) they were prevented from making fast to the shore by Talos, a bronze giant, who broke off lumps of rock from the cliff to hurl at them. A descendant of the brazen race that sprang from ash-trees, he had survived into the days of the demigods, and Zeus had given him to Europa to keep watch over Krete by running round the island on his bronze feet three times a day. His body and his limbs were brazen and invulnerable, except at one point: under a sinew by his ankle there was a blood-red vein protected only by a thin skin which to him meant life or death. He terrified the Argonauts, and exhausted though they were they hastily backed water. Indeed, what with thirst and other pains, they would have been driven away from Krete in a sorry frame of mind, but for Medeia (Medea), who stopped them as they turned the ship about. 'Listen to me,' she said. 'I think that I and I alone can get the better of that man, whoever he may be, unless there is immortal life in that bronze body. All I ask of you is to stay here keeping the ship out of range of his rocks till I have brought him down.'
>
> Apollonius Rhodius, Argonautica 4. 1638 ff (trans. Rieu) (Greek epic C3rd B.C.)

In this case, I will describe the specific properties of a specific character, Talos. I will first write sentences in natural language, and second I will propose a translation of each sentence in propositional logic. I will use all connectives $\{\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow, (, )\}$ to showcase their use.

- Talos is a brazen man.

  $$\texttt{Talos} \Rightarrow \texttt{brazen\_man}.$$

  Something to think about: what if I had used $\Leftrightarrow$ instead of $\Rightarrow$? Would that have been a good model for the situation? Why?

- A brazen man is a giant made of bronze.

  $$\texttt{brazen\_mans} \Leftrightarrow \texttt{giant} \wedge \texttt{made\_of\_bronze}.$$

  Note that this is a general rule about brazen men, not just Talos.

- Talos is either running around the island or hurling lumps of rocks at intruders.

  $$\texttt{Talos} \Rightarrow \texttt{running\_around} \vee \texttt{hurling\_lumps\_of\_rocks\_at\_intruders}.$$

[7] See also https://www.theoi.com/Gigante/GiganteTalos.html.

- Talos cannot be killed.

$$\texttt{Talos} \Rightarrow \neg\texttt{mortal}.$$