Contents

1 Logic, language, thought

- Be advised not to take logic as a model of human thought and reasoning.
- Thought and reasoning are fairly complicated mental processes that do not seem to behave as dictated – at least – by standard logics, and are the subject of psychology, linguistics, artificial intelligence and related fields.
- Laws of reasoning in logic, with some provisos, may be argued to apply to the end-products of thought, idealized as propositions, predicates, quantification, and so on, which are expressed in a formal language. Logical laws can at best be utilized in justifying these end-products, rather than characterizing how they are discovered or arise in the mind/brain of the thinker.[1]
- This distinction between discovery and justification starts to apply already in mathematics before even approaching "everyday" thinking. Theorems are *discovered* via largely intuitive (read as "not scientifically explained yet") means, they are *justified* (checked for validity) by formal tools of logic.[2]

2 Syntax of Propositional Logic

- Any specification of a language starts with its alphabet.

  Our alphabet for L_0 – the name we will give to our language – is made up of three sets:

  Basic symbols:

$$P = \{p, q, r, s, p_1, q_1, r_1, s_1, \ldots\} \tag{1}$$

[1] See the introduction to Reichenbach (1947) for some relevant discussion. The classical work dealing mainly with the distinction between human thought and inference on one hand and formal logic on the other is Johnson-Laird (1983)

[2] Although the book itself is advanced for introductory level, (Quine, 1958, §16) discusses this issue.

Connective symbols:

$$C_1 = \{-\} \tag{2}$$
$$C_2 = \{\wedge, \vee, \rightarrow\} \tag{3}$$

Parentheses: $\{(,)\}$

- We can also collect the parts of our alphabet under a single set:

$$\Sigma = P \cup C_1 \cup C_2 \cup \{(,)\} \tag{4}$$

- Let's call an **expression**, any finite sequence of (possibly repeated) elements from Σ. The following are some example expressions:

$$p(\wedge r_{4291841} \vee - \qquad p-p \qquad -)) \qquad \wedge\vee\wedge \qquad q_{345}\wedge$$

- It is not hard to see that there is no bound to the expressions we can form this way. Call this non-finite set of all the possible finite expressions formed by putting together a selection of symbols from Σ in a specific order Σ^*.
- Usually, we are interested in expressions fulfilling certain criteria – a subset of the set of all possible expressions. We distinguish these special expressions as the grammatical sentences of the language we are interested in.
- Now we can define the grammatical expressions (sentences or **well-formed formulas**) of our language L_0. And we will see that $L_0 \subset \Sigma^*$.
- As you might have already realized, we identify a language with the set of its grammatical sentences.

Definition 2.1
Well-formed formulas of L_0.

i. $\alpha \in L_0$, if $\alpha \in P$.
ii. if $\alpha \in L_0$, so is $(\gamma\alpha)$, for $\gamma \in C_1$.
iii. if $\alpha, \beta \in L_0$, then so is $(\alpha\gamma\beta)$, for $\gamma \in C_2$.
iv. Nothing else is in L_0.

- An analytic procedure for defining the notion "well-formed formula of L_0". We start from a complex expression and go down.

Definition 2.2 (wff's of L_0, "top-down")

i. α is a wff if $\alpha \in P$.
ii. $(-\alpha)$ is a wff iff α is a wff.
iii. $(\alpha \wedge \beta)$ is a wff iff α and β are wff's.
iv. $(\alpha \vee \beta)$ is a wff iff α and β are wff's.
v. Any expression that falls out of the above is not a wff.

- Here is a synthetic way of specifying well-formed formulas. We start with the simplest expressions and go up.

Definition 2.3 (L_0, "bottom-up")
An expression is a wff (or belong to L_0) iff it is built from the elements of P by applying a finite number of the following operations:[3]

$$f_{-}(\alpha) = (-\alpha) \tag{5}$$
$$f_{\wedge}(\alpha, \beta) = (\alpha \wedge \beta) \tag{6}$$
$$f_{\vee}(\alpha, \beta) = (\alpha \vee \beta) \tag{7}$$

---

[3] What is the difference between the parentheses on the left and right side of the equalities?

- This could be stated more formally:

Definition 2.4 (L_0, “bottom-up”)
An expression α is a wff iff there exists a sequence of expressions ordered in increasing length:

$$\alpha_1, \alpha_2, \ldots, \alpha_n$$

such that $\alpha = \alpha_n$, and for any α_i for $i \leq n$, either

i. $\alpha_i \in P$;
ii. or there exists $j,k < i$ such that $\alpha_i = f_\wedge(\alpha_j, \alpha_k)$ or $\alpha_i = f_\vee(\alpha_j, \alpha_k)$
iii. or there exists a $j < i$ such that $\alpha_i = f_-(\alpha_j)$,
where f_-, $f_\wedge$, and $f_\vee$ are as defined in Definition 2.3.

- All these definitions not only define what the wff’s of L_0 are, but also their structure.

- This is best observed over **derivation** (or construction) trees.

  Example 2.5 (Derivation trees)
  Draw the derivation tree of $((p \wedge q) \vee ((-r) \vee (q \wedge (-s))))$

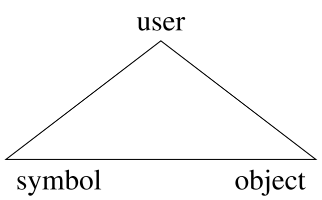
3 Semantics of Propositional Logic

- In the previous section, we established the first half of a formal system. We know which sequences of symbols constitute a well-formed expression of our system. However, we do not yet know what those well-formed expressions mean.

3.1 Symbols

- Signs and signification are central concepts in language, logic, and computation.

- An initial and rough conception of signification is a three-part relation:[4]



- Again some initial and rough characterizations:
  - A user uses a sign to **refer** to an object.
  - A sign **denotes** an object.
  - A user has certain **intentions** about an object.

3.2 Propositions

- Let’s characterize what a **proposition** is, indirectly by way of natural language. Our characterization has two parts:

  i. Any expression that you can insert into the contexts “___ is true” or “___ is false” *denotes* a proposition.[5]
  ii. There are no propositions that cannot be denoted as such.

  The following expressions denote propositions:

  The earth revolves around the sun.

  Dünya Güneş etrafında dönüyor.

  John likes Mary.

[4] If you are interested in the study of signs, a good place to start reading is the founders of the science of signs – named **semiotics** or **semiology**: Ferdinand de Saussure’s “Course in General Linguistics” and Charles Sanders Peirce’ (pronounced like the word *purse*) “Collected Papers”. Be warned that Peirce might be quite challenging for beginners, but Saussure is a must read for everyone with a serious interest in language.

[5] The contexts “It is the case that ___” or “It is not the case that ___” will equally do, barring some complications due to quotation. It gets yet more complicated when it comes to Turkish. Can you see why?

If you multiply an even number with an odd number, you obtain an even number.

while the following do not:

Around the sun

Dünya Güneş etrafında dönüyor mu?

Because John likes Mary

If you multiply an even number with an odd number

- In other words, **declarative sentences** denote propositions.
- Now let's assume that every proposition whatsoever has at least one declarative sentence to express it.
- From all we have above it follows that:

  Propositions are objects that can be (said to be) true or false.
- Another way of saying this is that propositions are objects that have **truth values**.

3.3 Propositional symbols

- Propositional logic is called so, because its **atomic**[6] symbols refer to objects called **propositions**.
- For instance, we can agree on things like, p stands for the proposition that the snow is white. Given this, we can see that p is interchangeable with declarative sentences that express this proposition. Therefore, we can say

  p is interchangeable with 'The snow is white.'

  p is interchangeable with 'Kar beyazdır.'

  and, so on.
- Sometimes you will see 'means', 'abbreviates', 'stands for', 'equals to', '=', and so on, in place of 'is interchangeable with'.
- When we use a basic symbol of L_0 as part of a well-formed formula of L_0, which possibly consists of only that symbol, we assert the proposition referred to by the symbol. So,

$$p \tag{8}$$

  says that the snow is white.
- In science and other discourses, we are not only interested in what propositions are expressed, but also whether the expressed propositions actually hold or not.
- For any proposition whether it holds or not is indicated by its **truth value**.[7] If a proposition holds, we will say that it is true, and its truth value is 1. If a proposition does not hold, we will say that it is false, and its truth value is 0. Using the first two natural numbers to designate truth values is totally arbitrary, you can pick any two symbols which will not lead to confusion.
- We will assume that, every proposition expressible in our system is either true or false, there is no case in between.
- A wff consisting only of a basic symbol of L_0 has the same truth value as the proposition it expresses. Therefore, (8) is true, or has the truth value 1, if the snow is actually white, and is false, or has the truth value 0, if it is not the case that snow is white.
- From now on we will use "formula" in place of "well-formed formula" and directly speak of the truth or falsity of formulas of L_0.

3.4 Valuations

- A **valuation** is a function from the set of propositional symbols to $\{0,1\}$. It tells which of our basic propositions are true and which are false.

[6] The adjective *atomic* serves to suggest that atomic symbols cannot be broken down to further components. This use of *atomic* is a remnant from a pre-nuclear conception of atoms.

[7] Note that we are counting on our intuitions here for the meaning of "holding" vs. "not holding". A mathematical rendition of this concept will follow soon.

- Assume we are dealing with a small subset of the language of propositional logic, where we have only three propositional symbols $\{p,q,r\}$. Then there are only eight possible valuations:

$$\begin{aligned} V_1 &= \{(p,1),(q,1),(r,1)\} \\ V_2 &= \{(p,1),(q,1),(r,0)\} \\ V_3 &= \{(p,1),(q,0),(r,1)\} \\ V_4 &= \{(p,1),(q,0),(r,0)\} \\ V_5 &= \{(p,0),(q,1),(r,1)\} \\ V_6 &= \{(p,0),(q,1),(r,0)\} \\ V_7 &= \{(p,0),(q,0),(r,1)\} \\ V_8 &= \{(p,0),(q,0),(r,0)\} \end{aligned} \tag{9}$$

- These eight functions can be pictured also as a table:

(10)

p	q	r
1	1	1
1	1	0
1	0	1
1	0	0
0	1	1
0	1	0
0	0	1
0	0	0

- You can think of each function in (9) and, equivalently, each row in (10) designating a state of the world we are interested in. For instance if you need to represent certain properties of a room for some reason and the propositional symbols are associated with the following propositions,

p : lights are on
q : the door is locked (11)
r : the room is occupied

each valuation corresponds to a certain state of the room. States are mutually exclusive, since at a given time the room is in exactly one of the possible states.

3.5 Interpretation

- To systematically characterize the meanings of formulas of propositional logic, we will make use of an interpretation function $\mathscr{I}$ that takes a valuation and a formula as input and gives a truth-value as output.
- For atomic formulas, those comprising of a single propositional symbol, defining the function is straightforward.

Definition 3.1
For any formula P and a valuation V,

$$\mathscr{I}(P,V) = V(P), \text{ if } P \text{ is atomic.}$$

□

- Given that V is a function from propositional symbols to truth values and P is an atomic formula comprising of a single propositional symbol, $V(P)$ gives you the truth-value of the expressed proposition according to the given valuation.

3.6 Truth-functional connectives

3.6.1 Conjunction

- As we saw above in defining the syntax of propositional logic, one way to build complex formulas out of simple ones is **conjunction**. On the meaning side, a conjunction asserts that both conjuncts are true. If at least one of the conjuncts fails to be true, then the conjunction fails to be true as well. If p stands for snow's being white and q stands for Berlin being the capital of France,

$$(p \wedge q) \tag{12}$$

is true if and only if the snow is white and Berlin is the capital of France.

- We can extend the definition of the interpretation function $\mathscr{I}$ as follows: For any formulas P and Q and valuation V:

Definition 3.2

$$\begin{aligned}\mathscr{I}((P \wedge Q), V) &= 1 \text{ if both } \mathscr{I}(P,V) = 1 \text{ and } \mathscr{I}(Q,V) = 1\\ &= 0, \text{ otherwise}\end{aligned}$$

□

- Note that the interpretation function $\mathscr{I}$ again takes two arguments, a formula and a valuation, but this time we want the first argument to be a conjunction.
- It is also possible to take the conjunction itself as a function that maps a pair of truth values to a truth value. The input pair is the truth-values of the left and the right conjuncts, respectively. Such functions are called **truth-functions**, as the value of the function depends entirely on the input truth-values. The customary way to express truth-functions is a **truth-table**:

P	Q	$P \wedge Q$
1	1	1
1	0	0
0	1	0
0	0	0

(13)

- A more compact way to express the same function is as follows, where the first column gives the values for the left conjunct and the first row gives the values for the right conjunct:

$\wedge$	1	0
1	1	0
0	0	0

(14)

Example 3.3
Given the formula $(p \wedge q)$ and the valuation $\{(p,1),(q,0)\}$,

$$\mathscr{I}((p \wedge q), \{(p,1),(q,0)\}) = 0 \tag{15}$$

since $\mathscr{I}(p,\{(p,1),(q,0)\}) = 1$, but $\mathscr{I}(q,\{(p,1),(q,0)\}) = 0$.

□

- As you might have noticed, in characterizing the interpretation function $\mathscr{I}$ we are using capital symbols like P instead of p. There is a reason for this. The lower-case symbols like p, q and so on are part of our alphabet and are used to refer to atomic propositions. On the other hand the upper-case symbols like P are used and will be used to refer to arbitrary formulas. An arbitrary formula can be of any complexity as well as being atomic. If you go back to (3.1) you will see that we started with P but then further restricted what P can stand for by the expression 'If P is atomic'.

Example 3.4
This time we are given a more complex formula $((p \wedge q) \wedge r)$ and the valuation $V = \{(p,1),(q,0),(r,1)\}$,

As our formula to be interpreted is a conjunction at the top level, we need to use the rule for conjunction given in (3.2). What we need to compute is the value of

$$\mathscr{I}(((p \wedge q) \wedge r), V) \tag{16}$$

which dictates that

$$\mathscr{I}((p \wedge q), V) \tag{17}$$

and

$$\mathscr{I}(r, V) \tag{18}$$

both be 1.

Further computations will reveal that while (18) is evaluated to 1, (17) is evaluated to 0; therefore the value of (16) is 0.

□

- As examples (3.3) and (3.4) show, in computing the truth-value of a formula we need to use the notions 'if', 'both', and 'and' borrowed from natural language. It is possible and desirable to define the interpretation function fully in mathematical form. Here is one way to do it:

Definition 3.5

$$\mathscr{I}((P \wedge Q),V) = \mathscr{I}(P,V) \times \mathscr{I}(Q,V)$$

□

- With definitions (3.5) and (3.1) at hand, we can express the computation involved in example (3.4) as follows:

$$\begin{aligned} \mathscr{I}(((p \wedge q) \wedge r),V) &= \mathscr{I}((p \wedge q),V) \times \mathscr{I}(r,V) && (19) \\ &= \mathscr{I}(p,V) \times \mathscr{I}(q,V) \times \mathscr{I}(r,V) \\ &= 1 \times 0 \times 1 \\ &= 0 && (20) \end{aligned}$$

3.6.2 Disjunction

- Another binary connective is **disjunction** (or **alternation**). It differs from conjunction in being more tolerant about falsehood. The disjunction of two sentences come out true if at least one of the disjuncts is true. With the same propositions being expressed by p and q as above,

$$(p \vee q) \tag{21}$$

is true if and only if the snow is white, or Berlin is the capital of France, *or both*.

- Here is the truth-table for disjunction:

P	Q	$P \vee Q$
1	1	1
1	0	1
0	1	1
0	0	0

(22)

and disjunction as a truth-function:

$\vee$	1	0
1	1	1
0	1	0

(23)

- We add disjunction to the interpretation function as follows:

$$\mathscr{I}((P \vee Q),V) = \mathscr{I}(P,V) + \mathscr{I}(Q,V) - \mathscr{I}(P,V) \times \mathscr{I}(Q,V) \tag{24}$$

3.6.3 Negation

- Negation is a device that flips the truth-value of whichever formula it is adjoined to.[8]

- The interpretation function $\mathscr{I}$ handles negation as follows: For any formula P and valuation V,

$$\mathscr{I}((-P),V) = |\mathscr{I}(P,V) - 1| \tag{25}$$

[8] Adjoining '$-$' and P yields '$(-P)$' according to the syntax we defined for L_0.

- The truth-table is,

P	$-P$
1	0
0	1

(26)

- Although negation does not connect any formulas to each other, it is established custom to call it a unary connective. Negation denotes the truth function,

$-$	
1	0
0	1

(27)

3.6.4 Conditional

- Under what circumstances would you consider the person who uttered the following sentence to have kept her word?

  "If I get a job next summer, then I will marry you."

- The connective '$\rightarrow$', named **conditional** (or **material conditional**) is read "if...then", or "only if" and has the following truth-table:

P	Q	$P \rightarrow Q$
1	1	1
1	0	0
0	1	1
0	0	1

- The conditional denotes the truth-function,

$\rightarrow$	1	0
1	1	0
0	1	1

(28)

- We extend the interpretation function to cover the conditional:

$$\mathscr{I}((P \rightarrow Q), V) = (\mathscr{I}(P, V) - \mathscr{I}(Q, V) - 1)^2 \bmod 3 \tag{29}$$

3.6.5 Biconditional

- The connective '$\leftrightarrow$', named **biconditional**. It is read "if and only if" and comes out true whenever the two components agree in their truth value.

- The truth-table, truth-function and the extension of the interpretation function for biconditional is left as an exercise.

4 Connective precedence and grouping

- Both connectives of this section bind less tightly then conjunction and alternation. Therefore '$p \vee q \rightarrow r$' is '$((p \vee q) \rightarrow r)$'; you can simplify '$(p \vee (q \rightarrow r))$' at most as '$p \vee (q \rightarrow r)$'.

- Our definition of L_0 requires the insertion of parenthesis in every step of connection, thereby making every possible sentence unambiguous with regards to what is grouped with what. However, for ease of inspection and writing, we will omit parentheses according to certain conventions.

- Outermost parentheses are optional.

- Conjunction and disjunction are associative, therefore $(P \wedge (Q \wedge R))$, $((P \wedge Q) \wedge R)$ and $P \wedge Q \wedge R$ are all equivalent.

- Negation binds most tightly, $-P \wedge Q$ is different from $-(P \wedge Q)$ and likewise for other connectives wrt negation. Conjunction and disjunction bind more tightly than conditional and biconditional: $P \wedge Q \rightarrow R$ is different from $P \wedge (Q \rightarrow R)$.

5 Truth-tree method

$\square$

- The interpretation function allows one to compute the truth value of a formula with respect to a given valuation.
- Some terminology: Given a valuation V and a formula F, we say that 'V **satisfies** F' iff $\mathscr{I}(F,V) = 1$.
- Sometimes one is interested in computing the *set* of valuations that satisfy a formula. From this perspective, every formula imposes a filter on valuations; it picks a subset of the set of all possible valuations. This subset is the set of valuations that satisfy the formula in question.
- One straightforward method to compute the valuations that satisfy a formula is to construct a truth-table for the entire formula, as we have been doing for connectives.

Example 5.1
To find the valuations that satisfy,

$$p \to q \vee -r \tag{30}$$

we construct a truth-table:

p	q	r	$-r$	$q \vee -r$	$p \to q \vee -r$
1	1	1	0	1	1
1	1	0	1	1	1
1	0	1	0	0	0
1	0	0	1	1	1
0	1	1	0	1	1
0	1	0	1	1	1
0	0	1	0	0	1
0	0	0	1	1	1

which shows that except the third valuation, namely $\{(p,1),(q,0),(r,1)\}$, all the valuations satisfy the formula $p \to q \vee -r$.

Truth-table method is a definitive method to find the valuations that satisfy a certain formula. However it has some redundant information. For instance, in (30), any valuation that maps p to 0 is guaranteed to satisfy the formula, given the semantics of the conditional. In the truth-table method you nevertheless fill the rows that start with a p value of 0 (the last four rows).

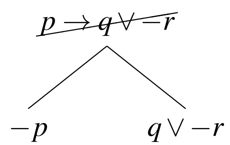
- The method of **truth-trees** (or **semantic tableau**) offers a faster discovery procedure.
- The method consists of breaking formulas into atomic propositional symbols or their negations. Let us illustrate the procedure over (30). In the truth-tree method you break a formula from its main connective into cases consisting of sub-formulas that are required to hold for the decomposed formula to hold.

Example 5.2
Let us apply the truth-tree method to the formula $p \to q \vee -r$. We start by writing the formula under investigation:

$$p \to q \vee -r$$

In each step of decomposition we are concerned with the main connective of the formula to be decomposed. Decomposition involves enumerating structurally simpler formulas whose truth guarantees the truth of the formula under investigation. Our ultimate aim is to leave nothing that is more complex than a negation of a propositional symbol.

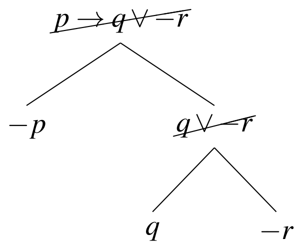
As we have a conditional, we decompose our formula as follows:



(31)

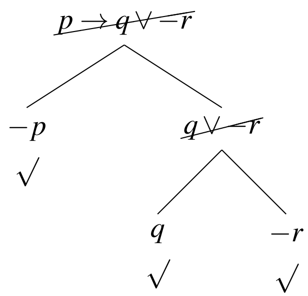
given that a conditional of the form $P \to Q$ is true whenever P is false or Q is true. When we are done with decomposing a formula we cross it out to avoid confusion.

In the next step, we decompose the disjunction on the right branch, how to decompose a disjunction should not be hard to see:



(32)

We stop when we have only propositional symbols (or variables) and their negations on the leaves of the tree. We check all the paths in the tree for contradictions. A path has contradiction if there is both P and $-P$ on it for some formula P. We mark a path with a contradiction with '$\times$' and that path is said to be **closed**. The paths without contradiction are marked with '$\surd$'. In our present case no path is closed:



(33)

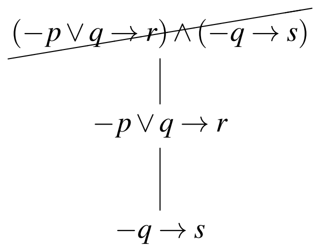
The tree in (33) contains all the information present in the truth-table (30) in a more concise way. Each non-contradictory path corresponds to a set of rows in the truth-table. For instance the left most path which has $-p$ stands for all the four rows in the table which has a 0 for p, and shows that in all these rows the main formula has the truth-value 1.

□

Example 5.3
Now let us have a more complicated example:

$$(-p \vee q \to r) \wedge (-q \to s)$$

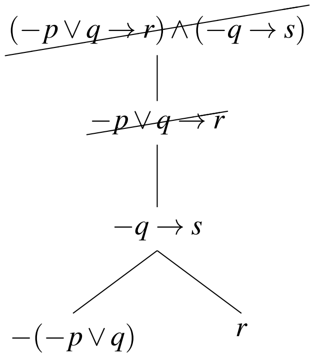
For a conjunction to hold both conjuncts must hold, licensing the following decomposition:



(34)

We do not branch as both conjuncts should hold at the same time, they are not alternatives as was the case in disjunction and conditional.

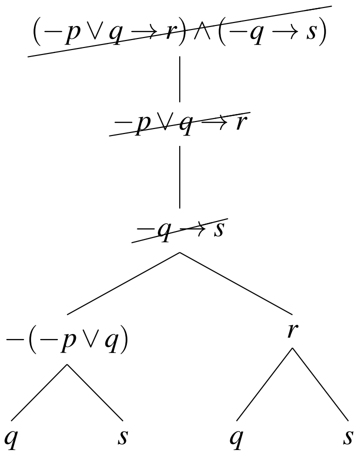
We always continue decomposing from the left- and topmost decomposable formula. In the present case we have a conditional:



(35)

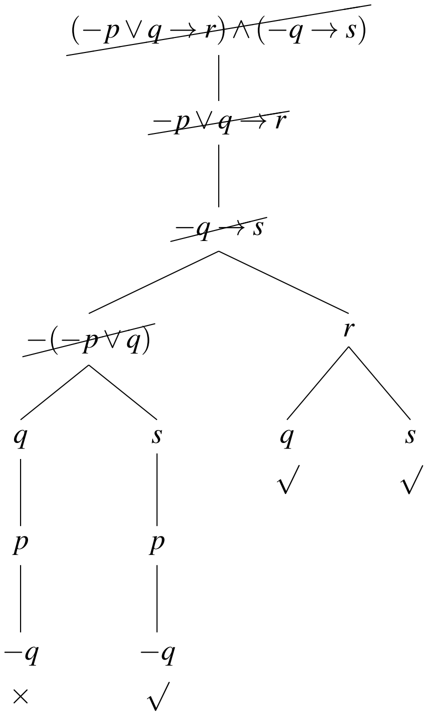
The left- and topmost decomposable formula is $-q \to s$, we continue from there. This decomposition leads to branching with $--q$ and s. We will always cancel double negations without showing it as a separate step. Therefore, the branches will be q and s.

At this point we need to be careful in applying the branching resulting from this decomposition to all the nodes that are reachable from the decomposed formula by going only down. The result is:



(36)

One more formula is left to be decomposed, which is the negation of a disjunction. A disjunction is false only when both disjuncts are false. Therefore decomposition proceeds as follows:

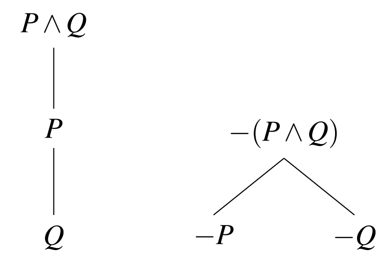


(37)

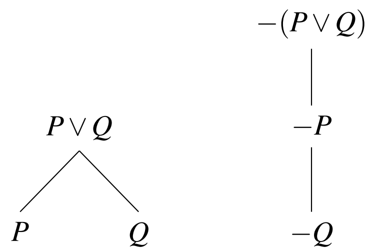
□

- Now it is time to list decomposition rules for each connective and their negation:

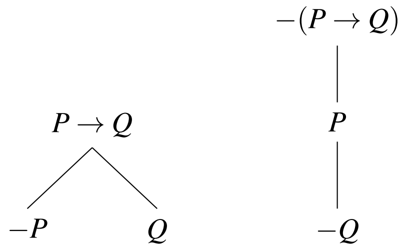
Truth-tree construction rules:



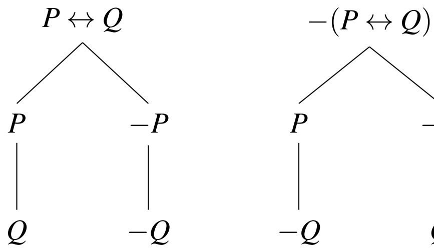
(38)



(39)



(40)



(41)

Exercise 5.4
Construct truth-trees for the following formulas:

1. $(p \to q) \to (p \to q \vee r)$
2. $(p \to q) \to (p \to q \wedge r)$
3. $(p \wedge s) \leftrightarrow (q \vee r)$
4. $((p \to q) \to s) \wedge (p \to (q \to s))$
5. $p \wedge (-q \vee -p)$
6. $((p \to -q) \to -p) \to q$
7. $(p \to q) \to (-p \to -q)$
8. $(p \to q) \to (-q \to -p)$
9. $((p \to q) \to p) \to p$
10. $(p \vee q) \to (p \to q)$
11. $(p \to q) \wedge (q \to r) \wedge (r \to p)$
12. $(p \to q) \vee (q \to r) \vee (r \to p)$
13. $-(-(-p \wedge -q) \wedge -p) \leftrightarrow -(q \vee -p)$

□

Exercise 5.5
Inspecting the trees for the previous question, do you find it easy to discover the valuations that make the formula false? Any useful strategy for this task?

□

Exercise 5.6
You are given that Mary, a graduate student, is such that:

> If she studies, she gets good grades.
> If she doesn't study, she enjoys life.
> If she doesn't get good grades, she doesn't enjoy life.

Symbolize these statements in propositional logic and construct a truth-tree to answer the following questions:

1. Does she enjoy life?
2. Does she get good grades?

□

Exercise 5.7
We have the following information:

> If samples are contaminated, then there is leakage in the reactor or there is non-consumed fuel.
> All the fuel is consumed.
> There is no leakage in the reactor or the detector is malfunctioning.
> The detector is functioning properly.

Someone asserts that samples are contaminated. Do we need to believe? Motivate your answer by constructing a truth-tree.

□

---

6 Validity, consistency, inconsistency

- A formula P is **valid** if and only if it comes out true with every valuation.

  We designate the validity of P by:

  $$\models P$$

  Valid formulas of propositional logic are also called **tautologous**.

  Some examples:

  $$\models p \vee -p \qquad (42)$$
  $$\models p \rightarrow (q \rightarrow p)$$

- A formula P is **inconsistent** (or **contradictory**), designated $\not\models P$, if and only if it comes out false in every valuation. For instance,

  $$\not\models -(p \rightarrow q) \wedge -p \qquad (43)$$

- A formula P is **consistent** (or **satisfiable** if and only if there exists at least one valuation V such that V satisfies P.[9]

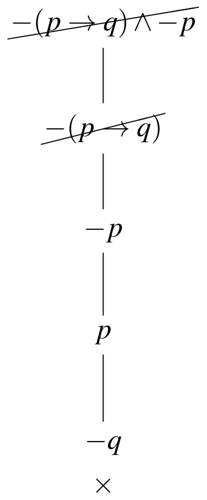
- Given a formula, one way to decide whether it is valid, consistent or inconsistent is to construct a truth-table and inspect the rows. For a valid formula, all the rows must have 1 in their rightmost cells; for an inconsistent one, they must have 0; and for a consistent formula there must be at least one row with a rightmost cell containing 1.

- Again truth-tree method proves to be a better tool for such judgments.

- An inconsistent formula has a truth-tree with all the branches closed; in other words, there is not a single way (=valuation) that the formula can be true.

  Example 6.1
  Let us show that the formula (43) is inconsistent.

[9] Note that every valid formula is consistent, but not every consistent formula is valid.



(44)

As all the branches are closed, we can conclude that (43) is inconsistent.

□

- Check for consistency is also straightforward, one unclosed branch guarantees consistency.
- When it comes to validity, however, furhter inspection might be required after the construction of the tree. For instance, construct the truth-tree for

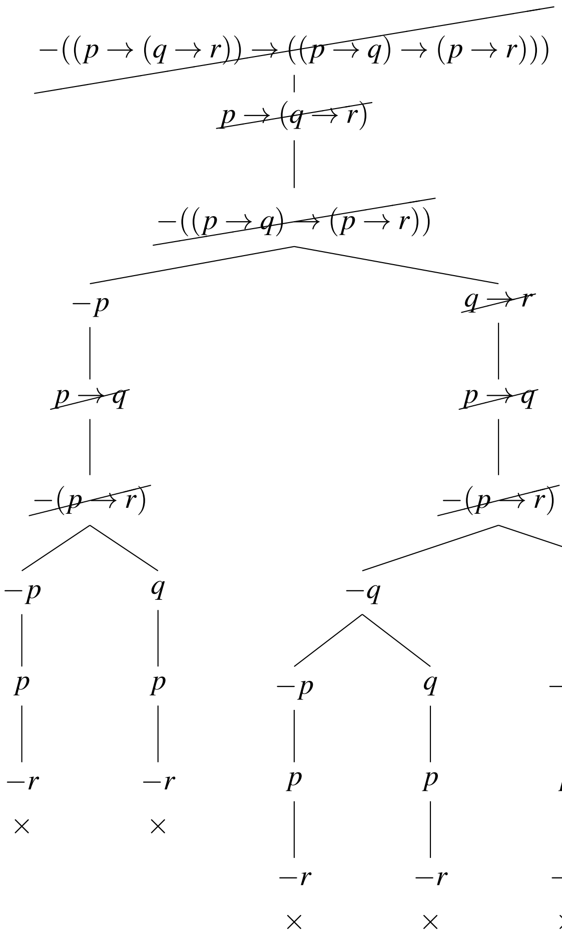
$$(p \to (q \to r)) \to ((p \to q) \to (p \to r)) \tag{45}$$

How easy is it to see whether it is valid or not?

- At this point we make use of a relation between validity and inconsistency. Given that a valid formula is true in every valuation, its denial must be false in every valuation–that is, it must be inconsistent. Therefore, whenever we aim to test a formula for validty, we simply take its negation and construct the truth-tree: If all the branches are closed, then the formula is valid, otherwise it is not valid. In the latter case, any open branch will tell show you valuation(s) that makes the formula false.

Example 6.2
Show that formula (45) is valid.

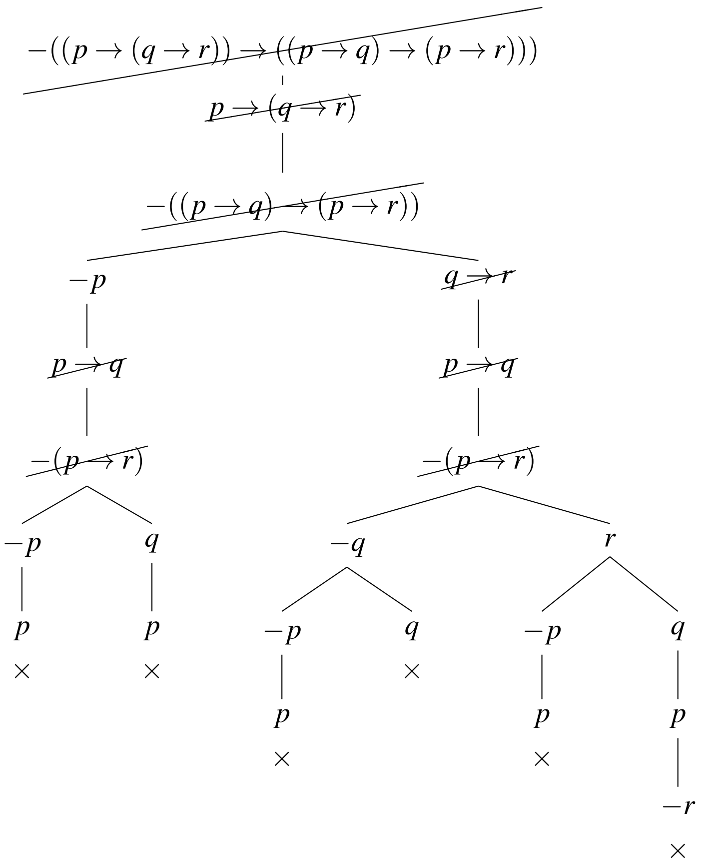
We construct the truth tree of its denial:



(46)

The tree shows that there is no valuation that would satisfy the formula, since if there had been any, we would have at least one branch that is not closed.

Notice that some of the branches in the tree can be observed to lead to a contradiction before the entire branch is completed. If the procedure we use to construct the tree checks for contradictions not only after the entire tree is constructed but after each expansion of the tree into branches, then some labor and space could be saved. Such a procedure would lead to the following tree:



(47)

□

7 Implication, equivalence and substitution

- A formula P **implies** Q, designated $P \models Q$, if and only if $\models P \to Q$.
- Two formulas P and Q are **equivalent**, designated $P \equiv Q$, if and only if $\models P \leftrightarrow Q$.
- Given a formula R, $R \equiv R_{[P/Q]}$ iff $P \equiv Q$, where $R_{[P/Q]}$ means the formula formed by substituting P for each occurrence of Q in R.

Exercise 7.1
Given '$p \to q$', state which of the following imply or are implied by it:

$-p$	q	$-p \vee q$	$q \wedge r$
	$p \to q \vee r$	$p \vee r \to q$	
$p \to q \wedge r$	$(p \to q) \vee r$	$-q \to -p$	$p \wedge r \to q$

---

8 Natural deduction

Natural deduction is a method for **deriving** a formula Q from (possibly empty) set of formulas $\{P_1, P_2, \ldots, P_n\}$, called **premisses**, such that Q is true in case all the premisses are jointly true. The relation of derivability is designated as:

$$P_1, P_2, \ldots, P_n \vdash Q$$

and,

$$P_1, P_2, \ldots, P_n \vdash Q \quad \text{iff} \quad P_1 \wedge P_2 \wedge \ldots \wedge P_n \models Q$$

The method of natural deduction involves a set of **inference rules** and **proof techniques**.

8.1 Inference rules

Simplification:

$$\frac{P \wedge Q}{P} \qquad \frac{P \wedge Q}{Q}$$

Adjunction:

$$\frac{\begin{array}{c}P\\Q\end{array}}{P \wedge Q} \qquad \frac{\begin{array}{c}P\\Q\end{array}}{Q \wedge P}$$

Addition:

$$\frac{P}{P \vee Q} \qquad \frac{Q}{P \vee Q}$$

Modus ponens (MP):

$$\frac{\begin{array}{l}P \to Q\\P\end{array}}{Q}$$

Modus tollens (MT):

$$\frac{\begin{array}{l}P \to Q\\-Q\end{array}}{-P}$$

Modus tollendo ponens (MTP):

$$\frac{\begin{array}{l}P \vee Q\\-P\end{array}}{Q} \qquad \frac{\begin{array}{l}P \vee Q\\-Q\end{array}}{P}$$

Double negation:

$$\frac{--P}{P} \qquad \frac{P}{--P}$$

Repetition:

$$\frac{P}{P}$$

8.2 Proof techniques

8.2.1 Direct proof

Let's illustrate how a direct proof works over an example,

$$p \wedge q \vdash q \wedge p$$

We start by designating our target – the formula we aim to derive:

In proofs we can pick and add any of our premisses at any point, if we believe it will be useful. Here we do that, and add our only premiss.

1.	*Show* $q \wedge p$	
2.	$p \wedge q$	Prem.

Next we observe that we can apply one of our rules, simplification, to the premiss twice, obtaining p and q. In the ideal case we write the justification of each step.

1.	*Show* $q \wedge p$	
2.	$p \wedge q$	Prem.
3.	p	2 Simp.
4.	q	2 SSimp.

Next we use another rule, adjunction, to form the desired formula

1.	*Show* $q \wedge p$	
2.	$p \wedge q$	Prem.
3.	p	2 Simp.
4.	q	2 Simp.
5.	$q \wedge p$	Adj.

In a direct proof, when we obtain the formula we wanted to derive, we “box” the proof, and cancel the initial *Show*.

1.	*~~Show~~* $q \wedge p$	
2.	$p \wedge q$	Prem.
3.	p	2 Simp.
4.	q	2 Simp.
5.	$q \wedge p$	3, 4 Adj.

Although direct proof is conceptually simple, it is seldom adequate on its own.

8.2.2 Conditional proof

When the target formula is a conditional, we assume the antecdent and show that the consequent is derviable under this assumption. Take,

$$-q \to -r, \quad p \to r \vdash p \to q \tag{48}$$

Again we start with a *Show* line.

1.	*Show* $p \to q$

In a conditional proof, we start with assuming the antecedent, p in this case:

1.	*Show* $p \to q$	
2.	p	Asmp.

The aim is to derive the consequent q. In this task, in addition to the premisses, we are allowed to make use of the assumption p.From here on we proceed as in a direct proof of q, namely applying the available rules to the formulas available. It is crucial to observe that we cannot apply MP to $p \to q$ and p. Any formula that has an uncancelled *Show* is UNavailable in a proof. In order to proceed, we take a premiss that we can feed into the MP rule together with p and obtain r.

1.	*Show* $p \to q$	
2.	p	Asmp.
3.	$p \to r$	Prem.
4.	r	2, 3 MP

The rest of the proof proceeds in a similar fashion, eventually obtaining q, boxing the proof and cancelling the *Show*.

1.	*~~Show~~* $p \to q$	
2.	p	Asmp.
3.	$p \to r$	Prem.
4.	r	2, 3 MP
5.	$--r$	4 DN
6.	$-q \to -r$	Prem.
7.	$--q$	5, 6 MT
8.	q	7 DN

Now we turn to an example that calls for nested *Show*s. Take,

$$p \to (q \to r), \quad p \to (r \to s) \vdash p \to (q \to s) \tag{49}$$

We start our conditional proof by assuming p:

1.	*Show* $p \to (q \to s)$	
2.	p	Asmp.

At this point we have a new goal, proving $q \to s$. If we can do that, then we can conclude that assuming p yields $q \to s$, achieving our initial goal.

1.	*~~Show~~* $p \to (q \to s)$	
2.	p	Asmp.
3.	*Show* $(q \to s)$	
4.		

In our **subproof** we proceed as in a conditional proof, assuming q and trying to obtain s:

1.	*Show* $p \to (q \to s)$	
2.	p	Asmp.
3.	*Show* $(q \to s)$	
4.	q	Asmp.
5.		

Once we obtain s we box its proof and cancel the *Show* preceeding our interim goal $q \to s$,

1.	*Show* $p \to (q \to s)$	
2.	p	Asmp.
3.	*~~Show~~* $(q \to s)$	
4.	q	Asmp.
5.	$p \to (q \to r)$	Prem.
6.	$q \to r$	2, 5 MP
7.	r	4, 6 MP
8.	$p \to (r \to s)$	Prem.
9.	$r \to s$	2, 8 MP
10.	s	7, 9 MP

Our initial aim was to see whether we can obtain $q \to s$ under the assumption p (and of course our premisses). Any formula that is preceeded by a cancelled *~~Show~~* is available for use in the unboxed parts of our proof. Given that, we see that we derived $q \to s$ on the assumption p. We box the proof and cancel our top-most *Show*:

1.	*~~Show~~* $p \to (q \to s)$	
2.	p	Asmp.
3.	*~~Show~~* $(q \to s)$	
4.	q	Asmp.
5.	$p \to (q \to r)$	Prem.
6.	$q \to r$	2, 5 MP
7.	r	4, 6 MP
8.	$p \to (r \to s)$	Prem.
9.	$r \to s$	2, 8 MP
10.	s	7, 9 MP

8.2.3 Indirect proof

An indirect proof starts with assuming the opposite (denial) of what is being tried to be proved. If this assumption leads to a contradiction – having both P and $-P$ for some formula P, we conclude that the formula we denied in the beginning holds. This technique is sometimes called 'proof by contradiction' or *reductio ad absurdum*.

Example 8.1
Prove 50 by natural deduction.

$$-p \to q, \quad p \to q \vdash q \tag{50}$$

We start by denying our target q:

1.	*Show* q	
2.	$-q$	Asmp.

From here on our aim is to derive a contradiction. *Any* formula P such that we have both P and $-P$. We achieve this aim as follows:

1.	*Show* q	
2.	$-q$	Asmp.
3.	$p \to q$	Prem.
4.	$-p$	2, 3 TP
5.	$-p \to q$	Prem.
6.	q	4, 5 MP

Our assumption $-q$ has allowed us to derive q, resulting in a contradiction. We can now conclude that the assumption $-q$ is unatainable, and therefore q must hold. This completes the proof. We box and cancel as usual.

1.	~~*Show*~~ q	
2.	$-q$	Asmp.
3.	$p \to q$	Prem.
4.	$-p$	2, 3 MP
5.	$-p \to q$	Prem.
6.	q	4, 5 MP

□

Example 8.2
Take the following argument:

> Harry is the murderer, only if he was at the apartment around 10pm. The police will find a fingerprint, provided that he was at the apartment around 10pm. It is not the case that if he forgot to wear a glove, the police will find a fingerprint. Therefore, Harry is not the murderer.

Given the symbolization,

m: Harry is the murderer;

r: Harry was at the apartment around 10pm;

p: The police will find a fingerprint;

t: Harry forgot to wear a glove,

the argument will be:[10]

$$m \to r, \quad r \to p \quad -(t \to p) \vdash -m \tag{51}$$

We proceed with an indirect proof:

1.	*Show* $-m$	
2.	$--m$	Asmp.
3.	m	5 DN

The same proof could be started as,

1.	*Show* $-m$	
2.	m	Asmp.

leaving the application of DN implicit. Now, the aim is to derive a contradiction. The most basic strategy is to derive some formula that contradicts what we already have in the proof and unused premisses, if there are any. Let's attempt to derive $t \to p$ via a conditional subproof:

1.	*Show* $-m$	
2.	m	Asmp.
3.	~~*Show*~~ $t \to p$	
4.	t	
5.	$m \to r$	Prem.
6.	r	2, 6 MP
7.	$r \to p$	Prem.
8.	p	9, 8 MP

[10]Note that 'p only if q' is $p \to q$, while 'p provided that q' is $q \to p$.

Having proved $t \to p$, we arrived at a contradiction, namely with the premiss $-(t \to p)$:

1.	*Show* $-m$	
2.	m	Asmp.
3.	$-(t \to p)$	Prem.
4.	~~*Show*~~ $t \to p$	
5.	t	
6.	$m \to r$	Prem.
7.	r	2, 6 MP
8.	$r \to p$	Prem.
9.	p	9, 8 MP

which completes our proof as:

1.	~~*Show*~~ $-m$	
2.	m	Asmp.
3.	$-(t \to p)$	Prem.
4.	~~*Show*~~ $t \to p$	
5.	t	
6.	$m \to r$	Prem.
7.	r	2, 6 MP
8.	$r \to p$	Prem.
9.	p	9, 8 MP

□

Self study

Except natural deduction, you can review what we have covered in propositional logic from Partee *et al.* (1990), Chapter 6, up to Section 6.5. Please note some terminological differences: they use "statement logic" for "propositional logic", "logical consequence" for "implication", "disjunction" for "alternation", "contingent" for "consistent", and so on.

Please do not try to memorize Table 6.2. The caption of the table, "Laws of statement logic" is misleading. These are not laws, they are just valid formulas, which are on an equal status with any other valid formula of propositional logic. Try to see and work out why they are valid, and that's it for now.

As for the inductive definitions of the well-formed expressions of L_0 and their top-down analysis and bottom-up generation (Section 2 of the notes), they can be reviewed from the notes. If you do not fully understand them at the moment, don't worry. Give priority to the logic part.

References

Johnson-Laird, P. (1983). *Mental Models*. Cambridge University Press, Cambridge.

Partee, B. H., ter Meulen, A., and Wall, R. E. (1990). *Mathematical Methods in Linguistics*. Kluwer, Dodrecht.

Quine, W. V. O. (1958). *Mathematical Logic*. Harper and Row, New York, NY. First published in 1940 by Harvard University Press, Cambridge, MA.

Reichenbach, H. (1947). *Elements of Symbolic Logic*. University of California Press, Berkeley, CA.