LOGIQUE

ENS Paris-Saclay

L3 informatique

2025-2026

cours 2

3. Natural deduction

How to prove things, formally.

We now return to the syntactic side, and define what it means to prove something.

In fact, there are many possible definitions of "proof", and we call a proof system or calculus one such choice.

We begin with a system called natural deduction, and we first look at propositional logic.

Let P_0 be a set of atomic propositions. A sequent is an element of $\mathcal{P}(\mathrm{Form}(P_0)) \times \mathrm{Form}(P_0)$.

We will denote such a pair by $\Gamma \Rightarrow \varphi$. (Capital Greek letter for "set of formulas", lowercase Greek letter for "formula". The arrow $\Rightarrow$ is just a notation.)

We define the set $\mathcal{D}$ of (natural deduction) derivable sequents inductively by:

- (Ax) • if $\varphi \in \Gamma$, then $\Gamma \Rightarrow \varphi$ is derivable;
- ($\wedge$I) • if $\Gamma_1 \Rightarrow \varphi_1$ and $\Gamma_2 \Rightarrow \varphi_2$ are derivable, then $\Gamma_1 \cup \Gamma_2 \Rightarrow \varphi_1 \wedge \varphi_2$ is derivable;
- ($\wedge$E) • if $\Gamma \Rightarrow \varphi \wedge \psi$ is derivable, then $\Gamma \Rightarrow \varphi$ is derivable and $\Gamma \Rightarrow \psi$ is derivable;
- ($\rightarrow$I) • if $\Gamma \cup \{\varphi\} \Rightarrow \psi$ is derivable, then $\Gamma \Rightarrow \varphi \rightarrow \psi$ is derivable,
- ($\rightarrow$E) • if $\Gamma_1 \Rightarrow \varphi$ is derivable and $\Gamma_2 \Rightarrow \varphi \rightarrow \psi$ is derivable, then $\Gamma_1 \cup \Gamma_2 \Rightarrow \psi$ is derivable.
- ($\bot$E) • if $\Gamma \Rightarrow \bot$ is derivable, then, for any φ, $\Gamma \Rightarrow \varphi$ is derivable.
- (C) • if $\Gamma \cup \{\neg\varphi\} \Rightarrow \bot$ is derivable, then $\Gamma \Rightarrow \varphi$ is derivable.

We define the notation $\Gamma \vdash \varphi$ for: "the sequent $\Gamma \Rightarrow \varphi$ is derivable".

We can now write the definition of $\vdash$ more succinctly as:

$$(Ax)\ \frac{\varphi \in \Gamma}{\Gamma \vdash \varphi}$$

$$(\wedge I)\ \frac{\Gamma_1 \vdash \varphi \qquad \Gamma_2 \vdash \psi}{\Gamma_1, \Gamma_2 \vdash \varphi \wedge \psi} \qquad (\wedge E_L)\ \frac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \varphi} \qquad (\wedge E_R)\ \frac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \psi}$$

$$(\to I)\ \frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \to \psi} \qquad (\to E)\ \frac{\Gamma_1 \vdash \varphi \qquad \Gamma_2 \vdash \varphi \to \psi}{\Gamma_1, \Gamma_2 \vdash \psi}$$

$$(\bot E)\ \frac{\Gamma \vdash \bot}{\Gamma \vdash \varphi} \qquad (C)\ \frac{\Gamma, \neg\varphi \vdash \bot}{\Gamma \vdash \varphi}$$

"proof by contradiction"

Notation: $\frac{A}{B}$ means: "if A, then B".

"premises" → "conclusion" → $\frac{A_1 \quad A_2}{B}$ means: "if A_1 and A_2, then B"

Γ, φ means: "$\Gamma \cup \{\varphi\}$" and Γ_1, Γ_2 means: "$\Gamma_1 \cup \Gamma_2$"

Example. $\overset{\Gamma=\emptyset}{\vdash} \varphi \to (\neg\varphi \to \psi)$ for any formulas φ, ψ.

Proof.

$$
(\to I)\ \dfrac{(\to I)\ \dfrac{(\bot E)\ \dfrac{(\to E)\ \dfrac{(Ax)\ \dfrac{}{\varphi \vdash \varphi} \qquad (Ax)\ \dfrac{}{\neg\varphi \vdash \neg\varphi}}{\varphi, \neg\varphi \vdash \bot}}{\varphi, \neg\varphi \vdash \psi}}{\varphi \vdash \neg\varphi \to \psi}}{\vdash \varphi \to (\neg\varphi \to \psi)}
$$

$$
\longrightarrow \quad \dfrac{\overset{\Gamma_1}{\varphi} \vdash \varphi \qquad \overset{\Gamma_2}{\varphi \to \bot} \vdash \varphi \to \overset{\psi}{\bot}}{\underset{\Gamma_1, \Gamma_2}{\varphi, \varphi \to \bot} \vdash \underset{\psi}{\bot}}
$$

Example $\neg\neg\varphi \vdash \varphi$ for any formula φ

$$
(C)\ \dfrac{(\to E)\ \dfrac{(Ax)\ \dfrac{}{\neg\neg\varphi \vdash \neg\neg\varphi} \qquad (Ax)\ \dfrac{}{\neg\varphi \vdash \neg\varphi}}{\neg\neg\varphi, \neg\varphi \vdash \bot}}{\neg\neg\varphi \vdash \varphi}
$$

$$
\dfrac{\Gamma_1 \vdash \varphi' \to \psi \qquad \Gamma_2 \vdash \varphi'}{\Gamma_1 \cup \Gamma_2 \vdash \bot}
$$

$\Gamma_1 = \{\neg\neg\varphi\}$
$\Gamma_2 = \{\neg\varphi\}$
$\varphi' = \varphi \to \bot$
$\psi = \bot$

(unordered & rooted) (usually finite) child relation root

A **tree** is a tuple (T, R, t_0), where T is a set, $R \subseteq T^2$, $t_0 \in T$ such that, for any $t \in T$, there exists a unique R-path from t_0 to t.

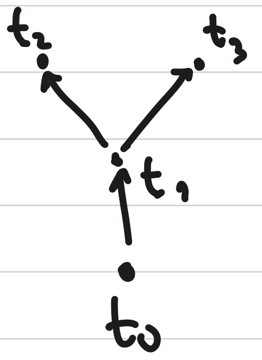
NB: proof theorists draw trees with the root at the bottom (which makes sense, botanically speaking)

A **labeling** of T by the elements of a set S is a function $\ell : T \to S$.

A **proof tree**, or a **natural deduction**, is a finite tree with a labeling by sequents, such that, for every node t, $\dfrac{\ell[R(t)]}{\ell(t)}$ is an instance of a rule of natural deduction.

(This notation means: put labels of the R-successors of t above the line, and label of t below.)

Example



$\ell(t_2) = \neg\neg\varphi \Rightarrow \neg\neg\varphi$ $\quad \ell(t_3) = \neg\varphi \Rightarrow \neg\varphi$

$\ell(t_1) = \neg\neg\varphi, \neg\varphi \Rightarrow \bot$

$\ell(t_0) = \neg\neg\varphi \Rightarrow \varphi$

$$\dfrac{\dfrac{\dfrac{}{\neg\neg\varphi \vdash \neg\neg\varphi}\,(Ax) \quad \dfrac{}{\neg\varphi \vdash \neg\varphi}\,(Ax)}{\neg\neg\varphi, \neg\varphi \vdash \bot}\,(\to E)}{\neg\neg\varphi \vdash \varphi}\,(C)$$

Then: $\dfrac{}{\ell(t_2)}$ and $\dfrac{}{\ell(t_3)}$ are instances of (Ax), $\dfrac{\ell(t_2) \quad \ell(t_3)}{\ell(t_1)}$ is an instance of $(\to E)$,

and $\dfrac{\ell(t_1)}{\ell(t_0)}$ is an instance of (C).

Theorem. Let $\Gamma \Rightarrow \varphi$ be a sequent. Then

$\Gamma \Rightarrow \varphi$ is derivable if, and only if, there exists a proof tree with label $\Gamma \Rightarrow \varphi$ at its root.

Proof. "only if": by induction on the derivability of $\Gamma \Rightarrow \varphi$, we construct a proof tree as claimed.

- case (Ax): the proof tree has a single node, labelled $\Gamma \Rightarrow \varphi$
- case ($\wedge$I): we have $\varphi = \varphi_1 \wedge \varphi_2$ and $\Gamma = \Gamma_1 \cup \Gamma_2$ with $\Gamma_1 \Rightarrow \varphi_1$ and $\Gamma_2 \Rightarrow \varphi_2$ derivable sequents.

  By the IH, pick proof trees π_i for $\Gamma_i \Rightarrow \varphi_i$, for $i = 1,2$. Create a new tree $\pi := \pi_0 \sqcup \pi_1 \sqcup \{t_0\}$, with the same labeling as π_1, π_2, and $\ell(t_0) := \Gamma \Rightarrow \varphi$. This is again a proof tree: for all $t \neq t_0$ this follows because π_1 and π_2 are proof trees, and for t_0, we have an instance of the rule ($\wedge$I).

  The other cases are proved similarly (Exercise).

"if": by induction on the height of the proof tree, we show that $\Gamma \vdash \varphi$.

- height = 0 : $\Gamma \Rightarrow \varphi$ must be an instance of (Ax), since it's the only rule without premises. So $\Gamma \vdash \varphi$.
- height = n+1: $\dfrac{\ell(R[t_0])}{\ell(t_0)}$ is an instance of a rule (R). We do a case distinction on (R).

"continued" (= Fr. "suite")

Proof (c't'd) • case $R = Ax$: impossible since height > 0.

• case $R = \wedge I$: we have

$$\begin{tikzcd}[column sep=small] t_1 & & t_2 \\ & t_0 \arrow[ul] \arrow[ur] & \end{tikzcd}$$

with $\dfrac{\ell(t_1) \quad \ell(t_2)}{\ell(t_0)}$ an instance of $(\wedge I)$. For $i = 1, 2$, the subtree rooted at t_i is a proof tree, of height n. By IH, $\ell(t_1)$ and $\ell(t_2)$ are derivable. By $(\wedge I)$ in the definition of derivability, $\ell(t_0)$ is derivable too.

• all other cases are proved in the same way (exercise). $\square$

Remark. This Theorem is in fact an instance of a very general principle: if a concept is defined inductively, then it can be characterized using a tree-like structures. In programming languages, this principle is used for implementing inductive data types.

Fact. If $\Gamma \vdash \varphi$, and ψ is any formula, then $\Gamma \vdash \varphi \vee \psi$. $(\vee I_L)$

Proof. Let $\pi : \Gamma \vdash \varphi$. (This means: "$\pi$ is a proof tree with root label $\Gamma \Rightarrow \varphi$".)

We build a proof of $\Gamma \vdash \varphi \vee \psi$:

$$\dfrac{\dfrac{\overset{\pi}{\Gamma \vdash \varphi} \qquad \dfrac{\Gamma, \neg\varphi \wedge \neg\psi \vdash \neg\varphi \wedge \neg\psi}{\Gamma, \neg\varphi \wedge \neg\psi \vdash \neg\varphi}\,(\mathrm{Ax})}{\Gamma, \neg\varphi \wedge \neg\psi \vdash \bot}\,(\to E)}{\Gamma \vdash \neg(\neg\varphi \wedge \neg\psi)}\,(\to I)$$

$\square$

Facts.

- If $\Gamma \vdash \psi$ and φ is any formula, then $\Gamma \vdash \varphi \vee \psi$. $(\vee I_R)$
- If $\Gamma, \varphi \vdash \theta$ and $\Gamma, \psi \vdash \theta$, then $\Gamma, \varphi \vee \psi \vdash \theta$. $(\vee E)$
- If $\Gamma, \varphi \vdash \bot$ then $\Gamma \vdash \neg\varphi$ $(\neg I)$
- If $\Gamma \vdash \varphi \wedge \psi$ and $\Gamma, \varphi, \psi \vdash \theta$, then $\Gamma \vdash \theta$. $(\wedge E')$ (Exercise.)

Remark. We could add these rules to the definition of "derivable", and it would not change $\Gamma \vdash \varphi$. Rules with this property are called admissible.

For Γ a finite set of formulas, we define a formula $\bigwedge\Gamma$ by induction on $\#\Gamma$:

$$\bigwedge\emptyset := \top \quad \text{and} \quad \bigwedge(\Gamma' \cup \{\varphi\}) := \left(\bigwedge\Gamma'\right) \wedge \varphi.$$

Similarly, $\bigvee\emptyset := \bot$ and $\bigvee(\Gamma'\cup\{\varphi\}) := \left(\bigvee\Gamma'\right)\vee\varphi$.

Deduction Theorem. Let Γ be a finite set of formulas and φ a formula.

Then $\Gamma \vdash \varphi$ if, and only if, $\vdash \left(\bigwedge\Gamma\right) \to \varphi$.

Proof. Induction on $\#\Gamma$. • When $\Gamma = \emptyset$, we need to show $\vdash\varphi$ iff $\vdash \top\to\varphi$. see TD

If $\vdash\varphi$, then we can add $\top$ on the left everywhere in the proof tree to get $\top\vdash\varphi$ (weakening).

Thus, $\top\vdash\varphi$, and by $(\to I)$ we get $\vdash\top\to\varphi$.

If π proves $\vdash \top\to\varphi$, then we get a proof of $\vdash\varphi$:

$$\dfrac{\dfrac{\dfrac{}{\bot\vdash\bot}\,(ax)}{\vdash\top}\,(\to I) \qquad \begin{array}{c}\pi\\ \vdash\top\to\varphi\end{array}}{\vdash\varphi}\,(\to E)$$

• $\Gamma = \Gamma'\cup\{\psi\}$. If $\Gamma\vdash\varphi$, then by $(\to I)$ we have $\Gamma'\vdash\psi\to\varphi$. By the IH we get $\vdash\bigwedge\Gamma'\to(\psi\to\varphi)$.

Now use $\vdash (\theta\to(\psi\to\varphi))\to((\theta\wedge\psi)\to\varphi)$ (exercise) to get $\vdash\left(\bigwedge\Gamma\right)\to\varphi$.

If $\vdash\left(\bigwedge\Gamma\right)\to\varphi$, use the converse of to get $\vdash\left(\bigwedge\Gamma'\right)\to(\psi\to\varphi)$. By IH and $(\to E)$, $\Gamma\vdash\varphi$. $\square$

Note. The goal of logic is not to formally write proof trees ad infinitum.
Rather, it is to study logical notions, which is typically done at the meta level.
For example, we will next prove:

Theorem. For any sequent $\Gamma \Rightarrow \varphi$, we have:

$$\Gamma \vdash \varphi \quad \text{if, and only if,} \quad \Gamma \models \varphi .$$

The left-to-right direction is called soundness (Fr: correction) of the natural deduction system.
" right-to-left ——" —— completeness (Fr: complétude) ——" ————.

Proof of soundness. Let $\Gamma \Rightarrow \varphi$ be a derivable sequent. We prove by induction that $\Gamma \vDash \varphi$, i.e., for every interpretation M, if $M \vDash \gamma$ for every $\gamma \in \Gamma$, then $M \vDash \varphi$.

- case (Ax). Then we must have $\varphi \in \Gamma$, so $\Gamma \vDash \varphi$ trivially.
- case ($\wedge$I): $\dfrac{\Gamma_1 \vdash \varphi_1 \quad \Gamma_2 \vdash \varphi_2}{\Gamma_1 \cup \Gamma_2 \vdash \varphi_1 \wedge \varphi_2}$. Suppose $M \vDash \Gamma_1 \cup \Gamma_2$ (i.e. $[\![\gamma]\!]_M = 1$ for all $\gamma \in \Gamma_1 \cup \Gamma_2$). By IH, $\Gamma_1 \vDash \varphi_1$ and $\Gamma_2 \vDash \varphi_2$. Thus, $[\![\varphi_1]\!]_M = 1$ and $[\![\varphi_2]\!]_M = 1$. By def. of $[\![\ ]\!]_M$, we have $[\![\varphi_1 \wedge \varphi_2]\!]_M = 1$
- case ($\wedge$E$_L$): $\dfrac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \varphi}$. Suppose $M \vDash \Gamma$. By IH, $\Gamma \vDash \varphi \wedge \psi$, so $M \vDash \varphi \wedge \psi$. Thus, $M \vDash \varphi$.
- case ($\wedge$E$_R$): ——— " ——— $M \vDash \psi$.
- case ($\rightarrow$I): $\dfrac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \rightarrow \psi}$. Suppose $M \vDash \Gamma$. Two cases: 1) $M \vDash \varphi$. Then by IH, $\Gamma, \varphi \vDash \psi$, so $M \vDash \psi$.
  2) $M \nvDash \varphi$. Then $M \vDash \varphi \rightarrow \psi$ by definition.
- case ($\bot$E): $\dfrac{\Gamma \vdash \bot}{\Gamma \vdash \varphi}$.
  By IH, $\Gamma \vDash \bot$. So there does not exist M such that $M \vDash \Gamma$. So $\Gamma \vDash \varphi$.
- case (C): $\dfrac{\Gamma, \neg\varphi \vdash \bot}{\Gamma \vdash \varphi}$. Suppose $M \vDash \Gamma$. Towards a contradiction, suppose $M \nvDash \varphi$. Then $M \vDash \neg\varphi$. By IH, $M \vDash \bot$, contradiction with the definition of "$M \vDash$".
- case ($\rightarrow$E): Exercise.

$\square$

Remark. We use the rules of our metatheory to prove soundness of the corresponding rules for $\vdash$.

4. Tableaux

How to construct natural deductions for all valid formulas.

Tableaux provide an algorithm which, for a propositional formula φ, outputs one of the following:

(1) a natural deduction **proof** of $\vdash \varphi$, or

(2) an **interpretation** $v : P_0(\varphi) \to 2$ such that $v \nvDash \varphi$.

atomic propositions used in φ

Note. By **soundness**, we know that the "or" is exclusive.

The fact that such an algorithm exists then in particular implies:

Classical Propositional Logic — Natural Deduction

Corollary (Weak **completeness** of CPL-ND)

For any propositional formula φ, if $v \vDash \varphi$ for all interpretations $v : P_0(\varphi) \to 2$, then $\emptyset \Rightarrow \varphi$ is derivable.

if $\vDash \varphi$ then $\vdash \varphi$

Proof that algorithm $\Rightarrow$ corollary. Suppose $\vDash \varphi$. Run the algorithm on input φ. Since no output of type (2) exists, it must return a proof of $\vdash \varphi$. $\square$

Tableau algorithm: preliminary definitions.

$\neg\varphi := \varphi \to \bot$
$\varphi \vee \psi := \neg\varphi \to \psi$
$\varphi \wedge \psi := \neg(\varphi \to \neg\psi)$

- A formula (for the tableau algorithm) is built from atoms in P_0, $\to$, and $\bot$.
- A signed formula is a pair (φ, s), where φ is a formula and $s \in \{+, -\}$. Notation: φ^s.

  $-+ := -$
  $-- := +$
- A clause is a finite set of signed formulas.
- A clause C is solved (or satisfiable) if:
  * the set C only contains signed atoms, and
  * there does not exist $p \in P_0$ such that both p^+ and p^- are in C.
- For $v : P_0 \to 2$, we write $v \models \varphi^+$ if $v(\varphi) = 1$ and $v \models \varphi^-$ if $v(\varphi) = 0$, and,

  for C a clause, we also write $v \models C$ if $v \models \varphi^s$ for all $\varphi^s \in C$.
- When $\mathscr{C}$ is a finite set of clauses we write $v \models \mathscr{C}$ if there exists $C \in \mathscr{C}$ such that $v \models C$.

Lemma. For any solved clause C, there exists $v : P_0 \to 2$ such that $v \models C$.

Proof. Define $v(p) := \begin{cases} 1 & \text{if } p^+ \in C \\ 0 & \text{otherwise} \end{cases}$. Since C is solved, if $p^- \in C$, then $p^+ \notin C$, so $v(p) = 0$. $\square$

- We can convert a clause into a formula:
  - for any formula φ, define $\gamma(\varphi^+) := \varphi$ and $\gamma(\varphi^-) := \neg\varphi$,
  - for any clause C, define $\Gamma(C) := \{\gamma(\varphi^s) \mid \varphi^s \in C\}$, $\gamma(C) := \bigwedge \Gamma(C)$.

Example If $C = \{p^+, (q \to r)^-, (p \to \bot)^-\}$, then $\gamma(C) = p \wedge \neg(q \to r) \wedge \neg(p \to \bot)$.

- We can also convert a finite set of clauses $\mathcal{C}$ into a formula: $\delta(\mathcal{C}) := \bigvee \{\gamma(C) \mid C \in \mathcal{C}\}$.

Exercise What is $\gamma(\emptyset)$? What is $\delta(\emptyset)$? What is $\delta(\{\emptyset\})$?

Observation. For any $v : P_0 \to 2$, we have $v \models C$ iff $v \models \gamma(C)$ and
$v \models \mathcal{C}$ iff $v \models \delta(\mathcal{C})$.

A formula φ is in disjunctive normal form if $\varphi = \delta(\mathcal{C})$ for some finite set of clauses $\mathcal{C}$.

Theorem. For any formula φ, there exists a finite set of clauses $\mathcal{C}$ such that $\models \varphi \leftrightarrow \delta(\mathcal{C})$

Proof 1. Let $\mathcal{C} = \{C$ a solved clause in $P_0(\varphi)$, and $\models \gamma(C) \to \varphi\}$. Then $\models \varphi \leftrightarrow \delta(\mathcal{C})$. $\square$

(atoms occurring in φ

Proof 2. Using tableaux, see below.

Proof 3. Rewriting (see TD).

Q. Complexity?

We recursively define a procedure DECIDE_AUX which takes as input a pair (C, D), where C is a solved clause, D is any clause, and returns either a proof of $\Gamma(C \cup D) \vdash \bot$,

or an interpretation v such that $v \nvDash \gamma(C \cup D)$:

- If $D = \emptyset$, return v such that $v \vDash C$ (since C is solved, by Lemma)
- If $D = D' \sqcup \{\varphi^s\}$, distinguish cases: (Note: φ^s is chosen from D non-deterministically.)
  * $\varphi = p \in P_0$ if $p^{-s} \in C$, return a proof of $\Gamma(C \cup D) \vdash \bot$ (1).
    if $p^{-s} \notin C$, return DECIDE_AUX$(C \cup \{p^s\}, D')$.
  * $\varphi^s = \bot^+$: return a proof (2).
  * $\varphi^s = \bot^-$: if DECIDE_AUX(C, D') is a proof, return a proof (3), else, return the same interpretation (i).
  * $\varphi = (\varphi_1 \to \varphi_2)^+$: let $r_1 :=$ DECIDE_AUX$(C, D' \cup \{\varphi_1^-\})$ and $r_2 :=$ DECIDE_AUX$(C, D' \cup \{\varphi_2^+\})$.
    if r_1 or r_2 is an interpretation, return it (ii), else return a proof (4).
  * $\varphi = (\varphi_1 \to \varphi_2)^-$: if DECIDE_AUX$(C, D' \cup \{\varphi_1^+, \varphi_2^-\})$ is an interpretation, return it (iii), else return a proof (5).

Example.

$C = \emptyset,\ D = \{(p \to (q \to r)) \to (q \to p)^-\}$ $\quad (\varphi_1 \to \varphi_2)^-$

$C = \emptyset,\ D = \{p \to (q \to r)^+,\ q \to p^-\}$ $\quad (\varphi_1 \to \varphi_2)^-$

$C = \emptyset,\ D = \{p \to (q \to r)^+,\ q^+,\ p^-\}$ $\quad$ 2 steps: $q^+,\ p^-$

$C = \{q^+, p^-\},\ D = \{p \to (q \to r)^+\}$

branching $(\varphi_1 \to \varphi_2)^+$

$C = \{q^+, p^-\},\ D = \{p^-\}$

$C = \{q^+, p^-\},\ D = \emptyset$

$v: p \mapsto 0$
$\quad q \mapsto 1$
$\quad r \mapsto 0$ (or 1)

$C = \{q^+, p^-\},\ D = \{(q \to r)^+\}$

branching $(\varphi_1 \to \varphi_2)^+$

$C = \{q^+, p^-\},\ D = \{q^-\}$ $\quad q^{-+} \in C$

$q, \neg p, q \vdash \bot$

$C = \{q^+, p^-\},\ D = \{r^+\}$ $\quad r^{-+} \notin C$

$C = \{q^+, p^-, r^+\},\ D = \emptyset$

$v: p \mapsto 0$
$\quad q \mapsto 1$
$\quad r \mapsto 1$

The above execution trace of DECIDE_AUX is called a *tableau*.

A more compact notation for the same calculation:

1. $(p \to (q \to r)) \to (q \to p)^-$
2. $p \to (q \to r)^+$ (1)
3. $(q \to p)^-$ (1)
4. q^+ (3)
5. p^- (3)

6. p^- (2)	7. $(q \to r)^+$ (2)	
O	8. q^- (7)	9. r^+ (7)
	X (4,8)	O

The left open branch gives $C_1 := \{p^-, q^+\}$, the right open branch gives $C_2 := \{p^-, q^+, r^+\}$.

We have $\delta(\{C_1, C_2\}) = (\neg p \wedge q) \vee (\neg p \wedge q \wedge r)$, which is equivalent to $\neg((p \to (q \to r)) \to (q \to p))$.

- Given the procedure DECIDE_AUX : solved clause $\times$ clause $\longrightarrow$ interpretation + proof, we define, for any clause D, $\text{DECIDE}(D) := \text{DECIDE\_AUX}(\emptyset, D)$, and for any formula φ, $\text{SOLVE}(\varphi) := \text{DECIDE}(\{\varphi^-\})$.
- If SOLVE(φ) returns a proof of $\neg\varphi \vdash \bot$, then applying rule (C) we get $\vdash \varphi$
- If SOLVE(φ) returns an interpretation v such that $v \models \neg\varphi$, then $v \not\models \varphi$.
- Our procedure DECIDE_AUX relies on five proof constructions (i.e. admissible rules):

(1) if $p \in \Gamma$ and $\neg p \in \Gamma$, then $\Gamma \vdash \bot$.

(2) if $\bot \in \Gamma$, then $\Gamma \vdash \bot$.

(3) if $\Gamma \vdash \bot$, then $\Gamma, \neg\bot \vdash \bot$.

(exercises)

(4) if $\Gamma, \neg\varphi_1 \vdash \bot$ and $\Gamma, \varphi_2 \vdash \bot$, then $\Gamma, \varphi_1 \to \varphi_2 \vdash \bot$

(5) if $\Gamma, \varphi_1, \neg\varphi_2 \vdash \bot$, then $\Gamma, \neg(\varphi_1 \to \varphi_2) \vdash \bot$.

and its correctness relies on three more facts:

(i) $v \models \Gamma \cup \{\neg\bot\}$ iff $v \models \Gamma$

(ii) if $v \models C \cup D' \cup \{\neg\varphi_1\}$ or $v \models C \cup D' \cup \{\varphi_2\}$, then $v \models C \cup D' \cup \{\varphi_1 \to \varphi_2\}$.

(iii) if $v \models C \cup D' \cup \{\varphi_1, \neg\varphi_2\}$, then $v \models C \cup D' \cup \{\varphi_1 \to \varphi_2\}$.

(Exercises)

(4) if $\Gamma, \neg\varphi_1 \overset{\pi_1}{\vdash} \bot$ and $\Gamma, \varphi_2 \overset{\pi_2}{\vdash} \bot$, then $\Gamma, \varphi_1 \to \varphi_2 \vdash \bot$

$$
\dfrac{
\dfrac{
\dfrac{\overset{\pi_1}{\Gamma, \neg\varphi_1 \vdash \bot}}{\Gamma \vdash \varphi_1}\,(C)
\qquad
\dfrac{}{\varphi_1 \to \varphi_2 \vdash \varphi_1 \to \varphi_2}\,(Ax)
}{\Gamma, \varphi_1 \to \varphi_2 \vdash \varphi_2}\,(\to E)
\qquad
\dfrac{\overset{\pi_2}{\Gamma, \varphi_2 \vdash \bot}}{\Gamma \vdash \varphi_2 \to \bot}\,(\to I)
}{\Gamma, \varphi_1 \to \varphi_2 \vdash \bot}\,(\to E)
$$

(5) if $\Gamma, \varphi_1, \neg\varphi_2 \overset{\pi}{\vdash} \bot$, then $\Gamma, \neg(\varphi_1 \to \varphi_2) \vdash \bot$.

$$
\dfrac{
\dfrac{
\dfrac{\overset{\pi}{\Gamma, \varphi_1, \neg\varphi_2 \vdash \bot}}{\Gamma, \varphi_1 \vdash \varphi_2}\,(C)
}{\Gamma \vdash \varphi_1 \to \varphi_2}\,(\to I)
\qquad
\dfrac{}{(\varphi_1 \to \varphi_2) \to \bot \vdash (\varphi_1 \to \varphi_2) \to \bot}\,(Ax)
}{\Gamma, \neg(\varphi_1 \to \varphi_2) \vdash \bot}\,(\to E)
$$

Theorem. DECIDE_AUX terminates and is correct on all inputs.

Proof. • For the termination, we define, for any formula φ, $n(\varphi)$ to be the number of symbols in φ, and, for any clause C, $n(C) := \sum_{\varphi^s \in C} n(\varphi)$. Observe that the definition of DECIDE_AUX(C, D) only makes recursive calls on pairs C', D' where $n(D') < n(D)$.

• For the correctness, reason by induction on $n(D)$:

* if $n(D) = 0$, then $D = \emptyset$, and $v \models C \cup D$ since $v \models C$.
* if $n(D) > 0$, distinguish cases according to $\varphi^s \in D$, and use (i)-(iii) to conclude that, in each case, if an assignment v is returned, then $v \models_g (C \cup D)$. □

Remark. Regarding $\neg, \vee, \wedge$ as abbreviations makes our proofs shorter (fewer cases to consider), but is not very practical. We can introduce additional cases in DECIDE_AUX:

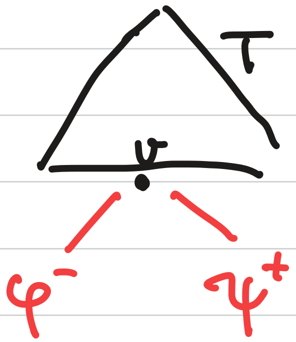
Given $D = D' \sqcup \{\varphi\}$:

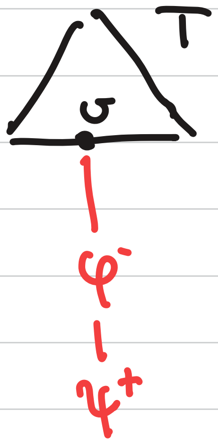
- if $\varphi = (\neg\psi)^s$, return DECIDE_AUX$(C, D' \cup \{\psi^{-s}\})$
- if $\varphi = (\psi_1 \vee \psi_2)^+$, if DECIDE_AUX$(C, D' \cup \{\psi_1^+\})$ or DECIDE_AUX$(C, D' \cup \{\psi_2^+\})$ is some v, return it,
  else, use the two proofs $C, D', \psi_1 \vdash \bot$ and $C, D', \psi_2 \vdash \bot$ to get $C, D \vdash \bot$
- if $\varphi = (\psi_1 \vee \psi_2)^-$, if DECIDE_AUX$(C, D' \cup \{\psi_1^-, \psi_2^-\})$ is some v, return it,
  else, use the proof $C, D', \neg\psi_1, \neg\psi_2 \vdash \bot$ to get $C, D', \neg(\psi_1 \vee \psi_2) \vdash \bot$.
- similarly for $\wedge$.

These "new" parts of DECIDE_AUX in fact have the same result as the "old" definition, where we regarded $\neg, \vee, \wedge$ as abbreviations.

Tableaux : formal definitions.

A **tableau** is a finite rooted tree with a labeling by signed formulas, inductively defined by:

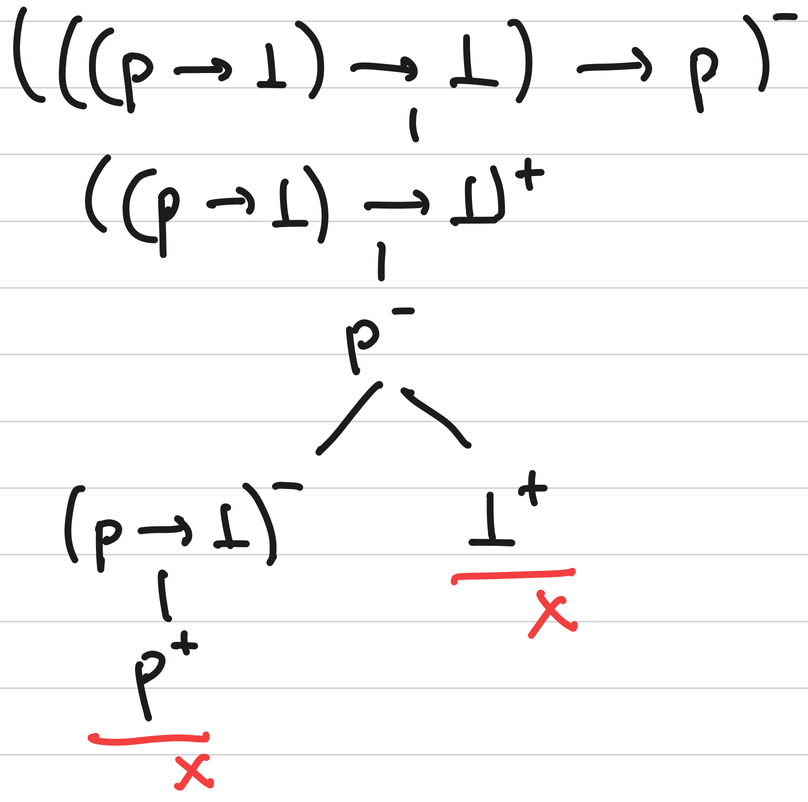
- for any φ^s, $\bullet\ \varphi^s$ is a tableau
- if T is a tableau and $(\varphi \to \psi)^+$ a label on a branch ending in v, then 

is a tableau
- ———— " ———— $(\varphi \to \psi)^-$ ———— " ———— 

———— " ————

- A **branch** is a path from the root to a leaf.
- A branch is **closed** if it contains either $\bot^+$, or both p^+ and p^-, for some atom p; **open** otherwise.
- A tableau is **closed** if all of its branches are closed.

Example



A closed tableau.

Compare with $\text{DECIDE\_AUX}(\emptyset, \{((p \to \bot) \to \bot) \to p^-\})$:

$\emptyset, \{(\neg\neg p \to p)^-\} \to \emptyset, \{\neg\neg p^+, p^-\} \to \emptyset, \{\bot^+, p^-\} \to \neg p, \bot \vdash \bot$

$\searrow \emptyset, \{\neg p^-, p^-\} \to \emptyset, \{p^+, \bot^-, p^-\} \to p, \neg\bot, \neg p \vdash \bot.$

From the tableau procedure, we can obtain a disjunctive normal form:

Let φ be a formula, and let T be a tableau with φ^s at the root, such that for every open branch B of T, no further rules can be applied (exhaustivity assumption).

To each open branch B of T, we associate the solved clause $C(B)$ of signed atoms occurring as labels of B, and we define $\mathcal{C}(T) := \{ C(B) \mid B \text{ an open branch in } T \}$

Theorem. $\gamma(\varphi^s)$ is equivalent to $\delta(\mathcal{C}(T))$.

Proof. By induction on φ. If φ is an atom, then there is one branch, and $C(B) = \{\varphi\}$.

If $\varphi^s = (\varphi_1 \to \varphi_2)^+$, then on any open branch B, we must have either φ_1^- or φ_2^+, by exhaustivity.

By IH, if $v \models C(B)$, then $v \models \neg\varphi_1$ or $v \models \varphi_2$. So $v \models \varphi_1 \to \varphi_2$. So $\delta(\mathcal{C}(T)) \models \varphi_1 \to \varphi_2$.

If, conversely, $v \models \varphi_1 \to \varphi_2$, then $v \models \neg\varphi_1$ or $v \models \varphi_2$. Say the first holds. Pick a branch B with a node labelled $\neg\varphi_1$. Since $v \models \neg\varphi_1$, $v \models C(B)$ by the IH. In particular $v \models \delta(\mathcal{C}(T))$.

Other cases: exercise. $\square$

5. Compactness

Using topology to strengthen the completeness theorem.

We will prove:

for all interpretations $v : P_0 \to 2$, if $v \models \Gamma$ then $v \models \varphi$.

Theorem. (Strong completeness of CPL-ND)

For any set of formulas Γ and any formula φ, if $\Gamma \models \varphi$, then $\Gamma \vdash \varphi$.

Note. The set Γ can be infinite.

Strong completeness will be deduced from weak completeness + compactness:

Theorem. (Compactness for CPL)

For any set of formulas Γ and any formula φ, if $\Gamma \models \varphi$, then there exists finite $\Gamma' \subseteq \Gamma$ such that $\Gamma' \models \varphi$.

Proof that weak completeness + compactness $\Rightarrow$ strong completeness.

Suppose $\Gamma \models \varphi$. By compactness, pick $\Gamma' \subseteq \Gamma$ finite such that $\Gamma' \models \varphi$.

Define $\varphi' := (\bigwedge \Gamma') \to \varphi$. (NB φ' is a formula since Γ' is finite. "$\bigwedge \Gamma \to \varphi$" is not a formula if Γ is infinite.)

Since $\Gamma' \models \varphi$, we have $\models \varphi'$ (exercise).

By weak completeness, $\vdash \varphi'$. Therefore, $\Gamma' \vdash \varphi$, by the Deduction Theorem. So $\Gamma \vdash \varphi$ by weakening. $\square$

Fr: quasi-compact

Recall that a topological space X is compact if any open cover of X contains a finite subcover. Here, an open cover is a set $\mathcal{U}$ of open subsets of X such that $X \subseteq \bigcup \mathcal{U}$, and a subcover is $\mathcal{U}' \subseteq \mathcal{U}$ such that $X \subseteq \bigcup \mathcal{U}'$.

We will use the (generalized) Cantor space, 2^{P_0}:

- the points of 2^{P_0} are interpretations, i.e., functions $v : P_0 \to 2$;
- the topology on 2^{P_0} is generated by the sets of the form $\hat{p} := \{ v \in 2^{P_0} \mid v(p) = 1 \}$ and $\widehat{\neg p} := \{ v \in 2^{P_0} \mid v(p) = 0 \}$, for $p \in P_0$.

(When P_0 is countably infinite, this is homeomorphic to the Cantor set $C \subseteq [0,1]$.)

Proposition The space 2^{P_0} is compact, for any set P_0.

Proof 1. By Tychonoff's theorem, any product of compact spaces is compact. Note that 2^{P_0} is the product $\prod_{p \in P_0} 2$, where 2 is the finite discrete space with two points. $\square$

We will see a second proof, not using Tychonoff's Theorem, later, for the case P_0 countable.

For any formula φ, define $\hat{\varphi} := \{ v \in 2^{P_0} \mid v \models \varphi \}$

Lemma. $\hat{\varphi}$ is clopen (i.e., closed and open) in 2^{P_0}.

Proof. Induction on φ. For $\varphi = p \in P_0$, note that $(\hat{p})^c = \widehat{\neg p}$ is open and $\hat{p}$ is open, by definition.

For $\varphi = \bot$, we have $\hat{\varphi} = \emptyset$, which is clopen.

For $\varphi = \varphi_1 \to \varphi_2$, we have $\hat{\varphi} = \hat{\varphi}_1^c \cup \hat{\varphi}_2$. By IH, $\hat{\varphi}_1$ and $\hat{\varphi}_2$ are clopen, so $\hat{\varphi}$ is too. $\square$

Proof of compactness of CPL. Suppose $\Gamma \models \varphi$. Then $\mathcal{U} := \{ \hat{\gamma}^c : \gamma \in \Gamma \} \cup \{ \hat{\varphi} \}$ is an open cover of 2^{P_0}:

For any $v \in 2^{P_0}$, if $v \notin \hat{\gamma}^c$ for all $\gamma \in \Gamma$, then $v \models \Gamma$, so $v \models \varphi$ by assumption.

Since 2^{P_0} is compact, pick a finite subcover $\mathcal{U}'$ of $\mathcal{U}$.

Pick $\Gamma' \subseteq \Gamma$ finite such that $\mathcal{U}' \setminus \{ \hat{\varphi} \} = \{ \hat{\gamma} \mid \gamma \in \Gamma' \}$, so $2^{P_0} \subseteq \bigcup_{\gamma \in \Gamma'} \hat{\gamma}^c \cup \hat{\varphi}$.

Now $\Gamma' \models \varphi$: if $v \models \Gamma'$, then $v \in 2^{P_0} - \bigcup_{\gamma \in \Gamma'} \hat{\gamma}^c$, so $v \in \hat{\varphi}$. $\square$

Proposition. A graph $G = (V, E)$ is 2-colorable if all of its finite subgraphs are 2-colorable.

Proof. Let $G = (V, E)$ be a graph.

Define $P_0 := V$, and define $\Gamma := \{ p \text{ XOR } q \mid \{p, q\} \in E \}$.

Note that $v : V \to 2$ is a 2-coloring of G if, and only if, $v \vDash \Gamma$.

Therefore, G is not 2-colorable iff $\Gamma \vDash \perp$.

In this case, by compactness, pick $\Gamma' \subseteq \Gamma$ finite such that $\Gamma' \vDash \perp$.

Then define $V' := \{ p \in V \mid p \text{ appears in } \Gamma' \}$, which is finite.

The subgraph on the set of nodes V' is not 2-colorable:

if $v : V' \to 2$ is a 2-coloring, then in particular $v \vDash \Gamma'$, which is impossible. $\square$

Exercise. Give a similar proof that, for any $k \geq 2$, G is k-colorable if all its finite subgraphs are k-colorable.

Proof 2 of compactness of the space 2^{P_0}. In case $P_0 = \mathbb{N}$.

$v|_n :=$ length-n prefix of v.

A **base** for the topology on 2 is $\{w 2^{\mathbb{N}} : w \in 2^*\}$, where $w 2^{\mathbb{N}} := \{ v \in 2^{\mathbb{N}} \mid \underbrace{v|_{|w|}}_{} = w \}$. (Exercise.)

Let $L \subseteq 2^*$ be a set of finite words and suppose $2^{\mathbb{N}} \subseteq \bigcup_{w \in L} w 2^{\mathbb{N}}$. If $\varepsilon \in L$, $\{2^{\mathbb{N}}\}$ is a finite subcover. Assume $\varepsilon \notin L$.

Claim. The set $F := 2^* \setminus (L \cdot 2^*) = \{ u \in 2^* \mid u \text{ has no prefix in } L\}$ is finite.

Proof. Suppose F were infinite. We will define $v \in 2^{\mathbb{N}}$ inductively in such a way that, for any $n \in \mathbb{N}$, the set

$v|_n 2^* \cap F$ is infinite. The base case of the induction holds by hypothesis $(v_0 = \varepsilon)$.

For the induction step, suppose $v|_n$ has been defined. Since $v|_n 2^* = v|_n 0 \cdot 2^* \cup v|_n 1 \cdot 2^*$, we can

choose $b \in \{0,1\}$ such that $v|_n b \cdot 2^* \cap F$ is still infinite, and define $v[n] := b$.

Now $v \in 2^{\mathbb{N}}$, but if $w \in L$ of length $n := |w|$, then $v|_n \neq w$, since $w 2^* \cap F$ is empty. $\square$

Consider $L' := \{ ub \mid u \in F,\ b \in \{0,1\} \text{ and } ub \in L\}$.

Claim. $\{ w 2^{\mathbb{N}} \mid w \in L' \}$ is a finite subcover.

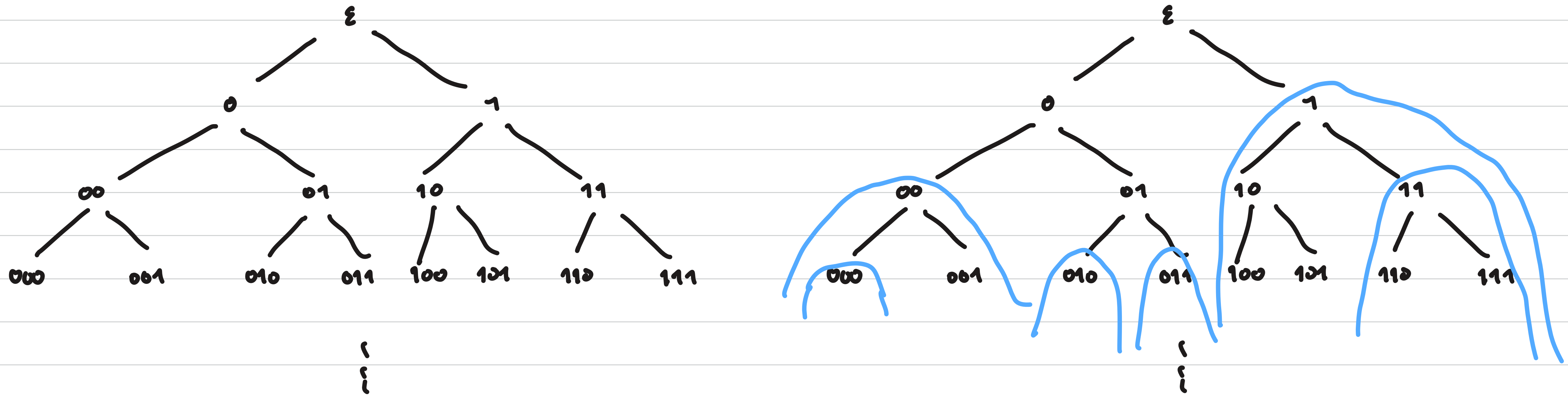
Proof. Let $v \in 2^{\mathbb{N}}$. Let w be the shortest prefix of v that is in L. Then $|w| > 0$ since $\varepsilon \notin L$, and $w|_{|w|-1} \in F$, so $w \in L'$. $\square$

The argument given in the first claim is an instance of:

König's Lemma. Any infinite, finitely branching tree has an infinite path. (Also see: Brouwer's "fan theorem".)

This uses a countable choice axiom. Extending to arbitrary P_0 requires stronger choice axioms.

Example. (Even though L is already finite, we still reduce.)



$L = \{00, 000, 010, 011, 1, 11\}$

$F = \{\varepsilon, 0, 01\}$

$L' = \{1, 00, 010, 011\}$

Logical interpretation: (negate everything!)

$\{\overset{00}{p\vee q},\ \overset{000}{p\vee q\vee r},\ \overset{010}{p\vee\neg q\vee r},\ \overset{011}{p\vee\neg q\vee\neg r},\ \overset{1}{\neg p},\ \overset{11}{\neg p\vee\neg q}\}$ is unsatisfiable,

but the subset $\{p\vee q,\ p\vee\neg q\vee r,\ p\vee\neg q\vee\neg r,\ \neg p\}$ is already unsatisfiable too.