Basic Logic propositional logic

Formulas defined inductively

$PROP = \{p, q, r, \dots\}$ countable set of propositional vars
φ, ψ etc are meta variables

FORMULAS (i) A prop. var is a formula $\bot$, ~~T~~ is formulas
(ii) if φ is a formula then $\neg\varphi$ is a formula
(iii) if φ, ψ are formulas then $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \Rightarrow \psi$ are formulas
~~(iv)~~

A proof system is a mechanism for deriving formulas from given formulas as assumptions

Basic format: $\Gamma \vdash \varphi$ Judgment
Γ a set of formulas, φ a formula.
Γ, φ for $\Gamma \cup \{\varphi\}$ · Γ_1, Γ_2 – $\Gamma_1 \cup \Gamma_2$

Rule
$$\frac{\Gamma_1 \vdash \varphi_1 \quad \cdots \quad \Gamma_n \vdash \varphi_n}{\Gamma \vdash \varphi}$$

AXIOM $\Gamma \vdash \varphi$ if $\varphi \in \Gamma$

RULES

INTRO

$$\frac{\Gamma_1 \vdash \varphi_1 \quad \Gamma_2 \vdash \varphi_2}{\Gamma \vdash \varphi_1 \wedge \varphi_2}$$

$$\frac{\Gamma, \varphi \vdash \psi}{\Gamma \vdash \varphi \Rightarrow \psi}$$

$$\frac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi \vee \psi} \qquad \frac{\Gamma \vdash \varphi}{\Gamma \vdash \psi \vee \varphi}$$

$$\cancel{\frac{\Gamma_1 \vdash \varphi \quad \Gamma_2 \vdash \neg\varphi}{\Gamma \vdash \bot}}$$

$$\frac{\Gamma, \varphi \vdash \bot}{\Gamma \vdash \neg\varphi}$$

ELIM

$$\frac{\Gamma \vdash \varphi_1 \wedge \varphi_2}{\Gamma \vdash \varphi_1} \qquad \frac{\Gamma \vdash \varphi_1 \wedge \varphi_2}{\Gamma \vdash \varphi_2}$$

$$\frac{\Gamma_1 \vdash \varphi \Rightarrow \psi \qquad \Gamma_2 \vdash \varphi}{\Gamma_1, \Gamma_2 \vdash \psi}$$

$$\frac{\Gamma_0 \vdash \varphi_1 \vee \varphi_2 \quad \Gamma_1, \varphi_1 \vdash \psi \quad \Gamma_2, \varphi_2 \vdash \psi}{\Gamma \vdash \psi}$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash \varphi}$$

$\neg\varphi$ is a macro for $\varphi \Rightarrow \bot$

$$\frac{\Gamma \vdash \neg\neg\varphi}{\Gamma \vdash \varphi}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\varphi,\ \varphi\Rightarrow\psi\vdash\varphi \qquad \varphi,\ \varphi\Rightarrow\psi\vdash\varphi\Rightarrow\psi}{\varphi,\ \varphi\Rightarrow\psi\vdash\psi} \qquad \psi\Rightarrow\eta\vdash\psi\Rightarrow\eta}{\varphi,\ \varphi\Rightarrow\psi,\ \psi\Rightarrow\eta\vdash\eta}}{\varphi\Rightarrow\psi,\ \psi\Rightarrow\eta\vdash\varphi\Rightarrow\eta}}{\varphi\Rightarrow\psi\vdash(\psi\Rightarrow\eta)\Rightarrow(\varphi\Rightarrow\eta)}}{\vdash(\varphi\Rightarrow\psi)\Rightarrow((\psi\Rightarrow\eta)\Rightarrow(\varphi\Rightarrow\eta))}$$

A PROOF is a tree whose leaves are axioms, every node is a rule instance & there is a conclusion of the form $\Gamma\vdash\varphi$.

We say $\Gamma\vdash\varphi$ is <u>derivable</u>

If Γ is empty we say $\cancel{\Gamma}\vdash\varphi$ is a <u>theorem</u>.

<u>SEMANTICS or MODELS</u>

We interpret a logic in ~~an~~ formal structure.

Prop logic $\rightarrow$ boolean algebra

$\Omega=\{T,F\}$ truth values $\quad \wedge, \vee, \Rightarrow, \neg$ are given by truth tables.

A <u>valuation</u> is an assignment of truth values to ~~formulas~~ propositional vars.

$p\mapsto T,\ q\mapsto F,\ r\mapsto T, \ldots$

$\vartheta : \mathrm{Prop}\rightarrow\Omega$

$\vartheta(\varphi)$ is defined by induction on the structure of φ

$$\vartheta(\varphi_1\wedge\varphi_2) = \vartheta(\varphi_1)\wedge\vartheta(\varphi_2)$$

notation in the logic $\qquad$ Boolean alg op.

similarly for the other formulas.

$\Gamma\vDash\varphi$ means for every ϑ s.t. $\forall\psi\in\Gamma$ $\vartheta(\psi)=T$ implies $\vartheta(\varphi)=T$.

If $\models \varphi$ then $\forall \vartheta \;\; \vartheta(\varphi) = T$

such a φ is called a tautology.

<u>Thm</u> If $\Gamma \vdash \varphi$ is derivable then $\Gamma \models \varphi$
(SOUNDNESS)

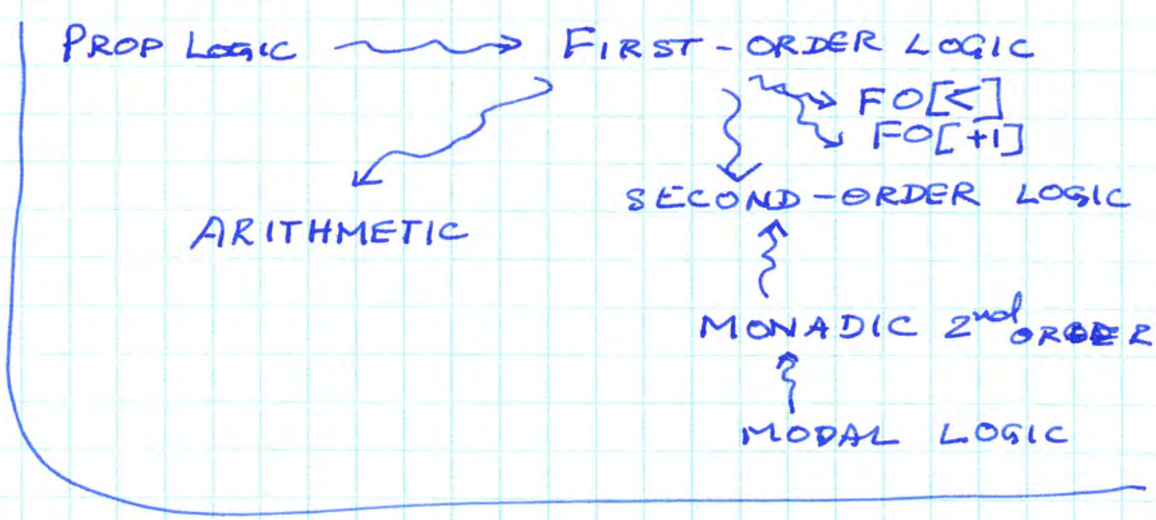
<u>Thm</u> If $\Gamma \models \varphi$ then $\Gamma \vdash \varphi$
COMPLETENESS.

Prop Logic satisfies soundness & completeness.

One can extend this to first-order logic.

~~Thm~~ ~~Def~~ If there is no ϑ s.t. $\forall \varphi \in \Gamma$ $\vartheta(\varphi) = T$ we say Γ is <u>unsatisfiable</u>.

~~If $\forall \vartheta, \forall \varphi \in \Gamma$~~ otherwise we say Γ is satisfiable.



<u>TEMPORAL LOGIC</u>

Syntax : $\text{PROP} = \{p, q, \cdots\}$

FORMULAS : BOOLEAN CONNECTIVE, $\bigcirc, (\Diamond, \Box,)\ \mathcal{U}$

$\varphi ::= \text{true} \mid p \mid \varphi_1 \wedge \varphi_2 \mid \neg \varphi \mid \bigcirc \varphi \mid \varphi_1 \mathcal{U} \varphi_2$

Defined operators

$\varphi_1 \vee \varphi_2$, $\varphi_1 \Rightarrow \varphi_2$

$\Diamond \varphi := \text{true}\ \mathcal{U}\ \varphi$ <u>eventually</u>

$\Box \varphi := \neg \Diamond \neg \varphi$ <u>always</u>

$\Box \Diamond \varphi$ infinitely often

$\Diamond \Box \varphi$ after some time forever

Key idea of LTL: the values of prop vars <u>change in time</u> so p may be true <u>now</u> but may become false later on.

So instead of valuations we work with sequences of valuations, now we call the valuations <u>states</u>.

So we have a sequence of ~~histories~~ states a <u>history</u>.

e.g. $(p, q) \to (p, \neg q) \to (p, \neg q) \to (\neg p, q) \to \cdots$

We write σ for such a history $\sigma[0]\,\sigma[1]\,\sigma[2]\ldots$

$\sigma[i..] = \sigma[i]\,\sigma[i+1]\,\sigma[i+2]\ldots$

$\sigma \models \text{true}$ $\quad$ $\sigma \models \bigcirc\varphi$ iff $\sigma[1..] \models \varphi$

$\sigma \models p$ if $\sigma[0] \models p$

$\sigma \models \varphi_1 \wedge \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$

$\sigma \models \neg\varphi$ iff $\sigma \not\models \varphi$

$\sigma \models \varphi \,\mathcal{U}\, \psi$ if $\exists j \geq 0\ \sigma[j..] \models \psi \ \&\ \forall i<j\ \sigma[i..] \models \varphi$

$\sigma \models \Diamond\varphi$ iff $\exists j \geq 0$ s.t. $\sigma[j..] \models \varphi$

$\sigma \models \Box\varphi$ iff $\forall j \geq 0$ $\sigma[j..] \models \varphi$

$\sigma \models \Box\Diamond\varphi$ iff $\forall i \geq 0$ $\sigma[i..] \models \Diamond\varphi$

iff $\forall i \geq 0$ $\exists j \geq i$ s.t $\sigma[j..] \models \varphi$

TRANSITION SYSTEMS $TS = (S, Act, \rightarrow, I, AP, L)$

$AP \rightarrow$ atomic propositions $I \subseteq S$ initial states
Act actions $\rightarrow$ transition $s \xrightarrow{a} s'$, $L : S \rightarrow 2^{AP}$
path : a sequence of states
run : $s_0 a_0 s_1 a_1 s_2 a_2 \cdots$
($\pi := s_0 s_1 s_2 \cdots$ s.t $s_0 \in I$) $\rightarrow$ ignore the actions

$\forall i$ $s_i \xrightarrow{a_i} s_{i+1}$

$trace(\pi) = L(s_0) L(s_1) L(s_2) \cdots$

run $\xrightarrow[act]{forget}$ path $\xrightarrow[state]{forget}$ sequence of 2^{AP}.

Given an LTS we say φ an LTL formula

$\pi \models \varphi$ if $trace(\pi) \models \varphi$

$s \cancel{\models \varphi \text{ if } \forall \pi \text{ s.t. } \pi = s \cdots}$
We say $Paths(s) = \{\pi \mid \pi = s \cdots\}$

$s \models \varphi$ if $\forall \pi \in Paths(s)$ $\pi \models \varphi$

LAWS FOR LTL

$\neg \bigcirc\varphi = \bigcirc\neg\varphi$

$\neg\Diamond\varphi = \Box\neg\varphi$

$\neg\Box\varphi = \Diamond\neg\varphi$

$\Diamond\Diamond\varphi \equiv \Diamond\varphi$

$\Box\Box\varphi \equiv \Box\varphi$

$\varphi U(\varphi U \psi) = \varphi U \psi$

$(\varphi U \psi) U \psi = \varphi U \psi$

$\Diamond\Box\Diamond\varphi \equiv \Box\Diamond\varphi$

$\Box\Diamond\Box\varphi \equiv \Diamond\Box\varphi$

$$\left.\begin{aligned} \varphi \mathbin{U} \psi &= \psi \lor (\varphi \land \bigcirc(\varphi \mathbin{U} \psi)) \\ \Diamond\varphi &= \varphi \lor \bigcirc\Diamond\varphi \\ \Box\varphi &= \varphi \land \bigcirc\Box\varphi \end{aligned}\right\} \text{fixed point laws}$$

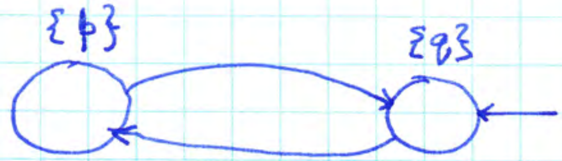
$$\bigcirc(\varphi \mathbin{U} \psi) \equiv \bigcirc\varphi \mathbin{U} \bigcirc\psi$$

$$\Diamond(\varphi \lor \psi) \equiv \Diamond\varphi \lor \Diamond\psi$$

$$\Box(\varphi \land \psi) \equiv \Box\varphi \land \Box\psi$$

<u>NOTE</u> $\Diamond(\varphi \land \psi) \not\equiv \Diamond\varphi \land \Diamond\psi$ <u>why not</u>?

$\Box(\varphi \lor \psi) \not\equiv \Box\varphi \lor \Box\psi$.



$TS \not\models \Diamond(p \land q)$ $\quad TS \models \Diamond p \land \Diamond q$

<u>EXAMPLES</u> $\Box(\text{red} \Rightarrow \Diamond\text{green})$.

TRAFFIC LIGHTS $\Box(\text{red} \Rightarrow \neg\bigcirc\text{green})$

$\Box(\text{red} \Rightarrow \bigcirc(\text{red} \mathbin{U} (\text{yellow} \land \bigcirc(\text{yellow} \mathbin{U} \text{green}))))$
