Formal semantics: Interpretation over transition systems

Recall:

$$\text{Words}(\varphi) = \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \varphi \}$$

We now want to interpret LTL f.las over transition systems. An obvious way is to first interpret LTL over paths and states.

π infinite path fragment of TS

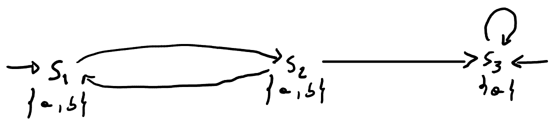
$$\pi \models \varphi \iff \text{trace}(\pi) \models \varphi$$
$$\iff \text{trace}(\pi) \in \text{Words}(\varphi)$$

Hence we can define

$$s \models \varphi \iff \forall \pi \in \text{Path}(s) : \pi \models \varphi \qquad \text{where } s \in S$$

And finally

$$TS \models \varphi \iff \forall s \in I : s \models \varphi$$
$$\iff TS \models \text{Words}(\varphi)$$

Exercise Let $AP = \{a, b\}$ and $TS =$ 

a) Is TS deterministic?

b) $s_1 \models \bigcirc(a \wedge b)$?

c) $s_2 \models \bigcirc(a \wedge b)$?

d) $TS \models \Box a$?

e) $TS \models \Box(\neg b \rightarrow a)$?

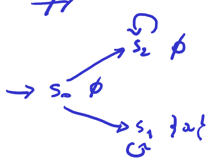
A note on negation

$$\text{Words}(\neg\varphi) = (2^{AP})^\omega \setminus \text{Words}(\varphi) \qquad \text{hence} \qquad \pi \models \varphi \iff \pi \not\models \neg\varphi$$

However negation is weird

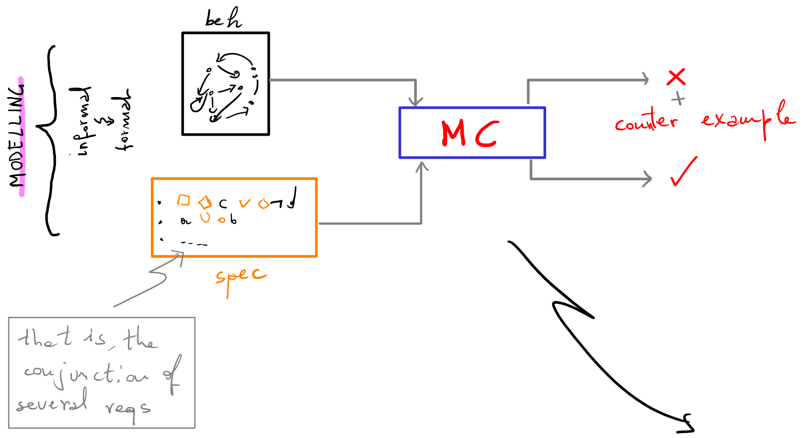
Exercise Show that $TS \not\models \varphi \not\Leftrightarrow TS \models \neg\varphi$

$\Leftarrow$ holds, but $\not\Rightarrow$

Take $TS =$  then $TS \not\models \Diamond a$ because of $s_0 s_2^\omega$
&
$TS \not\models \neg\Diamond a$ because of $s_0 s_1^\omega$

Model checking LTL (brief note)

Basic Algorithm (Vardi, Wolper 1986)



that is, the conjunction of several reqs

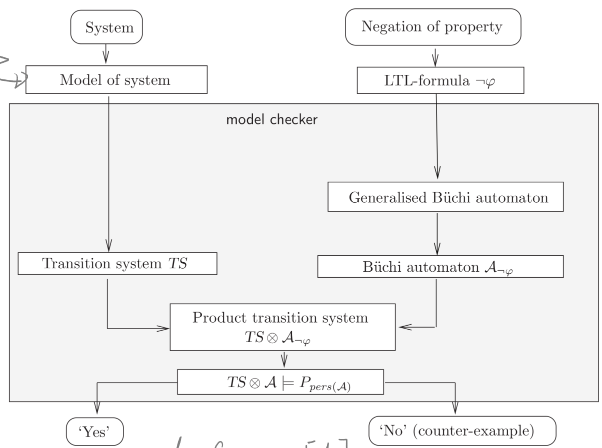
usually expressed in some DSL

LTL f.la

$TS \models \varphi \Leftrightarrow Traces(TS) \subseteq Words(\varphi)$

$\Leftrightarrow Traces(TS) \cap ((2^{AP})^\omega \setminus Words(\varphi)) = \emptyset$

$\Leftrightarrow Traces(TS) \cap Words(\neg\varphi) = \emptyset$



Borrowed from [1]

Figure 5.16: Overview of LTL model checking.

Input: φ LTL f.la & TS finite with no terminal states

Output: "yes" if $TS \models \varphi$, otherwise c. ex.

$\mathcal{A}_{\neg\varphi}$:= NBA s.t. $\mathcal{L}(\mathcal{A}_{\neg\varphi}) = Words(\neg\varphi)$

Thm: φ LTL f.la $\Rightarrow \varphi$ ω-regular $\Leftrightarrow \exists$ NBA accepting φ

state explosion

$\mathcal{A} := TS \times \mathcal{A}_{\neg\varphi}$

if $\exists \pi \in paths(\mathcal{A})$: π satisfies the accepting conditions of $\mathcal{A}$

  then return (expressive) bad prefix of π

  else return "yes"

fi

emptiness of NBA $\Leftrightarrow$ ∃ reachable accepting state on a cycle $\Downarrow$ emptiness of NBA is decidable

the actual procedure is more complex, but we do not consider the details due to

- choosing "good" counterexamples
- account for fairness
  - find $\varphi_{fair} \rightarrow \varphi$
  - algorithm to consider fair executions only