CS 228 : Logic in Computer Science

Krishna. S

So Far

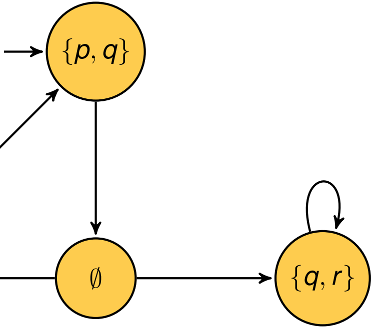
- Transition systems as models of actual systems
- Traces of transition systems capture behaviours of the system
- All traces are infinite

Transition Systems

A Transition System is a tuple $(S, Act, \to, I, AP, L)$ where

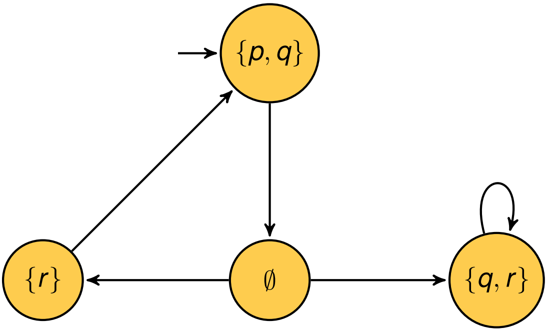
- S is a set of states
- Act is a set of actions
- $s \xrightarrow{\alpha} s'$ in $S \times Act \times S$ is the transition relation
- $I \subseteq S$ is the set of initial states
- AP is the set of atomic propositions
- $L : S \to 2^{AP}$ is the labeling function

Example Traces



$AP = \{p, q, r, t\}$

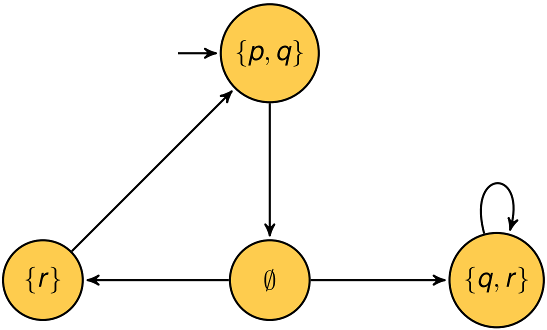
Example Traces



$AP = \{p, q, r, t\}$

- $\{p, q\}\emptyset\,\{q, r\}^{\omega}$

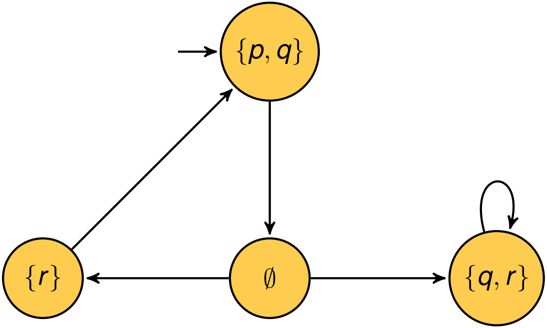
Example Traces



$AP = \{p, q, r, t\}$

- $\{p, q\}\emptyset\ \{q, r\}^\omega$
- $(\{p, q\}\emptyset\{r\})^\omega$

Example Traces



$AP = \{p, q, r, t\}$

- $\{p, q\}\emptyset\, \{q, r\}^{\omega}$
- $(\{p, q\}\emptyset\{r\})^{\omega}$
- $(\{p, q\}\emptyset\{r\})^{*}\, \{p, q\}\emptyset\, \{q, r\}^{\omega}$

Linear Time Properties

- Linear-time properties specify traces that a TS must have
- A LT property P over AP is a subset of $(2^{AP})^{\omega}$
- TS over AP satisfies a LT property P over AP

$$TS \models P \text{ iff } Traces(TS) \subseteq P$$

- $s \in S$ satisfies LT property P (denoted $s \models P$) iff $Traces(s) \subseteq P$

Syntax of Linear Temporal Logic

Given AP, a set of propositions,

Syntax of Linear Temporal Logic

Given *AP*, a set of propositions,

- Propositional logic formulae over *AP*
  - $a \in AP$ (atomic propositions)
  - $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$

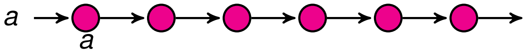
Syntax of Linear Temporal Logic

Given *AP*, a set of propositions,

- Propositional logic formulae over *AP*
  - $a \in AP$ (atomic propositions)
  - $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$
- Temporal Operators
  - $\bigcirc\varphi$ (Next φ)
  - $\varphi \, \mathsf{U} \psi$ (φ holds until a ψ-state is reached)
- LTL : Logic for describing LT properties

Semantics

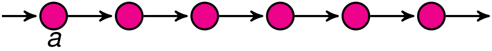
LTL formulae over AP interpreted over words over Σ^ω, $\Sigma = 2^{AP}$



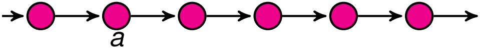
Semantics

LTL formulae over *AP* interpreted over words over Σ^ω, $\Sigma = 2^{AP}$

a

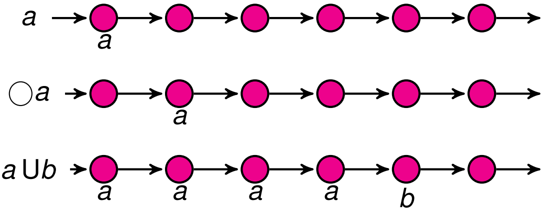


$\bigcirc a$



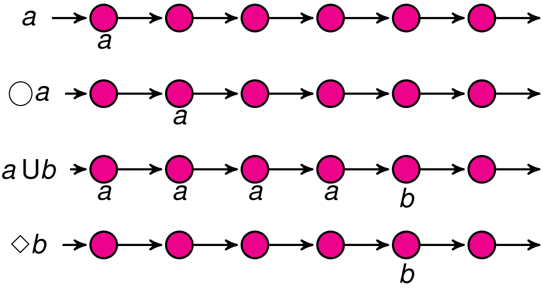
Semantics

LTL formulae over *AP* interpreted over words over Σ^ω, $\Sigma = 2^{AP}$



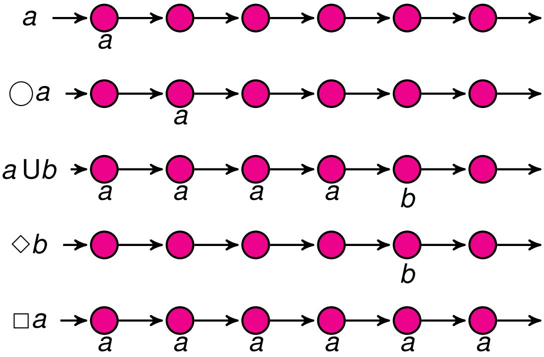
Semantics

LTL formulae over *AP* interpreted over words over Σ^ω, $\Sigma = 2^{AP}$



Semantics

LTL formulae over AP interpreted over words over Σ^ω, $\Sigma = 2^{AP}$



Derived Operators

- $\mathit{true} = \varphi \vee \neg\varphi$
- $\mathit{false} = \neg \mathit{true}$
- $\Diamond\varphi = \mathit{true}\, \mathrm{U} \varphi$ (Eventually φ)
- $\Box\varphi = \neg\Diamond\neg\varphi$ (Forever φ)

Precedence

- Unary Operators bind stronger than Binary
- $\bigcirc$ and $\neg$ equally strong
- U takes precedence over $\wedge, \vee, \rightarrow$
  - $a \vee b\, \mathrm{U} c \equiv a \vee (b\, \mathrm{U} c)$
  - $\bigcirc a\, \mathrm{U} \neg b \equiv (\bigcirc a)\, \mathrm{U} (\neg b)$

Examples

- Whenever the traffic light is red, it cannot become green immediately:

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\square(\mathit{red} \rightarrow \neg \bigcirc \mathit{green})$

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\Box(\mathit{red} \rightarrow \neg \bigcirc \mathit{green})$
- Eventually the traffic light will become yellow

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\square(\mathit{red} \rightarrow \neg \bigcirc \mathit{green})$
- Eventually the traffic light will become yellow
  $\diamond \mathit{yellow}$

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\Box(\mathit{red} \rightarrow \neg \bigcirc \mathit{green})$
- Eventually the traffic light will become yellow
  $\Diamond \mathit{yellow}$
- Once the traffic light becomes yellow, it will eventually become green

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\Box(\mathit{red} \to \neg \bigcirc \mathit{green})$
- Eventually the traffic light will become yellow
  $\Diamond \mathit{yellow}$
- Once the traffic light becomes yellow, it will eventually become green
  $\Box(\mathit{yellow} \to \Diamond \mathit{green})$

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\square(red \rightarrow \neg \bigcirc green)$
- Eventually the traffic light will become yellow
  $\Diamond yellow$
- Once the traffic light becomes yellow, it will eventually become green
  $\square(yellow \rightarrow \Diamond green)$
- Whenever the traffic light is red, it will eventually become green, but it must be yellow for sometime in between the red and the green

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\Box(red \rightarrow \neg \bigcirc green)$
- Eventually the traffic light will become yellow
  $\Diamond yellow$
- Once the traffic light becomes yellow, it will eventually become green
  $\Box(yellow \rightarrow \Diamond green)$
- Whenever the traffic light is red, it will eventually become green, but it must be yellow for sometime in between the red and the green
  $\Box(red \rightarrow \bigcirc(red\ \mathrm{U}[yellow$

Examples

- Whenever the traffic light is red, it cannot become green immediately:
  $\Box(\mathit{red} \rightarrow \neg \bigcirc \mathit{green})$
- Eventually the traffic light will become yellow
  $\Diamond \mathit{yellow}$
- Once the traffic light becomes yellow, it will eventually become green
  $\Box(\mathit{yellow} \rightarrow \Diamond \mathit{green})$
- Whenever the traffic light is red, it will eventually become green, but it must be yellow for sometime in between the red and the green
  $\Box(\mathit{red} \rightarrow \bigcirc(\mathit{red}\, \mathrm{U}[\mathit{yellow} \wedge \bigcirc (\mathit{yellow}\, \mathrm{U} \mathit{green})]))$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

- $\sigma \models a$ iff $a \in A_0$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

- $\sigma \models a$ iff $a \in A_0$
- $\sigma \models \varphi_1 \wedge \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

- $\sigma \models a$ iff $a \in A_0$
- $\sigma \models \varphi_1 \wedge \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$
- $\sigma \models \neg\varphi$ iff $\sigma \not\models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

- $\sigma \models a$ iff $a \in A_0$
- $\sigma \models \varphi_1 \land \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$
- $\sigma \models \neg\varphi$ iff $\sigma \not\models \varphi$
- $\sigma \models \bigcirc\varphi$ iff $A_1 A_2 \ldots \models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Let $\sigma = A_0 A_1 A_2 \ldots$.

- $\sigma \models a$ iff $a \in A_0$
- $\sigma \models \varphi_1 \wedge \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$
- $\sigma \models \neg\varphi$ iff $\sigma \not\models \varphi$
- $\sigma \models \bigcirc\varphi$ iff $A_1 A_2 \ldots \models \varphi$
- $\sigma \models \varphi \,\mathsf{U} \psi$ iff
  $\exists j \geqslant 0$ such that $A_j A_{j+1} \ldots \models \psi \wedge \forall 0 \leqslant i < j, A_i A_{i+1} \ldots \models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

Semantics over Infinite Words

Given LTL formula φ over AP,

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

- $\sigma \models \Diamond\varphi$ iff $\exists j \geqslant 0$, $A_j A_{j+1} \ldots \models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

- $\sigma \models \Diamond\varphi$ iff $\exists j \geqslant 0$, $A_j A_{j+1} \ldots \models \varphi$
- $\sigma \models \Box\varphi$ iff $\forall j \geqslant 0$, $A_j A_{j+1} \ldots \models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

- $\sigma \models \Diamond\varphi$ iff $\exists j \geqslant 0,\ A_j A_{j+1} \ldots \models \varphi$
- $\sigma \models \Box\varphi$ iff $\forall j \geqslant 0,\ A_j A_{j+1} \ldots \models \varphi$
- $\sigma \models \Box\Diamond\varphi$ iff $\forall j \geqslant 0,\ \exists i \geqslant j,\ A_i A_{i+1} \ldots \models \varphi$

Semantics over Infinite Words

Given LTL formula φ over AP,

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

- $\sigma \models \Diamond\varphi$ iff $\exists j \geqslant 0$, $A_j A_{j+1} \ldots \models \varphi$
- $\sigma \models \Box\varphi$ iff $\forall j \geqslant 0$, $A_j A_{j+1} \ldots \models \varphi$
- $\sigma \models \Box\Diamond\varphi$ iff $\forall j \geqslant 0$, $\exists i \geqslant j$, $A_i A_{i+1} \ldots \models \varphi$
- $\sigma \models \Diamond\Box\varphi$ iff $\exists j \geqslant 0$, $\forall i \geqslant j$, $A_i A_{i+1} \ldots \models \varphi$

If $\sigma = A_0 A_1 A_2 \ldots$, $\sigma \models \varphi$ is also written as $\sigma, 0 \models \varphi$. This simply means $A_0 A_1 A_2 \ldots \models \varphi$. One can also define $\sigma, i \models \varphi$ to mean $A_i A_{i+1} A_{i+2} \ldots \models \varphi$ to talk about a suffix of the word σ satisfying a property.