Linear Temporal Logic (LTL)

- Grammar of *well formed formulae* (wff) ϕ

$$
\begin{array}{lcll}
\phi & ::= & p & \text{(Atomic formula: } p \in AP\text{)} \\
 & | & \neg\phi & \text{(Negation)} \\
 & | & \phi_1 \vee \phi_2 & \text{(Disjunction)} \\
 & | & \mathbf{X}\phi & \text{(successor)} \\
 & | & \mathbf{F}\phi & \text{(sometimes)} \\
 & | & \mathbf{G}\phi & \text{(always)} \\
 & | & [\phi_1\ \mathbf{U}\ \phi_2] & \text{(Until)}
\end{array}
$$

- Details differ from Prior's tense logic – but similar ideas
- Semantics define when ϕ true in model M
  - where $M = (S, S_0, R, L)$ – a Kripke structure
  - notation: $M \models \phi$ means ϕ true in model M
  - model checking algorithms compute this (when decidable)

$M \models \phi$ means "wff ϕ is true in model M"

- If $M = (S, S_0, R, L)$ then

  $\boxed{\pi \text{ is an } M\text{-path starting from } s \text{ iff } \mathsf{Path}\ R\ s\ \pi}$

- If $M = (S, S_0, R, L)$ then we define $M \models \phi$ to mean:

  $\boxed{\phi \text{ is true on all } M\text{-paths starting from a member of } S_0}$

- We will define $[\![\phi]\!]_M(\pi)$ to mean

  $\boxed{\phi \text{ is true on the } M\text{-path } \pi}$

- Thus $M \models \phi$ will be formally defined by:

  $\boxed{M \models \phi \ \Leftrightarrow\ \forall \pi\ s.\ s \in S_0 \wedge \mathsf{Path}\ R\ s\ \pi \Rightarrow [\![\phi]\!]_M(\pi)}$

- It remains to actually define $[\![\phi]\!]_M$ for all wffs ϕ

Definition of $[\![\phi]\!]_M(\pi)$

- $[\![\phi]\!]_M(\pi)$ is the application of function $[\![\phi]\!]_M$ to path π
  - thus $[\![\phi]\!]_M : (\mathbb{N} \to S) \to \mathbb{B}$
- Let $M = (S, S_0, R, L)$

  $[\![\phi]\!]_M$ is defined by structural induction on ϕ

$$
\begin{array}{lcl}
[\![p]\!]_M(\pi) & = & p \in L(\pi\ 0) \\
[\![\neg\phi]\!]_M(\pi) & = & \neg([\![\phi]\!]_M(\pi)) \\
[\![\phi_1 \vee \phi_2]\!]_M(\pi) & = & [\![\phi_1]\!]_M(\pi) \ \vee \ [\![\phi_2]\!]_M(\pi) \\
[\![\mathbf{X}\phi]\!]_M(\pi) & = & [\![\phi]\!]_M(\pi{\downarrow}1) \\
[\![\mathbf{F}\phi]\!]_M(\pi) & = & \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \\
[\![\mathbf{G}\phi]\!]_M(\pi) & = & \forall i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \\
[\![[\phi_1\ \mathbf{U}\ \phi_2]]\!]_M(\pi) & = & \exists i.\ [\![\phi_2]\!]_M(\pi{\downarrow}i) \ \wedge \ \forall j.\ j{<}i \ \Rightarrow \ [\![\phi_1]\!]_M(\pi{\downarrow}j)
\end{array}
$$

- We look at each of these semantic equations in turn

$[\![p]\!]_M(\pi) = p(\pi\ 0)$

- Assume $M = (S, S_0, R, L)$
- We have: $[\![p]\!]_M(\pi) = p \in L(\pi\ 0)$
  - p is an atomic property, i.e. $p \in AP$
  - $\pi : \mathbb{N} \to S$ so $\pi\ 0 \in S$
  - $\pi\ 0$ is the first state in path π
  - $p \in L(\pi\ 0)$ is *true* iff atomic property p holds of state $\pi\ 0$
- $[\![p]\!]_M(\pi)$ means p holds of the first state in path π
- $\mathtt{T}, \mathtt{F} \in AP$ with $\mathtt{T} \in L(s)$ and $\mathtt{F} \notin L(s)$ for all $s \in S$
  - $[\![\mathtt{T}]\!]_M(\pi)$ is always true
  - $[\![\mathtt{F}]\!]_M(\pi)$ is always false

$$[\![\neg\phi]\!]_M(\pi) = \neg([\![\phi]\!]_M(\pi))$$
$$[\![\phi_1 \vee \phi_2]\!]_M(\pi) = [\![\phi_1]\!]_M(\pi) \vee [\![\phi_2]\!]_M(\pi)$$

- $[\![\neg\phi]\!]_M(\pi) = \neg([\![\phi]\!]_M(\pi))$
  - $[\![\neg\phi]\!]_M(\pi)$ true iff $[\![\phi]\!]_M(\pi)$ is not true

- $[\![\phi_1 \vee \phi_2]\!]_M(\pi) = [\![\phi_1]\!]_M(\pi) \vee [\![\phi_2]\!]_M(\pi)$
  - $[\![\phi_1 \vee \phi_2]\!]_M(\pi)$ true iff $[\![\phi_1]\!]_M(\pi)$ is true or $[\![\phi_2]\!]_M(\pi)$ is true

$[\![\mathbf{X}\phi]\!]_M(\pi) = [\![\phi]\!]_M(\pi{\downarrow}1)$

- $[\![\mathbf{X}\phi]\!]_M(\pi) = [\![\phi]\!]_M(\pi{\downarrow}1)$
  - $\pi{\downarrow}1$ is π with the first state chopped off

    $$\begin{array}{l} \pi{\downarrow}1(0) = \pi(1+0) = \pi(1) \\ \pi{\downarrow}1(1) = \pi(1+1) = \pi(2) \\ \pi{\downarrow}1(2) = \pi(1+2) = \pi(3) \\ \qquad\qquad \vdots \end{array}$$

- $[\![\mathbf{X}\phi]\!]_M(\pi)$ true iff $[\![\phi]\!]_M$ true *starting at the second state of* π

$[\![\mathbf{F}\phi]\!]_M(\pi) = \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i)$

- $[\![\mathbf{F}\phi]\!]_M(\pi) = \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i)$
  - $\pi{\downarrow}i$ is π with the first i states chopped off

    $$\begin{aligned} \pi{\downarrow}i(0) &= \pi(i+0) = \pi(i) \\ \pi{\downarrow}i(1) &= \pi(i+1) \\ \pi{\downarrow}i(2) &= \pi(i+2) \\ &\vdots \end{aligned}$$
  - $[\![\phi]\!]_M(\pi{\downarrow}i)$ true iff $[\![\phi]\!]_M$ true *starting i states along* π

- $[\![\mathbf{F}\phi]\!]_M(\pi)$ true iff $[\![\phi]\!]_M$ true *starting somewhere along* π

- "$\mathbf{F}\phi$" is read as "sometimes ϕ"

$[\![\mathbf{G}\phi]\!]_M(\pi) = \forall i.\ [\![\phi]\!]_M(\pi{\downarrow}i)$

- $[\![\mathbf{G}\phi]\!]_M(\pi) = \forall i.\ [\![\phi]\!]_M(\pi{\downarrow}i)$
  - $\pi{\downarrow}i$ is π with the first i states chopped off
  - $[\![\phi]\!]_M(\pi{\downarrow}i)$ true iff $[\![\phi]\!]_M$ true *starting i states along* π
- $[\![\mathbf{G}\phi]\!]_M(\pi)$ true iff $[\![\phi]\!]_M$ true *starting anywhere along* π
- "$\mathbf{G}\phi$" is read as "always ϕ" or "globally ϕ"
- $M \models \mathbf{AG}\,p$ defined earlier: $M \models \mathbf{AG}\,p \Leftrightarrow M \models \mathbf{G}(p)$
- $\mathbf{G}$ is definable in terms of $\mathbf{F}$ and $\neg$: $\mathbf{G}\phi = \neg(\mathbf{F}(\neg\phi))$

$$\begin{array}{rcl}
[\![\neg(\mathbf{F}(\neg\phi))]\!]_M(\pi) & = & \neg([\![\mathbf{F}(\neg\phi)]\!]_M(\pi)) \\
& = & \neg(\exists i.\ [\![\neg\phi]\!]_M(\pi{\downarrow}i)) \\
& = & \neg(\exists i.\ \neg([\![\phi]\!]_M(\pi{\downarrow}i))) \\
& = & \forall i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \\
& = & [\![\mathbf{G}\phi]\!]_M(\pi)
\end{array}$$

$$\boxed{[\![[\phi_1\ \mathbf{U}\ \phi_2]]\!]_M(\pi) = \ \exists i.\ [\![\phi_2]\!]_M(\pi{\downarrow}i) \wedge \forall j.\ j{<}i \Rightarrow [\![\phi_1]\!]_M(\pi{\downarrow}j)}$$

- $[\![[\phi_1\ \mathbf{U}\ \phi_2]]\!]_M(\pi) = \exists i.\ [\![\phi_2]\!]_M(\pi{\downarrow}i) \wedge \forall j.\ j{<}i \Rightarrow [\![\phi_1]\!]_M(\pi{\downarrow}j)$
  - $[\![\phi_2]\!]_M(\pi{\downarrow}i)$ true iff $[\![\phi_2]\!]_M$ true *starting i states along π*
  - $[\![\phi_1]\!]_M(\pi{\downarrow}j)$ true iff $[\![\phi_1]\!]_M$ true *starting j states along π*
- $[\![[\phi_1\ \mathbf{U}\ \phi_2]]\!]_M(\pi)$ is true iff

  $[\![\phi_2]\!]_M$ is true somewhere along π and up to then $[\![\phi_1]\!]_M$ is true
- "$[\phi_1\ \mathbf{U}\ \phi_2]$" is read as "$\phi_1$ until ϕ_2"
- **F** is definable in terms of $[-\ \mathbf{U}\ -]$: $\mathbf{F}\phi = [\mathtt{T}\ \mathbf{U}\ \phi]$

$$\begin{aligned}
&[\![[\mathtt{T}\ \mathbf{U}\ \phi]]\!]_M(\pi) \\
&= \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \wedge \forall j.\ j{<}i \Rightarrow [\![\mathtt{T}]\!]_M(\pi{\downarrow}j) \\
&= \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \wedge \forall j.\ j{<}i \Rightarrow \mathit{true} \\
&= \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \wedge \mathit{true} \\
&= \exists i.\ [\![\phi]\!]_M(\pi{\downarrow}i) \\
&= [\![\mathbf{F}\phi]\!]_M(\pi)
\end{aligned}$$

Review of Linear Temporal Logic (LTL)

- Grammar of *well formed formulae* (wff) ϕ

$$
\begin{array}{lll}
\phi ::= & p & \text{(Atomic formula: } p \in AP\text{)} \\
\mid & \neg\phi & \text{(Negation)} \\
\mid & \phi_1 \vee \phi_2 & \text{(Disjunction)} \\
\mid & \mathbf{X}\phi & \text{(successor)} \\
\mid & \mathbf{F}\phi & \text{(sometimes)} \\
\mid & \mathbf{G}\phi & \text{(always)} \\
\mid & [\phi_1\ \mathbf{U}\ \phi_2] & \text{(Until)}
\end{array}
$$

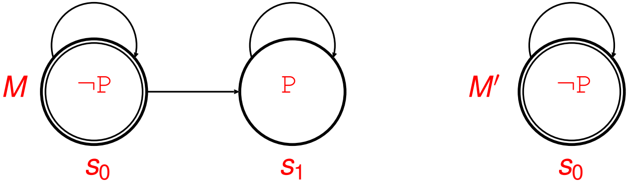
- $M \models \phi$ means ϕ holds on all M-paths
  - $M = (S, S_0, R, L)$
  - $[\![\phi]\!]_M(\pi)$ means ϕ is true on the M-path π
  - $M \models \phi \;\Leftrightarrow\; \forall\pi\ s.\ s \in S_0 \wedge \mathrm{Path}\ R\ s\ \pi \Rightarrow [\![\phi]\!]_M(\pi)$

LTL examples

- "DeviceEnabled holds infinitely often along every path"
  $\boxed{\mathbf{G}(\mathbf{F}\ \texttt{DeviceEnabled})}$
- "Eventually the state becomes permanently Done"
  $\boxed{\mathbf{F}(\mathbf{G}\ \texttt{Done})}$
- "Every Req is followed by an Ack"
  $\boxed{\mathbf{G}(\texttt{Req} \Rightarrow \mathbf{F}\ \texttt{Ack})}$
  Number of Req and Ack may differ - no counting
- "If Enabled infinitely often then Running infinitely often"
  $\boxed{\mathbf{G}(\mathbf{F}\ \texttt{Enabled}) \Rightarrow \mathbf{G}(\mathbf{F}\ \texttt{Running})}$
- "An upward going lift at the second floor keeps going up if a passenger requests the fifth floor"
  $$\boxed{\begin{array}{l}\mathbf{G}(\texttt{AtFloor2} \wedge \texttt{DirectionUp} \wedge \texttt{RequestFloor5} \\ \quad \Rightarrow [\texttt{DirectionUp}\ \mathbf{U}\ \texttt{AtFloor5}])\end{array}}$$

A property not expressible in LTL

- Let $AP = \{\mathtt{P}\}$ and consider models M and M' below



$$\begin{array}{lcl} M & = & (\{s_0, s_1\}, \{s_0\}, \{(s_0, s_0), (s_0, s_1), (s_1, s_1)\}, L) \\ M' & = & (\{s_0\}, \{s_0\}, \{(s_0, s_0)\}, L) \end{array}$$

where: $L \;=\; \lambda s.\ \mathit{if}\ s = s_0\ \mathit{then}\ \{\}\ \mathit{else}\ \{\mathtt{P}\}$

- Every M'-path is also an M-path
- So if ϕ true on every M-path then ϕ true on every M'-path
- Hence in LTL for any ϕ if $M \models \phi$ then $M' \models \phi$
- Consider $\phi_{\mathtt{P}} \Leftrightarrow$ "can always reach a state satisfying $\mathtt{P}$"
  - $\phi_{\mathtt{P}}$ holds in M but not in M'
  - but in LTL can't have $M \models \phi_{\mathtt{P}}$ and not $M' \models \phi_{\mathtt{P}}$
- hence $\phi_{\mathtt{P}}$ not expressible in LTL

LTL expressibility

"can always reach a state satisfying `p`"

- In LTL $M \models \phi$ says ϕ holds of all paths of M
- LTL formulae ϕ are evaluated on paths .... path formulae
- Want to say that from any state there exists a path to some state satisfying p
  - $\forall s.\ \exists \pi.\ \mathrm{Path}\ R\ s\ \pi\ \wedge\ \exists i.\ p \in L(\pi(i))$
  - but this isn't expressible in LTL (see slide 57)
- CTL properties are evaluated at a state ... state formulae
  - they can talk about both some or all paths
  - starting from the state they are evaluated at

Computation Tree Logic (CTL)

- LTL formulae ϕ are evaluated on paths .... path formulae
- CTL formulae ψ are evaluated on states .. state formulae

- Syntax of CTL well-formed formulae:

$$
\begin{array}{rcll}
\psi & ::= & p & \text{(Atomic formula } p \in AP\text{)} \\
& | & \neg\psi & \text{(Negation)} \\
& | & \psi_1 \wedge \psi_2 & \text{(Conjunction)} \\
& | & \psi_1 \vee \psi_2 & \text{(Disjunction)} \\
& | & \psi_1 \Rightarrow \psi_2 & \text{(Implication)} \\
& | & \mathbf{AX}\psi & \text{(All successors)} \\
& | & \mathbf{EX}\psi & \text{(Some successors)} \\
& | & \mathbf{A}[\psi_1 \ \mathbf{U} \ \psi_2] & \text{(Until – along all paths)} \\
& | & \mathbf{E}[\psi_1 \ \mathbf{U} \ \psi_2] & \text{(Until – along some path)}
\end{array}
$$

Semantics of CTL

- Assume $M = (S, S_0, R, L)$ and then define:

$$
\begin{array}{lcl}
[\![p]\!]_M(s) & = & p \in L(s) \\
[\![\neg\psi]\!]_M(s) & = & \neg([\![\psi]\!]_M(s)) \\
[\![\psi_1 \wedge \psi_2]\!]_M(s) & = & [\![\psi_1]\!]_M(s) \ \wedge \ [\![\psi_2]\!]_M(s) \\
[\![\psi_1 \vee \psi_2]\!]_M(s) & = & [\![\psi_1]\!]_M(s) \ \vee \ [\![\psi_2]\!]_M(s) \\
[\![\psi_1 \Rightarrow \psi_2]\!]_M(s) & = & [\![\psi_1]\!]_M(s) \ \Rightarrow \ [\![\psi_2]\!]_M(s) \\
[\![\mathbf{AX}\psi]\!]_M(s) & = & \forall s'.\ R\ s\ s' \ \Rightarrow \ [\![\psi]\!]_M(s') \\
[\![\mathbf{EX}\psi]\!]_M(s) & = & \exists s'.\ R\ s\ s' \ \wedge \ [\![\psi]\!]_M(s') \\
[\![\mathbf{A}[\psi_1\ \mathbf{U}\ \psi_2]]\!]_M(s) & = & \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \\
 & & \quad \Rightarrow \exists i.\ [\![\psi_2]\!]_M(\pi(i)) \\
 & & \qquad\qquad \wedge \\
 & & \qquad\qquad \forall j.\ j{<}i \ \Rightarrow \ [\![\psi_1]\!]_M(\pi(j)) \\
[\![\mathbf{E}[\psi_1\ \mathbf{U}\ \psi_2]]\!]_M(s) & = & \exists \pi.\ \mathrm{Path}\ R\ s\ \pi \\
 & & \quad \wedge \ \exists i.\ [\![\psi_2]\!]_M(\pi(i)) \\
 & & \qquad\qquad \wedge \\
 & & \qquad\qquad \forall j.\ j{<}i \ \Rightarrow \ [\![\psi_1]\!]_M(\pi(j))
\end{array}
$$

The defined operator **AF**

- Define $\mathbf{AF}\psi = \mathbf{A}[\top\ \mathbf{U}\ \psi]$
- $\mathbf{AF}\psi$ true at s iff ψ true somewhere on every R-path from s

$$
\begin{aligned}
[\![\mathbf{AF}\psi]\!]_M(s) &= [\![\mathbf{A}[\top\ \mathbf{U}\ \psi]]\!]_M(s) \\
&= \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \\
&\qquad \Rightarrow \\
&\qquad \exists i.\ [\![\psi]\!]_M(\pi(i)) \ \wedge\ \forall j.\ j < i \ \Rightarrow\ [\![\top]\!]_M(\pi(j)) \\
&= \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \\
&\qquad \Rightarrow \\
&\qquad \exists i.\ [\![\psi]\!]_M(\pi(i)) \ \wedge\ \forall j.\ j < i \ \Rightarrow\ \mathit{true} \\
&= \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \ \Rightarrow\ \exists i.\ [\![\psi]\!]_M(\pi(i))
\end{aligned}
$$

The defined operator EF

- Define $\mathbf{EF}\psi = \mathbf{E}[\mathtt{T}\ \mathbf{U}\ \psi]$
- $\mathbf{EF}\psi$ true at s iff ψ true somewhere on some R-path from s

$$
\begin{aligned}
[\![\mathbf{EF}\psi]\!]_M(s) &= [\![\mathbf{E}[\mathtt{T}\ \mathbf{U}\ \psi]]\!]_M(s) \\
&= \exists \pi.\ \mathrm{Path}\ R\ s\ \pi \\
&\qquad \wedge \\
&\qquad \exists i.\ [\![\psi]\!]_M(\pi(i))\ \wedge\ \forall j.\ j < i\ \Rightarrow\ [\![\mathtt{T}]\!]_M(\pi(j)) \\
&= \exists \pi.\ \mathrm{Path}\ R\ s\ \pi \\
&\qquad \wedge \\
&\qquad \exists i.\ [\![\psi]\!]_M(\pi(i))\ \wedge\ \forall j.\ j < i\ \Rightarrow\ \mathit{true} \\
&= \exists \pi.\ \mathrm{Path}\ R\ s\ \pi\ \wedge\ \exists i.\ [\![\psi]\!]_M(\pi(i))
\end{aligned}
$$

- "can reach a state satisfying p" is $\mathbf{EF}\ p$

The defined operator AG

- Define $\mathbf{AG}\psi = \neg\mathbf{EF}(\neg\psi)$
- $\mathbf{AG}\psi$ true at s iff ψ true everywhere on every R-path from s

$$
\begin{aligned}
[\![\mathbf{AG}\psi]\!]_M(s) &= [\![\neg\mathbf{EF}(\neg\psi)]\!]_M(s) \\
&= \neg([\![\mathbf{EF}(\neg\psi)]\!]_M(s)) \\
&= \neg(\exists\pi.\ \mathsf{Path}\ R\ s\ \pi \wedge \exists i.\ [\![\neg\psi]\!]_M(\pi(i))) \\
&= \neg(\exists\pi.\ \mathsf{Path}\ R\ s\ \pi \wedge \exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \forall\pi.\ \neg(\mathsf{Path}\ R\ s\ \pi \wedge \exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \forall\pi.\ \neg\mathsf{Path}\ R\ s\ \pi \vee \neg(\exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \forall\pi.\ \neg\mathsf{Path}\ R\ s\ \pi \vee \forall i.\ \neg\neg[\![\psi]\!]_M(\pi(i)) \\
&= \forall\pi.\ \neg\mathsf{Path}\ R\ s\ \pi \vee \forall i.\ [\![\psi]\!]_M(\pi(i)) \\
&= \forall\pi.\ \mathsf{Path}\ R\ s\ \pi \Rightarrow \forall i.\ [\![\psi]\!]_M(\pi(i))
\end{aligned}
$$

- $\mathbf{AG}\psi$ means ψ true at all reachable states
- $[\![\mathbf{AG}(p)]\!]_M(s)\ \equiv\ \forall s'.\ R^*\ s\ s'\ \Rightarrow\ p \in L(s')$
- "can always reach a state satisfying p" is $\mathbf{AG}(\mathbf{EF}\ p)$

The defined operator **EG**

- Define $\mathbf{EG}\psi = \neg\mathbf{AF}(\neg\psi)$

- $\mathbf{EG}\psi$ true at s iff ψ true everywhere on some R-path from s

$$
\begin{aligned}
[\![\mathbf{EG}\psi]\!]_M(s) &= [\![\neg\mathbf{AF}(\neg\psi)]\!]_M(s) \\
&= \neg([\![\mathbf{AF}(\neg\psi)]\!]_M(s)) \\
&= \neg(\forall\pi.\ \mathrm{Path}\ R\ s\ \pi \Rightarrow \exists i.\ [\![\neg\psi]\!]_M(\pi(i))) \\
&= \neg(\forall\pi.\ \mathrm{Path}\ R\ s\ \pi \Rightarrow \exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \exists\pi.\ \neg(\mathrm{Path}\ R\ s\ \pi \Rightarrow \exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \exists\pi.\ \mathrm{Path}\ R\ s\ \pi \wedge \neg(\exists i.\ \neg[\![\psi]\!]_M(\pi(i))) \\
&= \exists\pi.\ \mathrm{Path}\ R\ s\ \pi \wedge \forall i.\ \neg\neg[\![\psi]\!]_M(\pi(i)) \\
&= \exists\pi.\ \mathrm{Path}\ R\ s\ \pi \wedge \forall i.\ [\![\psi]\!]_M(\pi(i))
\end{aligned}
$$

The defined operator $\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]$

- $\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]$ is a 'partial correctness' version of $\mathbf{A}[\psi_1\ \mathbf{U}\ \psi_2]$
- It is true at s if along all R-paths from s:
  - ψ_1 always holds on the path, or
  - ψ_2 holds sometime on the path, and until it does ψ_1 holds
- Define

$$
\begin{array}{l}
[\![\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]]\!]_M(s) \\
\quad = [\![\neg\mathbf{E}[(\psi_1\wedge\neg\psi_2)\ \mathbf{U}\ (\neg\psi_1\wedge\neg\psi_2)]]\!]_M(s) \\
\quad = \neg[\![\mathbf{E}[(\psi_1\wedge\neg\psi_2)\ \mathbf{U}\ (\neg\psi_1\wedge\neg\psi_2)]]\!]_M(s) \\
\quad = \neg(\exists\pi.\ \mathrm{Path}\ R\ s\ \pi \\
\qquad\qquad\quad \wedge \\
\qquad\qquad\quad \exists i.\ [\![\neg\psi_1\wedge\neg\psi_2]\!]_M(\pi(i)) \\
\qquad\qquad\qquad \wedge \\
\qquad\qquad\qquad \forall j.\ j<i \ \Rightarrow\ [\![\psi_1\wedge\neg\psi_2]\!]_M(\pi(j)))
\end{array}
$$

- Exercise: understand the next two slides!

$\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]$ continued (1)

- Continuing:

$$
\begin{array}{ll}
 & \neg(\exists \pi.\ \mathrm{Path}\ R\ s\ \pi \\
 & \qquad \wedge \\
 & \qquad \exists i.\ [\![\neg\psi_1 \wedge \neg\psi_2]\!]_M(\pi(i))\ \wedge\ \forall j.\ j<i\ \Rightarrow\ [\![\psi_1 \wedge \neg\psi_2]\!]_M(\pi(j))) \\
= & \forall \pi.\ \neg(\mathrm{Path}\ R\ s\ \pi \\
 & \qquad \wedge \\
 & \qquad \exists i.\ [\![\neg\psi_1 \wedge \neg\psi_2]\!]_M(\pi(i))\ \wedge\ \forall j.\ j<i\ \Rightarrow\ [\![\psi_1 \wedge \neg\psi_2]\!]_M(\pi(j))) \\
= & \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \\
 & \qquad \Rightarrow \\
 & \qquad \neg(\exists i.\ [\![\neg\psi_1 \wedge \neg\psi_2]\!]_M(\pi(i))\ \wedge\ \forall j.\ j<i\ \Rightarrow\ [\![\psi_1 \wedge \neg\psi_2]\!]_M(\pi(j))) \\
= & \forall \pi.\ \mathrm{Path}\ R\ s\ \pi \\
 & \qquad \Rightarrow \\
 & \qquad \forall i.\ \neg[\![\neg\psi_1 \wedge \neg\psi_2]\!]_M(\pi(i)) \vee \neg(\forall j.\ j<i\ \Rightarrow\ [\![\psi_1 \wedge \neg\psi_2]\!]_M(\pi(j)))
\end{array}
$$

$\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]$ continued (2)

- Continuing:

$$= \forall\pi.\ \mathsf{Path}\ R\ s\ \pi \Rightarrow \forall i.\ \neg[\![\neg\psi_1\wedge\neg\psi_2]\!]_M(\pi(i)) \vee \neg(\forall j.\ j<i \ \Rightarrow\ [\![\psi_1\wedge\neg\psi_2]\!]_M(\pi(j)))$$

$$= \forall\pi.\ \mathsf{Path}\ R\ s\ \pi \Rightarrow \forall i.\ \neg(\forall j.\ j<i \ \Rightarrow\ [\![\psi_1\wedge\neg\psi_2]\!]_M(\pi(j))) \vee \neg[\![\neg\psi_1\wedge\neg\psi_2]\!]_M(\pi(i))$$

$$= \forall\pi.\ \mathsf{Path}\ R\ s\ \pi \Rightarrow \forall i.\ (\forall j.\ j<i \ \Rightarrow\ [\![\psi_1\wedge\neg\psi_2]\!]_M(\pi(j))) \ \Rightarrow\ [\![\psi_1\vee\psi_2]\!]_M(\pi(i))$$

- Exercise: explain why this is $[\![\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]]\!]_M(s)$?
  - this exercise illustrates the subtlety of writing CTL!

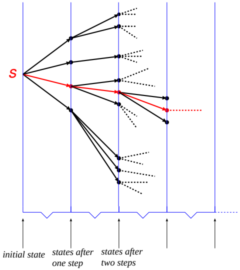
Sanity check: $\mathbf{A}[\psi\ \mathbf{W}\ \mathtt{F}]\ =\ \mathbf{AG}\ \psi$

- From last slide:
  $[\![\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]]\!]_M(s)$
  $= \ \forall\pi.\ \mathrm{Path}\ R\ s\ \pi$
  $\Rightarrow\ \forall i.\ (\forall j.\ j{<}i \Rightarrow [\![\psi_1{\wedge}\neg\psi_2]\!]_M(\pi(j))) \Rightarrow [\![\psi_1{\vee}\psi_2]\!]_M(\pi(i))$
- Set ψ_1 to ψ and ψ_2 to $\mathtt{F}$:
  $[\![\mathbf{A}[\psi\ \mathbf{W}\ \mathtt{F}]]\!]_M(s)$
  $= \ \forall\pi.\ \mathrm{Path}\ R\ s\ \pi$
  $\Rightarrow\ \forall i.\ (\forall j.\ j{<}i\ \Rightarrow\ [\![\psi{\wedge}\neg\mathtt{F}]\!]_M(\pi(j)))\ \Rightarrow\ [\![\psi{\vee}\mathtt{F}]\!]_M(\pi(i))$
- Simplify:
  $[\![\mathbf{A}[\psi\ \mathbf{W}\ \mathtt{F}]]\!]_M(s)$
  $= \ \forall\pi.\ \mathrm{Path}\ R\ s\ \pi \Rightarrow \forall i.\ (\forall j.\ j{<}i \Rightarrow [\![\psi]\!]_M(\pi(j))) \Rightarrow [\![\psi]\!]_M(\pi(i))$
- By induction on i:
  $[\![\mathbf{A}[\psi\ \mathbf{W}\ \mathtt{F}]]\!]_M(s)\ =\ \forall\pi.\ \mathrm{Path}\ R\ s\ \pi\ \Rightarrow\ \forall i.\ [\![\psi]\!]_M(\pi(i))$

- Exercises
  1. Describe the property: $\mathbf{A}[\mathtt{T}\ \mathbf{W}\ \psi]$.
  2. Describe the property: $\neg\mathbf{E}[\neg\psi_2\ \mathbf{U}\ \neg(\psi_1{\vee}\psi_2)]$.
  3. Define $\mathbf{E}[\psi_1\ \mathbf{W}\ \psi_2] = \mathbf{E}[\psi_1\ \mathbf{U}\ \psi_2] \vee \mathbf{EG}\psi_1$.
     Describe the property: $\mathbf{E}[\psi_1\ \mathbf{W}\ \psi_2]$?

Recall model behaviour computation tree

- Atomic properties are true or false of individual states
- General properties are true or false of whole behaviour
- Behaviour of (S, R) starting from $s \in S$ as a tree:



- A path is shown in red
- Properties may look at all paths, or just a single path
  - CTL: Computation Tree Logic (all paths from a state)
  - LTL: Linear Temporal Logic (a single path)

Summary of CTL operators (primitive + defined)

- CTL formulae:

p	(Atomic formula - $p \in AP$)
$\neg\psi$	(Negation)
$\psi_1 \wedge \psi_2$	(Conjunction)
$\psi_1 \vee \psi_2$	(Disjunction)
$\psi_1 \Rightarrow \psi_2$	(Implication)
$\mathbf{AX}\psi$	(All successors)
$\mathbf{EX}\psi$	(Some successors)
$\mathbf{AF}\psi$	(Somewhere – along all paths)
$\mathbf{EF}\psi$	(Somewhere – along some path)
$\mathbf{AG}\psi$	(Everywhere – along all paths)
$\mathbf{EG}\psi$	(Everywhere – along some path)
$\mathbf{A}[\psi_1\ \mathbf{U}\ \psi_2]$	(Until – along all paths)
$\mathbf{E}[\psi_1\ \mathbf{U}\ \psi_2]$	(Until – along some path)
$\mathbf{A}[\psi_1\ \mathbf{W}\ \psi_2]$	(Unless – along all paths)
$\mathbf{E}[\psi_1\ \mathbf{W}\ \psi_2]$	(Unless – along some path)

Example CTL formulae

- $\mathbf{EF}(\mathit{Started} \wedge \neg \mathit{Ready})$

  It is possible to get to a state where Started holds but Ready does not hold

- $\mathbf{AG}(\mathit{Req} \Rightarrow \mathbf{AF}\mathit{Ack})$

  If a request Req occurs, then it will eventually be acknowledged by Ack

- $\mathbf{AG}(\mathbf{AF}\mathit{DeviceEnabled})$

  DeviceEnabled is always true somewhere along every path starting anywhere: i.e. DeviceEnabled holds infinitely often along every path

- $\mathbf{AG}(\mathbf{EF}\mathit{Restart})$

  From any state it is possible to get to a state for which Restart holds

  Can't be expressed in LTL!

More CTL examples (1)

- $\mathbf{AG}(\mathit{Req} \Rightarrow \mathbf{A}[\mathit{Req}\ \mathbf{U}\ \mathit{Ack}])$

  If a request Req occurs, then it continues to hold, until it is eventually acknowledged

- $\mathbf{AG}(\mathit{Req} \Rightarrow \mathbf{AX}(\mathbf{A}[\neg \mathit{Req}\ \mathbf{U}\ \mathit{Ack}]))$

  Whenever Req is true either it must become false on the next cycle and remains false until Ack, or Ack must become true on the next cycle

  Exercise: is the **AX** necessary?

- $\mathbf{AG}(\mathit{Req} \Rightarrow (\neg \mathit{Ack} \Rightarrow \mathbf{AX}(\mathbf{A}[\mathit{Req}\ \mathbf{U}\ \mathit{Ack}])))$

  Whenever Req is true and Ack is false then Ack will eventually become true and until it does Req will remain true

  Exercise: is the **AX** necessary?

More CTL examples (2)

- $\mathbf{AG}(\mathit{Enabled} \Rightarrow \mathbf{AG}(\mathit{Start} \Rightarrow \mathbf{A}[\neg \mathit{Waiting}\ \mathbf{U}\ \mathit{Ack}]))$

  If Enabled is ever true then if Start is true in any subsequent state then Ack will eventually become true, and until it does Waiting will be false

- $\mathbf{AG}(\neg \mathit{Req}_1 \wedge \neg \mathit{Req}_2 \Rightarrow \mathbf{A}[\neg \mathit{Req}_1 \wedge \neg \mathit{Req}_2\ \mathbf{U}\ (\mathit{Start} \wedge \neg \mathit{Req}_2)])$

  Whenever Req_1 and Req_2 are false, they remain false until Start becomes true with Req_2 still false

- $\mathbf{AG}(\mathit{Req} \Rightarrow \mathbf{AX}(\mathit{Ack} \Rightarrow \mathbf{AF}\ \neg \mathit{Req}))$

  If Req is true and Ack becomes true one cycle later, then eventually Req will become false

Some abbreviations

- $\mathbf{AX}_i\ \psi \ \equiv\ \underbrace{\mathbf{AX}(\mathbf{AX}(\cdots(\mathbf{AX}\ \psi)\cdots))}_{i \text{ instances of } \mathbf{AX}}$

  ψ is true on all paths i units of time later

- $\mathbf{ABF}_{i..j}\ \psi \ \equiv\ \mathbf{AX}_i\ \underbrace{(\psi \vee \mathbf{AX}(\psi \vee\ \cdots\ \mathbf{AX}(\psi \vee \mathbf{AX}\ \psi)\cdots))}_{j-i \text{ instances of } \mathbf{AX}}$

  ψ is true on all paths sometime between i units of time later and j units of time later

- $\mathbf{AG}(\mathit{Req} \Rightarrow \mathbf{AX}(\mathit{Ack}_1 \wedge \mathbf{ABF}_{1..6}(\mathit{Ack}_2 \wedge \mathbf{A}[\mathit{Wait}\ \mathbf{U}\ \mathit{Reply}])))$

  One cycle after Req, Ack_1 should become true, and then Ack_2 becomes true 1 to 6 cycles later and then eventually Reply becomes true, but until it does Wait holds from the time of Ack_2

- More abbreviations in 'Industry Standard' language PSL