Linear and Branching Temporal Logics

Julien Schmaltz

Institute for Computing and Information Sciences
Radboud University Nijmegen
The Netherlands
julien@cs.ru.nl

April 19, 2010

Agenda coming lectures ...

- Part I: Linear Time
- Part II: Branching Time
- Part III: Comparison
- Part IV: Binary Decision Diagrams and Symbolic Model Checking
- Part V: The SAL tool
- Part VI: Bounded Model-Checking with SAT
- Part VII: k-induction, SMT, and disjunctive invariants

Agenda for today

- Part I: Linear Time
- Part II: Branching Time
- Part III: Comparison

Part I

Linear Time Logic

Principles: next time or until ...

- Temporal logic = logic about time
- Abstract notion of (discrete) time = sequence of events
- Two principal operators
  - Next A: at the next "time" A should holds
  - A until B: A should holds until B holds
- Application to software/hardware specification
  - At the next clock cycle, the request signal must be high
  - The request signal must be high until the acknowledge is high
  - Eventually the request signal must become low again
  - The arbiter always grants at most one request
  - The elevator should never travel when the doors are open

Syntax

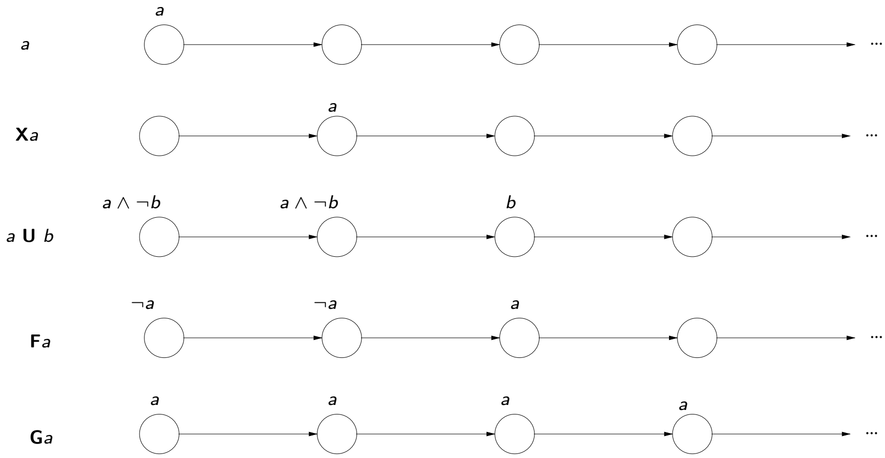
modal logic over infinite sequences [Pnueli 1977]

- Propositional logic
  - Atomic propositions: $a \in AP$
  - Boolean connectives: $\neg a$ and $\varphi \wedge \psi$
- Temporal operators
  - "Next" noted X φ or $\bigcirc\varphi$
  - "Until" noted φ U ψ or $\varphi \cup \psi$

Derived operators

- $\varphi \vee \psi \equiv \neg(\neg\varphi \wedge \neg\psi)$
- $\varphi \Rightarrow \psi \equiv \neg\varphi \vee \psi$
- $\varphi \Leftrightarrow \psi \equiv (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$
- True (or $\top$) $\equiv \varphi \vee \neg\varphi$
- False (or $\bot$) $\equiv \neg\top$
- $\mathbf{F}\varphi$ (also noted $\Diamond\varphi$) $\equiv \top\ \mathbf{U}\ \varphi$ "eventually φ"
- $\mathbf{G}\varphi$ (also noted $\Box\varphi$) $\equiv \neg\mathbf{F}\neg\varphi$ "globally φ"

Intuitive semantics



Example: traffic lights

- If the light is red, it cannot become green immediately

$$\mathbf{G}(red \Rightarrow \neg \mathbf{X} green)$$

- The traffic light eventually becomes green

$$\mathbf{F} green$$

- Once red, the light eventually becomes green

$$\mathbf{G}(red \Rightarrow \mathbf{F} green)$$

- After being red, the light goes yellow and then eventually becomes green

$$\mathbf{G}(red \Rightarrow \mathbf{X}(red \mathbf{U}(yellow \wedge \mathbf{X}(yellow \mathbf{U} green))))$$

Practical Properties of LTL

- Reachability
  - negated reachability: $\mathbf{F}\neg\psi$
  - conditional reachability: $\varphi\mathbf{U}\psi$
  - reachability from any state: not expressible
- Safety
  - simple safety: $\mathbf{G}\neg\psi$
  - conditional safety (weak until): $(\varphi\mathbf{U}\psi) \vee \mathbf{F}\varphi$
- Liveness: $\mathbf{G}(\varphi \Rightarrow \mathbf{F}\psi)$ and others
- Fairness: $\mathbf{GF}\psi$ and others

Semantics over words

A word σ is an infinite sequence of atomic propositions.
An LTL property ϕ defines the set of words for which the property is true.
$\text{Words}(\varphi) = \{\sigma \in (2^{AP})^\omega | \sigma \models \varphi\}$

$$
\begin{array}{lllll}
\sigma & \models & a & \text{iff} & a \in A_0 \text{ (or } A_0 \models a) \\
\sigma & \models & \varphi \wedge \psi & \text{iff} & \sigma \models \varphi \text{ and } \sigma \models \psi \\
\sigma & \models & \neg\varphi & \text{iff} & \sigma \not\models \varphi \\
\sigma & \models & \mathbf{X}\varphi & \text{iff} & \sigma[1..] = A_1 A_2 A_3 \ldots \models \varphi \\
\sigma & \models & \varphi \mathbf{U} \psi & \text{iff} & \exists j \geq 0 : \sigma[j..] \models \psi \text{ and } \sigma[i..] \models \varphi, 0 \leq i < j
\end{array}
$$

for $\sigma = A_0 A_1 A_2 \ldots$, $\sigma[i..] = A_i A_{i+1} A_{i+2} \ldots$ is the suffix of σ from index i

More semantics ...

$\sigma \models \mathbf{F}\psi$ iff

More semantics ...

$$\sigma \models \mathbf{F}\psi \quad \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi$$

More semantics ...

$$\begin{array}{lllll} \sigma & \models & \mathbf{F}\psi & \text{iff} & \exists j \geq 0 : \sigma[j..] \models \psi \\ \sigma & \models & \mathbf{G}\psi & \text{iff} & \end{array}$$

More semantics ...

$$
\begin{array}{llll}
\sigma & \models & \mathbf{F}\psi & \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{G}\psi & \text{iff} \quad \forall j \geq 0 : \sigma[j..] \models \psi
\end{array}
$$

More semantics ...

$$
\begin{array}{llll}
\sigma & \models & \mathbf{F}\psi & \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{G}\psi & \text{iff} \quad \forall j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{GF}\psi & \text{iff}
\end{array}
$$

More semantics ...

$$
\begin{array}{llll}
\sigma & \models & \mathbf{F}\psi & \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{G}\psi & \text{iff} \quad \forall j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{GF}\psi & \text{iff} \quad \forall j \geq 0, \exists i \geq j : \sigma[i..] \models \psi
\end{array}
$$

More semantics ...

$$
\begin{array}{llll}
\sigma & \models & \mathbf{F}\psi & \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{G}\psi & \text{iff} \quad \forall j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{GF}\psi & \text{iff} \quad \forall j \geq 0, \exists i \geq j : \sigma[i..] \models \psi \\
\sigma & \models & \mathbf{FG}\psi & \text{iff}
\end{array}
$$

More semantics ...

$$
\begin{array}{llll}
\sigma & \models & \mathbf{F}\psi & \text{iff} \quad \exists j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{G}\psi & \text{iff} \quad \forall j \geq 0 : \sigma[j..] \models \psi \\
\sigma & \models & \mathbf{GF}\psi & \text{iff} \quad \forall j \geq 0, \exists i \geq j : \sigma[i..] \models \psi \\
\sigma & \models & \mathbf{FG}\psi & \text{iff} \quad \exists j \geq 0, \forall i \geq j : \sigma[i..] \models \psi
\end{array}
$$

Duality

From the semantics, we have $\neg \mathbf{F} \neg \varphi = \mathbf{G} \varphi$.
Proof.

$$\sigma \models \neg \mathbf{F} \neg \varphi$$

Duality

From the semantics, we have $\neg\mathbf{F}\neg\varphi = \mathbf{G}\varphi$.
Proof.

$$\begin{array}{ll} \sigma \models \neg\mathbf{F}\neg\varphi & \\ \sigma \models \neg\exists j \geq 0 : \sigma[j..] \models \neg\varphi & \text{(Def. of } \mathbf{F}\text{)} \end{array}$$

Duality

From the semantics, we have $\neg\mathbf{F}\neg\varphi = \mathbf{G}\varphi$.
Proof.

$$\begin{aligned}
&\sigma \models \neg\mathbf{F}\neg\varphi \\
&\sigma \models \neg\exists j \geq 0 : \sigma[j..] \models \neg\varphi && \text{(Def. of } \mathbf{F}\text{)} \\
&\sigma \models \forall j \geq 0 : \sigma[j..] \models \varphi && \text{(Def. of } \neg\text{)}
\end{aligned}$$

Duality

From the semantics, we have $\neg\mathbf{F}\neg\varphi = \mathbf{G}\varphi$.
Proof.

$$\begin{array}{ll} \sigma \models \neg\mathbf{F}\neg\varphi & \\ \sigma \models \neg\exists j \geq 0 : \sigma[j..] \models \neg\varphi & \text{(Def. of } \mathbf{F}\text{)} \\ \sigma \models \forall j \geq 0 : \sigma[j..] \models \varphi & \text{(Def. of } \neg\text{)} \\ \sigma \models \mathbf{G}\varphi & \text{(Def. of } \mathbf{G}\text{)} \end{array}$$

Semantics over paths, states, and transition systems

Let $TS = (S, \Sigma, T, I, AP, L)$ be an LTS and φ be an LTL formula over AP.

- For infinite path π of TS, the traces are the words that are valid in the states of the path.

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

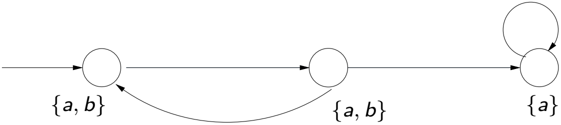
- A state $s \in S$ satisfies φ iff all paths from s satisfy φ

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

- A transition system satisfies φ iff φ holds from the initial state(s)

$$TS \models \varphi \text{ iff } Traces(TS) \subseteq Words(\varphi) \text{ iff } \forall s_0 \in I : s_0 \models \varphi$$

Example



$TS \models \mathbf{G}a$

$TS \models \mathbf{X}(a \wedge b)$

$TS \models \mathbf{G}(\neg b \Rightarrow \mathbf{G}(a \wedge \neg b))$

$TS \not\models b\mathbf{U}(a \wedge \neg b)$

Semantics of negation

For paths, it holds $\pi \models \varphi$ iff $\pi \not\models \neg\varphi$ since:

$$Words(\neg\varphi) = (2^{AP})^{\omega} \setminus Words(\varphi)$$

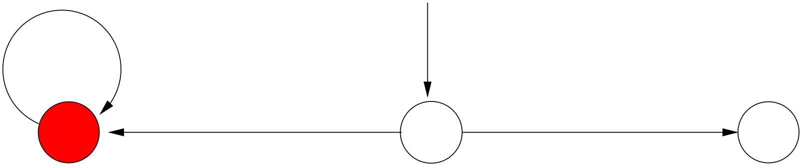
But: $TS \not\models \varphi$ and $TS \models \neg\varphi$ are not equivalent in general

It holds: $TS \models \neg\varphi$ implies $TS \neg \models \varphi$.

TS neither satisfies φ or $\neg\varphi$ if there are paths π_1 and π_2 such that $\pi_1 \models \varphi$ and $\pi_2 \models \neg\varphi$.

Example

A transition system for which $TS \not\models \mathbf{F}a$ and $TS \not\models \neg\mathbf{F}a$.



More dualities and idempotent laws

- Duality

$$\begin{aligned} \neg \mathbf{G}\varphi &\equiv \mathbf{F}\neg\varphi \\ \neg \mathbf{F}\varphi &\equiv \mathbf{G}\neg\varphi \\ \neg \mathbf{X}\varphi &\equiv \mathbf{X}\neg\varphi \end{aligned}$$

- Idempotency

$$\begin{aligned} \mathbf{GG}\varphi &\equiv \mathbf{G}\varphi \\ \mathbf{FF}\varphi &\equiv \mathbf{F}\varphi \\ \varphi\mathbf{U}(\varphi\mathbf{U}\psi) &\equiv \varphi\mathbf{U}\psi \\ (\varphi\mathbf{U}\psi)\mathbf{U}\psi &\equiv \varphi\mathbf{U}\psi \end{aligned}$$

Absorption and distributive laws

- Absorption

$$\begin{aligned} \mathbf{FGF}\varphi &\equiv \mathbf{GF}\varphi \\ \mathbf{GFG}\varphi &\equiv \mathbf{FG}\varphi \end{aligned}$$

- Distribution

$$\begin{aligned} \mathbf{X}(\varphi \mathbf{U} \psi) &\equiv (\mathbf{X}\varphi)\mathbf{U}(\mathbf{X}\psi) \\ \mathbf{F}(\varphi \vee \psi) &\equiv \mathbf{F}\varphi \vee \mathbf{F}\psi \\ \mathbf{G}(\varphi \wedge \psi) &\equiv \mathbf{G}\varphi \wedge \mathbf{G}\psi \end{aligned}$$

- But we have:

$$\begin{aligned} \mathbf{F}(\varphi \wedge \psi) &\not\equiv \mathbf{F}\varphi \wedge \mathbf{F}\psi \\ \mathbf{G}(\varphi \vee \psi) &\not\equiv \mathbf{G}\varphi \vee \mathbf{G}\psi \end{aligned}$$

Absorption Laws(1)

$\mathbf{FGF}\varphi \equiv \mathbf{GF}\varphi$

More formally: $\mathbf{GF}\varphi$ means $\forall i \geq 0, \exists j \geq i : \sigma[j..] \models \varphi$

$\mathbf{FGF}\varphi$ means $\exists k \geq 0, \forall i \geq k, \exists j \geq i : \sigma[j..] \models \varphi$

Absorption Laws(2)

$\mathbf{GFG}\varphi \equiv \mathbf{FG}\varphi$

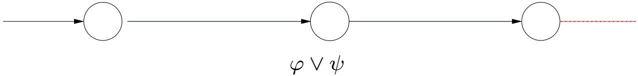
More formally: $\mathbf{FG}\varphi$ means $\exists i \geq 0, \forall j \geq i : \sigma[j..] \models \varphi$

$\mathbf{GFG}\varphi$ means $\forall k \geq 0, \exists i \geq k, \forall j \geq i : \sigma[j..] \models \varphi$

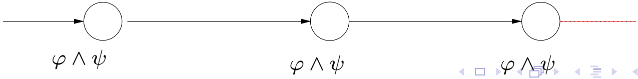
Distributive Laws (1)

$$\mathbf{X}(\varphi \mathbf{U} \psi) \equiv (\mathbf{X}\varphi)\mathbf{U}(\mathbf{X}\psi)$$

$$\mathbf{F}(\varphi \vee \psi) \equiv \mathbf{F}\varphi \vee \mathbf{F}\psi$$

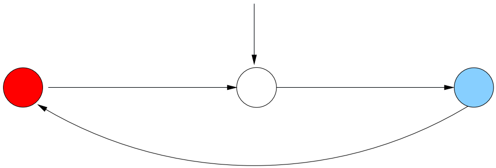


$$\mathbf{G}(\varphi \wedge \psi) \equiv \mathbf{G}\varphi \wedge \mathbf{G}\psi$$



Distributive Laws (2)

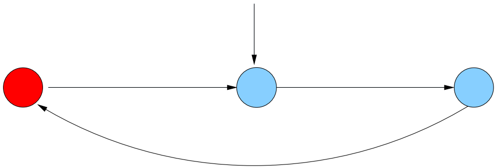
$\mathbf{F}(a \wedge b) \not\equiv \mathbf{F}a \wedge \mathbf{F}b$



$TS \not\models \mathbf{F}(a \wedge b)$ and $TS \models \mathbf{F}a \wedge \mathbf{F}b$

Distributive Laws (3)

$\mathbf{G}(a \vee b) \not\equiv \mathbf{G}a \vee \mathbf{G}b$



$TS \models \mathbf{G}(a \vee b)$ and $TS \not\models \mathbf{G}a \vee \mathbf{G}b$

Part II

Branching Time Logic

4 Branching Time Logic

5 Syntax and Semantics
- Syntax of CTL
- Semantics

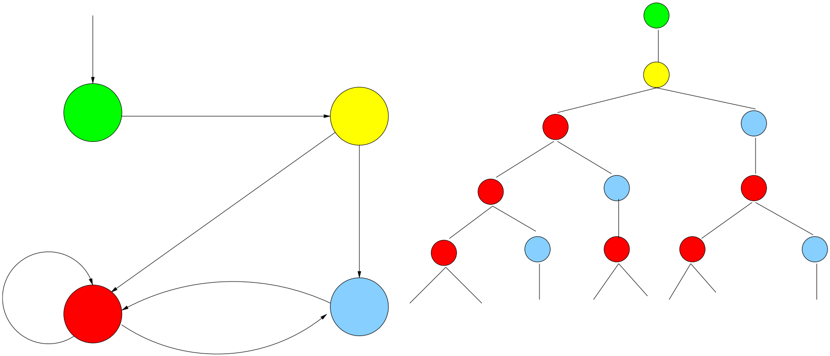
Linear and Branching Timed

- Linear time
  - Properties about all paths in state s
  - $s \models \mathbf{G}\varphi$ iff for all paths starting in s, φ holds for all time instants ("always" or "globally")
- Branching time
  - Properties about all or some paths starting in state s
  - $s \models \mathbf{AG}\varphi$ iff for all paths starting in s, φ holds globally on the path
  - $s \models \mathbf{EG}\varphi$ iff for some path starting in s, φ holds globally on the path

Linear vs. Branching Timed

- Semantics based on a branching notion of time
  - infinite tree of states obtained by unfolding a transition system
  - one "time instant" may have several successor states for the next "time instants"
  - Linear time: "one only lives one future"
  - Branching time: "one has many possible futures"
- Expressiveness: incomparable
  - There are linear properties that cannot be stated as branching properties
  - There are branching properties that cannot be stated as linear properties

Transition Systems and Trees



Computational Tree Logic (CTL)

modal logic over infinite trees [Clarke & Emerson 1981]

- State formulae containing path quantifiers
  - atomic proposition: $a \in AP$
  - Boolean connectives: $\neg\varphi$ and $\varphi \wedge \psi$
  - there exists a path satisfying φ: $\mathbf{E}\varphi$ or $\exists\varphi$
  - all paths satisfy φ: $\mathbf{A}\varphi$ or $\forall\varphi$
- Paths formulae containing temporal operators
  - Next φ: $\mathbf{X}\varphi$ or $\bigcirc\varphi$
  - φ until ψ: $\varphi\mathbf{U}\psi$
- In a CTL formula path and state formulae alternate

Derived Operators

- Potentially φ: $\mathbf{EF}\varphi = \mathbf{E}(\top \mathbf{U} \varphi)$
- Inevitably φ: $\mathbf{AF}\varphi = \mathbf{A}(\top \mathbf{U} \varphi)$
- Potentially always φ: $\mathbf{EG}\varphi = \neg \mathbf{AF} \neg \varphi$
- Invariantly φ: $\mathbf{AG}\varphi = \neg \mathbf{EF} \neg \varphi$
- Weak until
  - $\mathbf{E}(\varphi \mathbf{W} \psi) = \neg \mathbf{A}((\varphi \wedge \neg \psi) \mathbf{U} (\neg \varphi \wedge \neg \psi))$
  - $\mathbf{A}(\varphi \mathbf{W} \psi) = \neg \mathbf{E}((\varphi \wedge \neg \psi) \mathbf{U} (\neg \varphi \wedge \neg \psi))$

Operators

- Basic operators: **EX**, **EG**, **EU**
- Derived operators:
  - $\mathbf{AX}\varphi = \neg\mathbf{EX}(\neg\varphi)$
  - $\mathbf{EF}\varphi = \mathbf{E}(\top\mathbf{U}\varphi)$
  - $\mathbf{AG}\varphi = \neg\mathbf{EF}(\neg\varphi)$
  - $\mathbf{AF}\varphi = \neg\mathbf{EG}(\neg\varphi)$
  - $\mathbf{A}(\varphi\mathbf{U}\psi) = \neg\mathbf{E}(\neg\psi\mathbf{U}(\neg\varphi \wedge \neg\psi) \wedge \neg\mathbf{EG}\neg\psi$

Some typical CTL formulae

- It is possible to get to a state where *Start* holds but *Ready* does not

$$\mathbf{EF}(Start \wedge \neg Ready)$$

- If a request occurs, then it will be eventually acknowledged

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

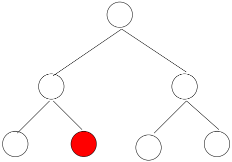
- Proposition *Ready* holds infinitely often on every path

$$\mathbf{AG}(\mathbf{AF}\,Ready)$$

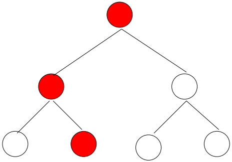
- From any state it is possible to *Restart*

$$\mathbf{AG}(\mathbf{EF}\,Restart)$$

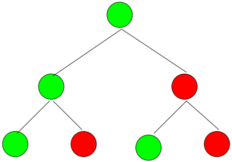
Informal Semantics



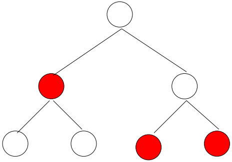
EF *red*



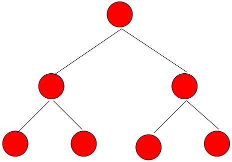
EG *red*



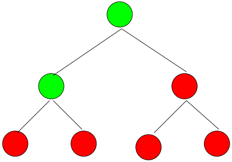
E(*green* **U** *red*)



AF *red*



AG *red*



A(*green* **U** *red*)

Give semantics of typical formulae ...

- $\mathbf{EF}p$
- $\mathbf{AG}(\mathbf{AF}p)$
- $\mathbf{AG}(\mathbf{EF}p)$
- $\mathbf{AG}(p \Rightarrow \mathbf{AF}q)$

Semantics of state-formulae

$s \models \varphi$ iff formula φ holds in state s

$$
\begin{array}{llll}
s & \models & a & \text{iff} \quad a \in L(s) \\
s & \models & \neg\varphi & \text{iff} \quad \neg(s \models \varphi) \\
s & \models & \varphi \wedge \psi & \text{iff} \quad (s \models \varphi) \text{ and } (s \models \psi) \\
s & \models & \mathbf{E}\varphi & \text{iff} \quad \pi \models \varphi \text{ for some path } \pi \text{ from } s \\
s & \models & \mathbf{A}\varphi & \text{iff} \quad \pi \models \varphi \text{ for all paths } \pi \text{ from } s
\end{array}
$$

Semantics of path-formulae

$\pi \models \varphi$ iff path π satisfies φ

$$
\begin{array}{llll}
\pi & \models & \mathbf{X}\varphi & \text{iff} \quad \pi[1] \models \varphi \\
\pi & \models & \varphi\mathbf{U}\psi & \text{iff} \quad (\exists j \geq 0 : \pi[j] \models \psi \wedge (\forall 0 \leq k < j : \pi[k] \models \varphi))
\end{array}
$$

where $\pi[i]$ denotes the state with index i (s_i) in π

Transition System Semantics

- TS satisfies CTL-formula φ iff φ holds in all initial states

$$TS \models \varphi \text{ iff } \forall s_0 \in I : s_0 \models \varphi$$

- Point of attention: $TS \not\models \varphi$ and $TS \not\models \neg\varphi$ is possible !
  - because of several initial states. We can have $s_0 \models \mathbf{EG}\varphi$ and $s'_0 \not\models \mathbf{EG}\varphi$

Part III

LTL vs. CTL

6 LTL vs. CTL

7 CTL*

- Syntax
- Semantics
- CTL, LTL, and CTL*

Equivalence of LTL and CTL formulae

- CTL-formula ϕ and LTL-formula φ (both over AP) are equivalent, denoted $\phi \equiv \varphi$, if for any transition system TS (over AP):

$$TS \models \phi \qquad \text{if and only if} \qquad TS \models \varphi$$

- Let ϕ be a CTL-formula, and φ the LTL-formula obtained by eliminating all path quantifiers in ϕ. Then;

  $\phi \equiv \varphi$ or there does not exist any LTL-formula that is equivalent to ϕ

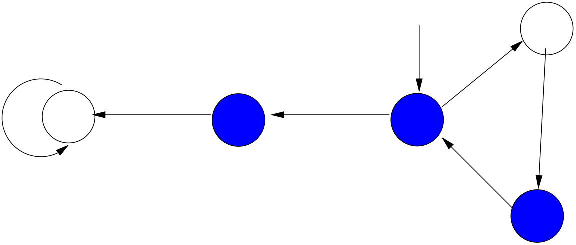
LTL and CTL are incomparable

- Some LTL-formulae cannot be expressed in CTL
  - $\mathbf{FG}a$
  - $\mathbf{F}(a \wedge \mathbf{X}a)$
- Some CTL-formulae cannot be expressed in LTL
  - $\mathbf{AFAG}a$
  - $\mathbf{AF}(a \wedge \mathbf{A}Xa)$
  - $\mathbf{AGEF}a$ ("restart" property)

"Cannot be expressed" means there does not exist an equivalent formula

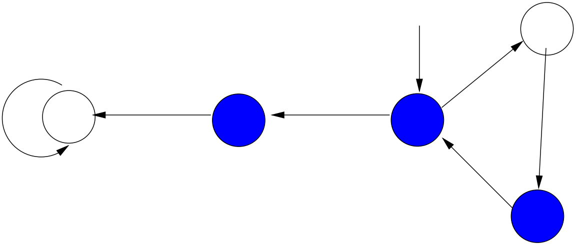
Comparing LTL and CTL (1)

$$\mathbf{F}(a \wedge \mathbf{X}a) \not\equiv \mathbf{AF}(a \wedge \mathbf{AX}a)$$



Comparing LTL and CTL (1)

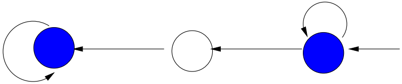
$$\mathbf{F}(a \wedge \mathbf{X}a) \not\equiv \mathbf{AF}(a \wedge \mathbf{AX}a)$$



$s_0 \models \mathbf{F}(a \wedge \mathbf{X}a)$
$s_0 \not\models \mathbf{AF}(a \wedge \mathbf{AX}a)$
Counter-examples: $s_0 s_1 (s_2)^\omega$

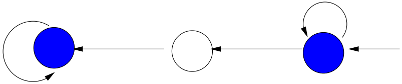
Comparing LTL and CTL (2)

$$\mathbf{AFAG}a \neq \mathbf{FG}a$$



Comparing LTL and CTL (2)

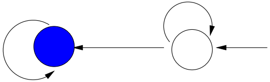
$$\mathbf{AFAG}a \neq \mathbf{FG}a$$



$s_0 \models \mathbf{FG}a$
$s_0 \not\models \mathbf{AFAG}a$
Counter-examples: s_0^ω

Comparing LTL and CTL (3)

$$\mathbf{AGEF}a \not\equiv \mathbf{GF}a$$



$s_0 \not\models \mathbf{GF}a$ but $s_0 \models \mathbf{AGEF}a$

Syntax of CTL*

- CTL* state-formulae are formed according to:

$$\phi ::= \top | a | \phi_1 \vee \phi_2 | \neg\phi | \mathbf{E}\psi$$

  where $a \in AP$, ϕ, ϕ_1, ϕ_2 are state-formulae, and ψ is a path-formula
- CTL* path-formulae are formed according to:

$$\psi ::= \phi | \psi_1 \ \vee \psi_2 | \neg\psi | \mathbf{X}\psi | \psi_1 \ \mathbf{U}\psi_2$$

  where ϕ is a state-formula, and $\varphi, \varphi_1, \varphi_2$ are path-formulae
- Path-quantifiers and temporal operators do not have to alternate anymore
- In CTL* we have $\mathbf{A}\psi = \neg\mathbf{E}\neg\psi$ which cannot be written in CTL!

Semantics of CTL*

$$
\begin{array}{llll}
s & \models & a & \text{iff} \quad a \in L(s) \\
s & \models & \neg\phi & \text{iff} \quad \neg(s \models \phi) \\
s & \models & \phi_1 \wedge \phi_2 & \text{iff} \quad (s \models \phi_1) \text{ and } (s \models \phi_2) \\
s & \models & \mathbf{E}\psi & \text{iff} \quad \pi \models \psi \text{ for some path } \pi \text{ from } s \\
s & \models & \mathbf{A}\psi & \text{iff} \quad \pi \models \psi \text{ for all paths } \pi \text{ from } s
\end{array}
$$

$$
\begin{array}{lll}
\pi & \models \phi & \text{iff} \quad \pi[0] \models \phi \\
\pi & \models \psi_1 \wedge \psi_2 & \text{iff} \quad \pi \models \psi_1 \text{ and } \pi \models \psi_2 \\
\pi & \models \neg\psi & \text{iff} \quad \pi \not\models \psi \\
\pi & \models \mathbf{X}\psi & \text{iff} \quad \pi[1..] \models \psi \\
\pi & \models \psi_1 \mathbf{U} \psi_2 & \text{iff} \quad (\exists j \geq 0 : \pi[j..] \models \psi_2 \wedge (\forall 0 \leq k < j : \pi[k..] \models \psi_1))
\end{array}
$$

for path $\pi = s_0 s_1 s_2 \cdots$, $\pi[i..]$ denotes suffix of σ from index i on

Embedding LTL in CTL*

For LTL formula ψ, transition system TS, and state s:

$$s \models_{LTL} \psi \text{ if and only } s \models_{CTL*} \mathbf{A}\psi$$

We also have:

$$TS \models_{LTL} \psi \text{ if and only if } TS \models_{CTL*} \mathbf{A}\psi$$

CTL* is more expressive tan LTL and CTL

We have seen that $\mathbf{FG}a$ cannot be expressed in CTL.
We have seen that $\mathbf{AGEF}b$ cannot be expressed in LTL.
The CTL* formula $\phi = (\mathbf{AFG}a) \vee (\mathbf{AGEF}b)$ is in CTL* !

LTL, CTL, and CTL*