6. Linear Temporal Logic

Computer-Aided Verification

Dave Parker

University of Birmingham

2017/18

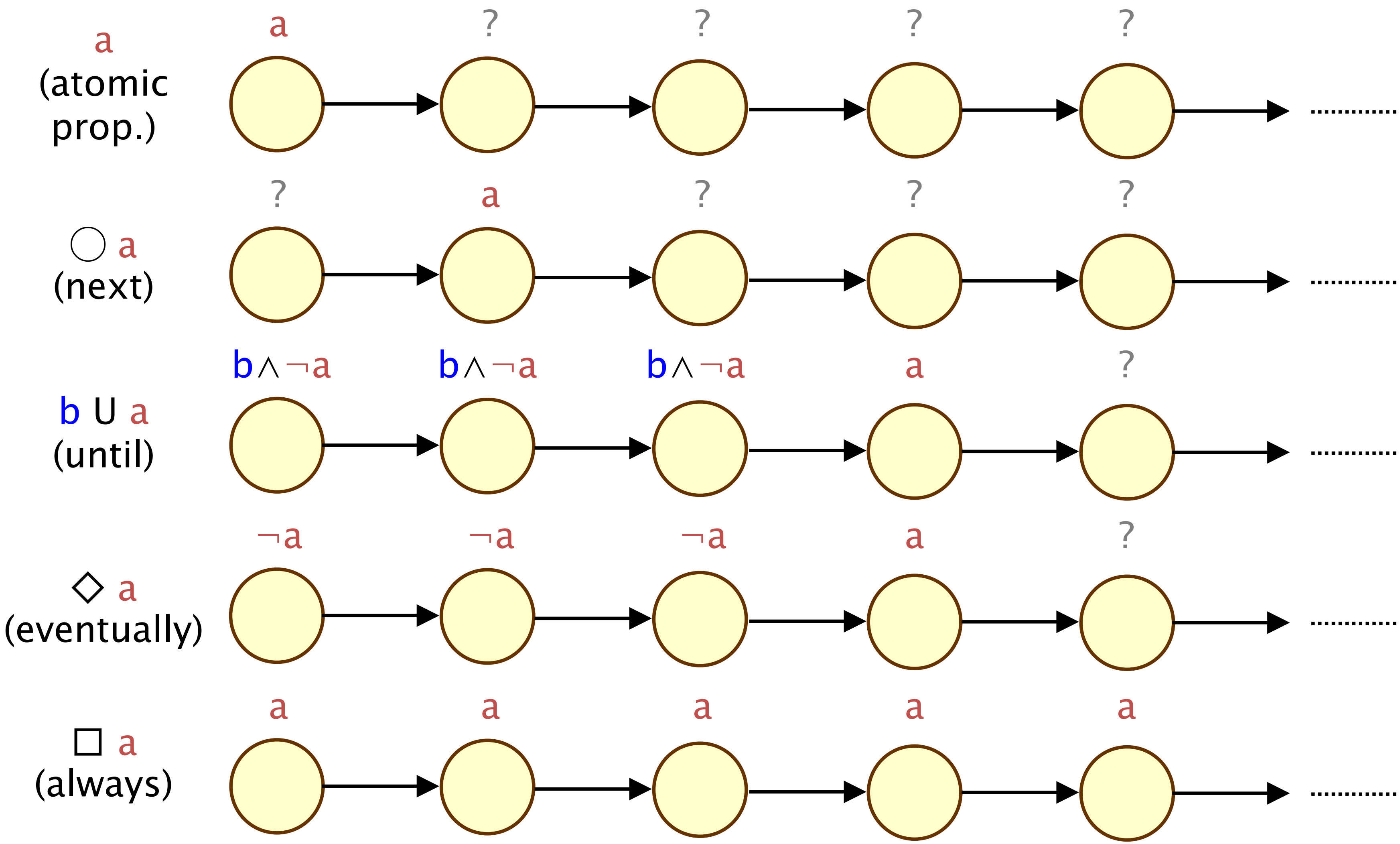
Recap: Temporal logic

- Propositional logic
  - syntax, semantics, equivalences (derived operators)
  - a, b (atomic propositions), $\wedge$ (conjunction), $\vee$ (disjunction), $\neg$ (negation), $\rightarrow$ (implication), etc.

- Temporal logic
  - precise, unambiguous specification of correctness properties
  - extends propositional logic with temporal operators
  - $\bigcirc$ (next), U (until), $\Diamond$ (eventually), $\Box$ (always)

- Linear temporal logic (LTL)

LTL – Intuitive semantics

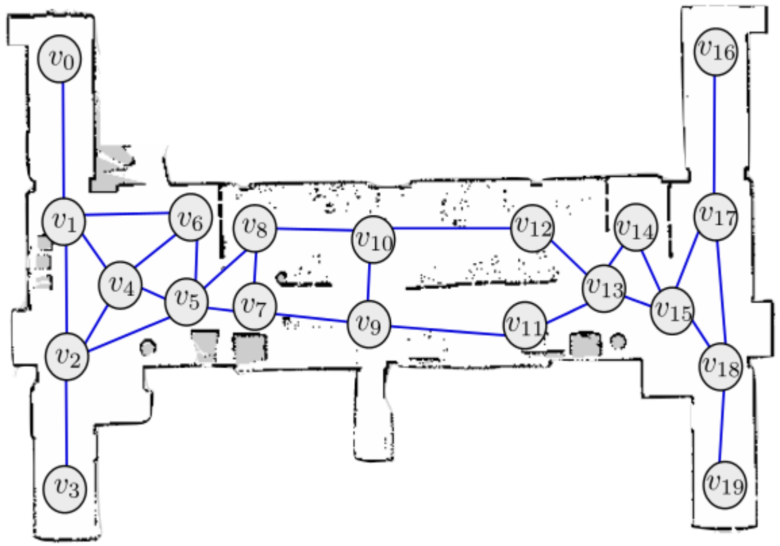


LTL – More properties

- LTL syntax:
  - $\psi ::= \text{true} \mid a \mid \psi \wedge \psi \mid \neg\psi \mid \bigcirc\, \psi \mid \psi \,\mathsf{U}\, \psi \mid \Diamond\, \psi \mid \Box\, \psi$
  - many more properties formed by combining temporal operators
  - simple examples: $(\Diamond\, a) \wedge (\Diamond\, b)$, $\bigcirc \bigcirc a$, $a \wedge \bigcirc \bigcirc a$
- $\Box(a \rightarrow \Diamond b)$
  - "b always follows a"
- $\Box(a \rightarrow \bigcirc b)$
  - "b always immediately follows a"
- $\Box \Diamond a$
  - "a is true infinitely often"
- $\Diamond \Box a$
  - "a becomes true and remains true forever"

Other uses of LTL

- Example: robot task specifications
  - $\neg \mathrm{zone}_3 \ \mathrm{U} \ (\mathrm{zone}_1 \wedge (\Diamond \ \mathrm{zone}_4)$
    - visit zone 1 (without passing through zone 3), and then go to zone 4
  - $(\Box \neg \mathrm{zone}_3) \wedge (\Box \ \Diamond \ \mathrm{zone}_5)$
    - avoid zone 3 and patrol zone_5 infinitely often



LTL semantics

- Recall: we define properties in terms of:
  - infinite words $\sigma = A_0A_1A_2A_3\ldots$ over 2^{AP}
- Some notation:
  - $\sigma[j]$ is the (j+1)th symbol, i.e. A_j
  - $\sigma[j\ldots]$ is the suffix starting in $\sigma[j]$, i.e. $A_jA_{j+1}A_{j+2}\ldots$
- LTL semantics ($\sigma \vDash \psi$, for infinite word σ and LTL formula ψ)
  - $\sigma \vDash \text{true}$ always
  - $\sigma \vDash a \quad \Leftrightarrow \quad a \in \sigma[0]$
  - $\sigma \vDash \psi_1 \wedge \psi_2 \quad \Leftrightarrow \quad \sigma \vDash \psi_1$ and $\sigma \vDash \psi_2$
  - $\sigma \vDash \neg\psi \quad \Leftrightarrow \quad \sigma \nvDash \psi$
  - $\sigma \vDash \bigcirc\, \psi \quad \Leftrightarrow \quad \sigma[1\ldots] \vDash \psi$
  - $\sigma \vDash \psi_1 \,\mathsf{U}\, \psi_2 \quad \Leftrightarrow \quad \exists k \geq 0$ s.t. $\sigma[k\ldots] \vDash \psi_2$ and $\forall i<k\ \sigma[i\ldots] \vDash \psi_1$

LTL semantics

- When does an LTS M satisfy an LTL formula ψ?
  - intuitively, if all paths of M satisfy ψ

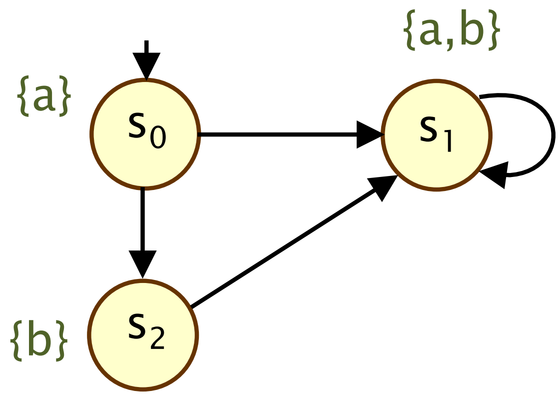
- More precisely:
  - if all traces of all paths of M satisfy ψ:
  - $M \models \psi \;\Leftrightarrow\; trace(\pi) \models \psi$ for every $\pi \in Paths(M)$

- Alternatively (using a linear-time property):
  - $Words(\psi) = \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \psi \}$
  - $M \models \psi \;\Leftrightarrow\; Traces(M) \subseteq Words(\psi)$

Examples

- $M \models \Box\,(a \vee b)$?
- $M \models b$?
- $M \models \bigcirc\, b$?
- $M \models \Box \bigcirc\, b$?
- $M \models \Box \Diamond \neg a$?
- $M \models \Box((a \wedge \neg b) \rightarrow \Diamond \neg b)$?

M:



What can we express in LTL?

- Invariants?
  - yes: $\square\Phi$, for some propositional formula Φ
    - in fact, *all* invariants can be represented

- Safety properties?
  - yes: e.g. $\square(\text{receive} \rightarrow \bigcirc \text{ack})$
    - "ack always immediately follows receive"

- Liveness properties?
  - yes: e.g. $\Diamond \text{terminates}$
    - "the program eventually terminates"
  - yes: e.g. $\square\Diamond \text{ready}$
    - "the server always gets back into a ready state"

Equivalence

- LTL formulae ψ_1 and ψ_2 are equivalent, written $\psi_1 \equiv \psi_2$ if:
  - they are satisfied by exactly the same traces
  - $\sigma \vDash \psi_1 \Leftrightarrow \sigma \vDash \psi_2$ (for any trace σ)
  - i.e. $\text{Words}(\psi_1) = \text{Words}(\psi_2)$

With respect to some set AP of propositions

- Or, equivalently:
  - if they are satisfied by exactly the same models
  - $M \vDash \psi_1 \Leftrightarrow M \vDash \psi_2$ (for any LTS M)

- This gives us a notion of expressiveness of LTL
  - "expressiveness" = "expressivity" = '"expressive power"
  - i.e. which models can LTL distinguish between?

LTL equivalences

- Equivalences
  - shorthand for common formulae, e.g.: $\Diamond\ \psi \equiv \text{true}\ U\ \psi$
  - simplifications, e.g.: $\neg\neg p \equiv p$
  - syntax vs. semantics

Does not add expressive power to LTL

- Equivalences for: propositional logic + temporal operators

- Temporal operator equivalences:
  - $\Box\psi \equiv \neg\Diamond\neg\psi$ (duality)
  - $\Box\Box\psi \equiv \Box\psi$ (idempotency)
  - $\Diamond\psi \equiv \psi \vee \bigcirc\Diamond\psi$ (expansion law)
  - $\Box(\psi_1 \wedge \psi_2) \equiv \Box\psi_1 \wedge \Box\psi_2$ (distributive law)

Example 1

- Prove (or disprove):

$$\Diamond\psi \equiv \psi \vee \bigcirc\Diamond\psi \quad ? \qquad\qquad \text{Yes}$$

- Can prove directly, using the relevant semantics for LTL:
  - $\sigma \vDash \psi_1 \vee \psi_2 \iff \sigma \vDash \psi_1$ or $\sigma \vDash \psi_2$
  - $\sigma \vDash \bigcirc\,\psi \iff \sigma\,[1\ldots] \vDash \psi$
  - $\sigma \vDash \psi_1 \,\mathsf{U}\, \psi_2 \iff \exists k \geq 0$ s.t. $\sigma[k\ldots] \vDash \psi_2$ and $\forall i<k\ \sigma[i\ldots] \vDash \psi_1$

Example 2

- Prove (or disprove):

$$\neg(\Box a \rightarrow \Diamond b) \;\equiv\; \Box a \wedge \Box\neg b \quad ? \quad \text{Yes}$$

- Can prove by reusing simpler known equivalences
  - $\psi_1 \rightarrow \psi_2 \;\equiv\; \neg\psi_1 \vee \psi_2$
  - $\Box\psi \;\equiv\; \neg\Diamond\neg\psi$
  - etc.

Example 3

- Prove (or disprove):

$$\Box\Diamond a \wedge \Box\Diamond b \equiv \Box\Diamond(a \wedge b) \quad ? \quad \text{No}$$

- Just need to provide a single trace as a counterexample
  - e.g. {a} {b} {a} {b} …
  - (which is satisfied by the left formula only)

LTL & Negation

- Are these statements equivalent? (for trace σ and LTL formula ψ)
  - $\sigma \models \neg\psi$
  - $\sigma \not\models \psi$
- Yes
  - in fact, this is just the semantics of LTL

- Are these statements equivalent? (for LTS M and LTL formula ψ)
  - $M \models \neg\psi$
  - $M \not\models \psi$
- No:
  - $M \models \neg\psi$ means no trace satisfies ψ
  - $M \not\models \psi$ means it is not true that all traces satisfy ψ
    - i.e. there exists some trace that does not satisfy ψ

Existential properties

- Can we verify this, using LTL?
  - "there exists an execution that reaches program location l_2"
- Yes: $M \nvDash \Box \neg l_2$

- Can we verify this, using LTL?
  - "there exists an execution that visits l_2 infinitely often, and never passes through program location l_4"
- Yes: $M \nvDash \neg((\Box \Diamond l_2) \wedge (\Box \neg l_4))$

- Can we verify this, using LTL?
  - "for every execution, it is always possible to return to the initial state of the program"
- No…