8. CTL and LTL

Computer-Aided Verification

Dave Parker

University of Birmingham

2017/18

Reminders

- Assignment 1
  - marks & individual feedback out today
  - also covered in this week's tutorials…
- Tutorials this week
  - Today (Thur) 4pm (surnames A–L, by default):
    - UG06, Murray Learning Centre
  - Tomorrow (Fri) 10am (surnames M–Z, by default):
    - Lecture Theatre 1, Sports and Exercise Sciences
- Assignment 2 (temporal logic)
  - out today, due in a week (12 noon, Thur 8 Feb)

Recap + Overview

- Temporal logic: Negation, existence of paths

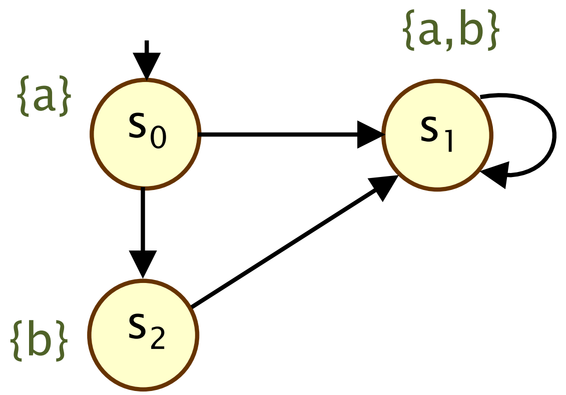
- Computation Tree Logic (CTL)
  - usual temporal operators ($\bigcirc$, U, $\Diamond$, $\Box$)
  - plus path quantifiers: $\forall$ (for all paths), $\exists$ (there exists a path)
  - evaluated over states, not paths

- Today
  - CTL equivalences and normal form
  - CTL vs. LTL (and CTL*)
  - fairness

Examples

- $s_0 \vDash \forall \bigcirc\, b$?
- $s_0 \vDash \exists \bigcirc\, \neg b$?
- $s_0 \vDash \exists(a \;\mathsf{U}\; a \wedge b)$?
- $s_0 \vDash \exists \bigcirc\, \forall \Box\, (a \wedge b)$?

M:



CTL semantics

- Semantics of state formulae:
  - $s \models \phi$ denotes "s satisfies ϕ" or "ϕ is true in s"
- For a state s of an LTS $(S, Act, \rightarrow, I, AP, L)$:
  - $s \models true$ always
  - $s \models a \Leftrightarrow a \in L(s)$
  - $s \models \phi_1 \wedge \phi_2 \Leftrightarrow s \models \phi_1$ and $s \models \phi_2$
  - $s \models \neg\phi \Leftrightarrow s \not\models \phi$
  - $s \models \forall\ \psi \Leftrightarrow \pi \models \psi$ for all $\pi \in Path(s)$
  - $s \models \exists\ \psi \Leftrightarrow \pi \models \psi$ for some $\pi \in Path(s)$
- and for a path π:
  - $\pi \models \bigcirc\ \phi \Leftrightarrow \pi[1] \models \phi$
  - $\pi \models \phi_1\ U\ \phi_2 \Leftrightarrow \exists k \geq 0$ s.t. $\pi[k] \models \phi_2$ and $\forall i<k\ \pi[i] \models \phi_1$

(i+1)th state of path π

CTL equivalences

- Again various operators can be derived
  - propositional logic: $\vee$, $\rightarrow$, $\leftrightarrow$, $\oplus$

- Path quantifier duality:
  - $\forall\psi \equiv \neg\exists\neg\psi$
  - $\exists\psi \equiv \neg\forall\neg\psi$

- Temporal operators:
  - $\Diamond \phi \equiv$ true U ϕ
  - $\Box \phi \equiv$?
- For example:
  - $\forall\Box \phi \equiv \neg\exists\Diamond(\neg\phi)$

Existential normal form (ENF)

- Often useful to consider normal forms for logics
  - e.g. checking equality, simplifying algorithms/proofs
- Recall: full syntax for CTL formula ϕ:
  - ϕ ::= true | a | $\phi \wedge \phi$ | $\neg\phi$ | $\forall\, \psi$ | $\exists\, \psi$
  - ψ ::= $\bigcirc\, \phi$ | ϕ U ϕ | $\Diamond\, \phi$ | $\Box\, \phi$
- Existential normal form (ENF) for CTL
  - no universal path quantifier ($\forall$) allowed, and no $\exists\Diamond$ formulae
  - ϕ ::= true | a | $\phi \wedge \phi$ | $\neg\phi$ | $\exists\bigcirc\phi$ | $\exists(\phi$ U $\phi)$ | $\exists\Box\phi$
- $\forall$ can be removed using path quantifier duality:
  - $\forall\, \psi \equiv \neg\exists\, \neg\psi$

Conversion to ENF

- Allowed:
  - $\exists\bigcirc$, $\exists U$, $\exists\Box$
- Not allowed:
  - $\exists\Diamond$, $\forall\bigcirc$, $\forall U$, $\forall\Diamond$, $\forall\Box$

- Can always convert to ENF:
  - $\exists\Diamond\phi \equiv \exists(\text{true } U\ \phi)$
  - $\forall\bigcirc\phi \equiv \neg\exists\bigcirc\neg\phi$
  - $\forall\Diamond\phi \equiv \neg\exists\Box\neg\phi$
  - $\forall\Box\phi \equiv \neg\exists\Diamond\neg\phi \equiv \neg\exists(\text{true } U\ \neg\phi)$
  - $\forall(\phi_1\ U\ \phi_2) \equiv \neg\exists((\neg\phi_2\ U\ (\neg\phi_1 \wedge \neg\phi_2))) \wedge \neg\exists(\Box\neg\phi_2)$

ENF Conversion – Example

- CTL formula ϕ
  - $\phi = \forall\Diamond(\forall\bigcirc(b\vee\neg c) \vee \exists\Diamond(a \wedge b))$
- Convert to equivalent CTL formula ϕ' in ENF
  - $\phi' = \neg\exists\Box(\exists\bigcirc(\neg b\wedge c) \wedge \neg\exists(\text{true } U \ (a \wedge b))$
- (Start at the outside and work in)

CTL vs LTL

- How do we compare the expressiveness of CTL and LTL?
  - evaluated over states and paths, respectively

- Satisfaction of a CTL formula ϕ by an LTS M:
  - $M \models \phi$ if $s_0 \models \phi$ for all initial states s_0 of M

- CTL formulae ϕ_1 and ϕ_2 are equivalent ($\phi_1 \equiv \phi_2$) if
  - $M \models \phi_1 \Leftrightarrow M \models \phi_2$ (for any LTS M)

- CTL formula ϕ and LTL formula ψ are equivalent ($\phi \equiv \psi$) if
  - $M \models \phi \Leftrightarrow M \models \psi$ (for any LTS M)

Expressiveness of CTL and LTL

- Is CTL more expressive than LTL?

- What can we express in LTL that we cannot in CTL?

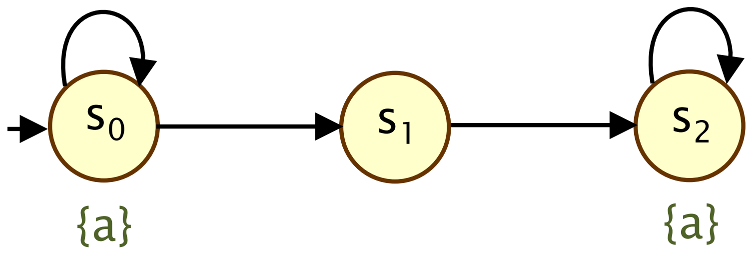
  - what about $\Box\Diamond a$?
  - no: $\forall\Box\forall\Diamond a \;\equiv\; \Box\Diamond a$

  - similarly: $\forall\Box(a\rightarrow\forall\bigcirc b) \;\equiv\; \Box(a\rightarrow\bigcirc b)$

  - what about $\Diamond\Box a$?
  - $\forall\Diamond\forall\Box a \;\equiv\; \Diamond\Box a$?

Expressiveness of CTL and LTL

- Counterexample showing: $\forall\Diamond\forall\Box a \not\equiv \Diamond\Box a$:
  - (note that a counterexample is now an LTS, not a trace)



- In fact, $\Diamond\Box a$ has no equivalent formula in CTL

- Similarly, $\forall\Box\exists\Diamond a$ has no equivalent formula in LTL

- The expressiveness of CTL and LTL are incomparable

CTL vs. LTL

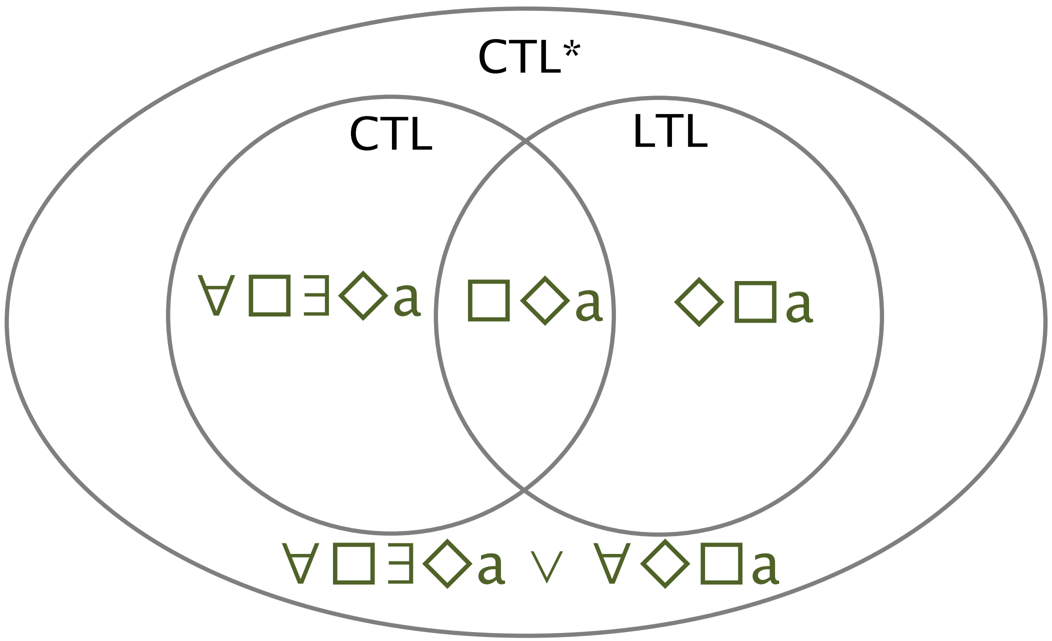
- Key differences between CTL and LTL:
  - branching-time vs. linear-time
  - state-based vs. path-based
  - expressiveness: incomparable
  - model checking algorithms differ
    - CTL simpler and lower complexity than LTL
    - (linear in size of ϕ vs. exponential in size of ψ)
  - fairness dealt with more easily in LTL

- Both CTL and LTL are a subset of the logic CTL*
  - path quantifiers ($\forall$,$\exists$) arbitrarily nested with temporal operators

CTL*

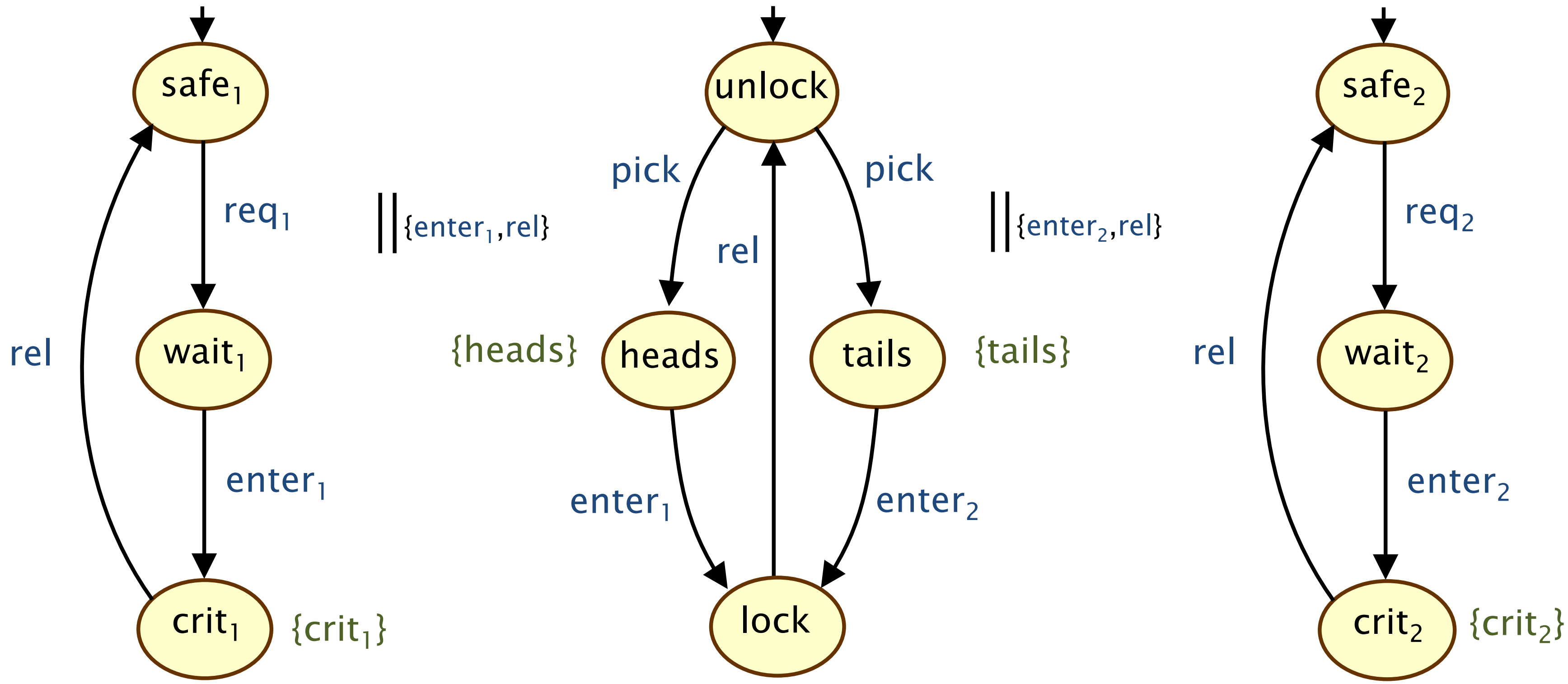
- CTL* syntax
  - $\phi \ ::= \ \text{true} \mid a \mid \phi \wedge \phi \mid \neg\phi \mid \forall\, \psi \mid \exists\, \psi$
  - $\psi \ ::= \ \phi \mid \psi \wedge \psi \mid \neg\psi \mid \bigcirc\, \psi \mid \psi\, \mathsf{U}\, \psi \mid \Diamond\, \psi \mid \Box\, \psi$

- Example
  - $\forall\bigcirc\Box a \wedge \exists\Diamond\Box b$

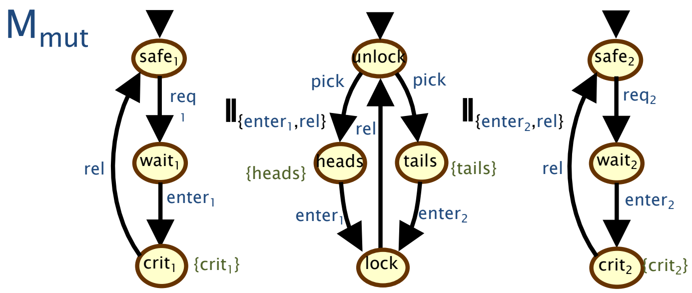


Fairness – motivation

- Rules out (infinite) behaviour considered to be unrealistic
  - often needed in order to verify liveness properties
- Example: two-process mutual exclusion + randomised arbiter
  - properties: $\Box\neg(crit_1 \wedge crit_2)$ and $\Box\Diamond crit_1 \wedge \Box\Diamond crit_2$



Verification under fairness



- For the example M_{mut}:
  - $M_{mut} \not\models \Box\Diamond crit_1 \wedge \Box\Diamond crit_2$
- LTL semantics
  - $M \models \psi \iff trace(\pi) \models \psi$ for every path π of M
- LTL under fairness
  - $M \models_{fair} \psi \iff trace(\pi) \models \psi$ for every fair path π of M
- Many fairness conditions can be expressed in LTL
  - e.g. π is fair $\Leftrightarrow \pi \models fair$ where $fair = \Box\Diamond heads \wedge \Box\Diamond tails$
  - for the example: $M_{mut} \models_{fair} \Box\Diamond crit_1 \wedge \Box\Diamond crit_2$
- LTL verification under fairness
  - $M \models_{fair} \psi \iff M \models (fair \rightarrow \psi)$ (assuming M has no terminal states)

Summary

- Temporal logic
  - extends propositional logic with modal/temporal operators
- Linear temporal logic (LTL)
  - logic for linear time properties (over traces, LTSs)
  - syntax ($\bigcirc$, U, $\Diamond$, $\Box$), semantics, equivalences
- Computation tree logic (CTL)
  - branching-time logic (over states, LTSs)
  - syntax ($\forall\psi$,$\exists\psi$), semantics (computation trees)
- Equivalences, expressiveness, negation, duality
- CTL vs LTL, CTL*, fairness