Probabilistic Model Checking

Alessandro Abate

Lecture 4, p1: Linear-Time Properties

Linear-Time Properties

- consider non-blocking, finite TS
- recall notions of TS path, of TS trace, of reachability sets ($Paths(TS)$, $Reach(TS)$, $Traces(TS)$)

Linear-Time Properties

- consider non-blocking, finite TS
- recall notions of TS path, of TS trace, of reachability sets ($Paths(TS)$, $Reach(TS)$, $Traces(TS)$)

- linear-time properties specify traces that a TS should have (the admissible, desired behaviour of the TS)
- (LTL is a logical formalism to express linear-time properties)

Definition
A linear-time (LT) property over the AP set is a subset of $(2^{AP})^{\omega}$.

Linear-Time Properties

- LT properties can then express requirements over TS traces, properties over all words of TS defined over AP

Definition

Consider a $TS = (S, \rightarrow, I, AP, L)$ and let P be an LT-property over AP. Then, $TS \models P$ iff $Traces(TS) \subseteq P$.
State $s \in S$ satisfies P, namely $s \models P$, whenever $Traces(s) \subseteq P$.

Linear-Time Properties

- LT properties can then express requirements over TS traces, properties over all words of TS defined over AP

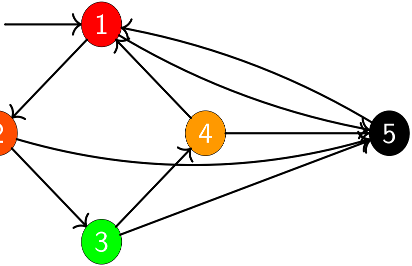
Definition

Consider a $TS = (S, \to, I, AP, L)$ and let P be an LT-property over AP. Then, $TS \models P$ iff $Traces(TS) \subseteq P$.
State $s \in S$ satisfies P, namely $s \models P$, whenever $Traces(s) \subseteq P$.

- given a $TS = (S, \to, I, AP, L)$, an LT property P may depend on symbols in $AP' \subset AP$
- given a path $\pi = s_0 s_1 \ldots$ of TS, we consider $Traces_{AP'}(\pi) = (L(s_0) \cap AP')(L(s_1) \cap AP') \ldots$

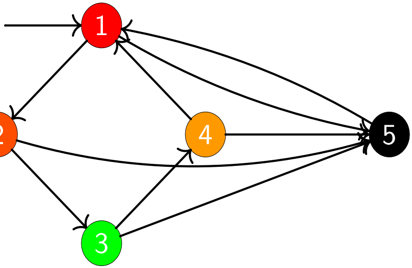
$\Rightarrow$ $Traces_{AP'}(TS)$

Linear-Time Properties: Example



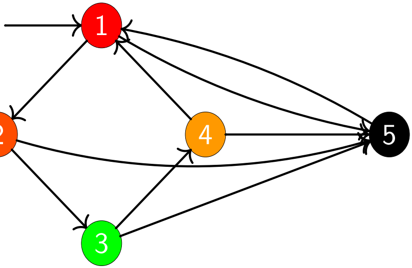
- consider traffic light system, and associated TS model
- recall characterisation of $Traces(TS)$ over AP set

Linear-Time Properties: Example



- consider traffic light system, and associated TS model
- recall characterisation of $\mathit{Traces}(TS)$ over AP set
- $P =$ "eventually, the green light is ON" - does it hold?

Linear-Time Properties: Example



- consider traffic light system, and associated TS model
- recall characterisation of $Traces(TS)$ over AP set
- $P =$ “eventually, the green light is ON” - does it hold?
- $P =$ “eventually, the red light is ON” - does it hold?

Trace Relationship and Linear-Time Properties

- compare two models TS, TS' (with same AP) via their traces

Trace Relationship and Linear-Time Properties

- compare two models TS, TS' (with same AP) via their traces

? trace equivalence: if they have the same traces, do they satisfy the same LT properties?

- if $TS \models P$, then $Traces(TS) \subseteq P$;
  since $Traces(TS) = Traces(TS')$, then $TS' \models P$

Trace Relationship and Linear-Time Properties

- compare two models TS, TS' (with same AP) via their traces

? trace equivalence: if they have the same traces, do they satisfy the same LT properties?

- if $TS \models P$, then $Traces(TS) \subseteq P$;
since $Traces(TS) = Traces(TS')$, then $TS' \models P$
- similarly, if $TS \not\models P$, then there is a trace in TS that is prohibited by P; then, since $Traces(TS) = Traces(TS')$, $TS' \not\models P$

Trace Relationship and Linear-Time Properties

- compare two models TS, TS' (with same AP) via their traces

? trace equivalence: if they have the same traces, do they satisfy the same LT properties?

- if $TS \models P$, then $Traces(TS) \subseteq P$;
  since $Traces(TS) = Traces(TS')$, then $TS' \models P$
- similarly, if $TS \not\models P$, then there is a trace in TS that is prohibited by P; then, since $Traces(TS) = Traces(TS')$, $TS' \not\models P$

- trace inclusion: $Traces(TS) \subseteq Traces(TS')$,
  TS is a correct implementation (a refinement) of TS'
  (TS' is an abstraction of TS)

Trace Relationship and Linear-Time Properties

Definition

TS and TS' are trace equivalent w.r.t. AP if

$$Traces_{AP}(TS) = Traces_{AP}(TS')$$

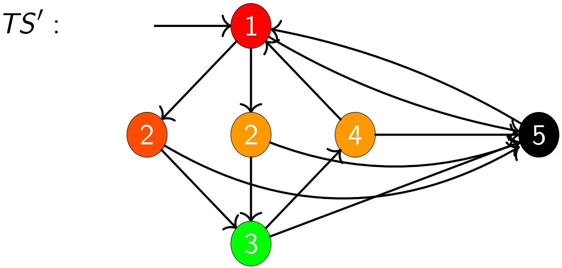
Theorem

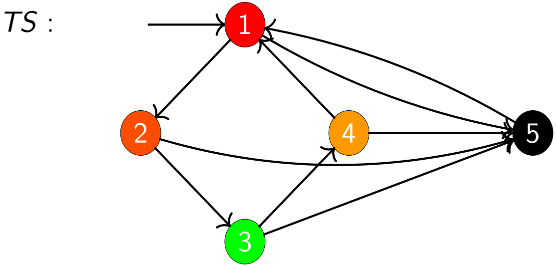
$Traces(TS) = Traces(TS') \Leftrightarrow$
for any LT property P, $TS' \models P \Leftrightarrow TS \models P$
that is, iff TS *and* TS' *satisfy the same set of LT properties*

Theorem

$Traces(TS) \subseteq Traces(TS') \Leftrightarrow$
for any LT property P, $TS' \models P \Rightarrow TS \models P$

Trace Relationship and Linear-Time Properties





- $Traces(TS) \subseteq Traces(TS')$

Linear-Time Properties: Invariants

- a given condition holds always (over entire reach space)

Definition

An LT property P over AP is an invariant if there is a logical formula Φ over AP such that

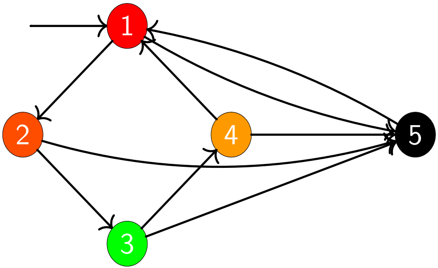
$$P = \left\{ A_0 A_1 A_2 \ldots \in (2^{AP})^\omega \mid \forall j \geq 0, A_j \models \Phi \right\}$$

(Φ is called an invariant condition for P)

- $TS \models P$ iff $\forall \pi \in Paths(TS), Trace(\pi) \in P$
- $TS \models P$ iff $\forall \pi \in Paths(TS), \forall s \in \pi, L(s) \models \Phi$
- $TS \models P$ iff $\forall s \in Reach(TS), L(s) \models \Phi$

$\rightarrow$ checking invariant via reachability analysis

Linear-Time Properties: Invariants



- P = "the traffic light is never simultaneously green and red"
- $\Phi = \neg red \vee \neg green$, so that
  $P = \neg\Diamond(red \wedge green) = \Box(\neg red \vee \neg green)$
- $TS \models P$

Linear-Time Properties: Safety

- nothing bad ever happens

Definition

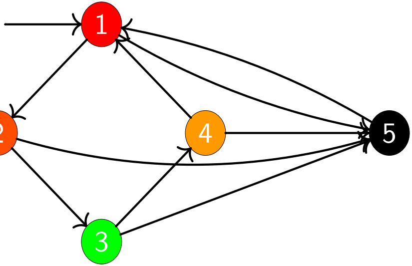
LT property P is a safety property if, for all words $\sigma \in (2^{AP})^\omega \setminus P$, there exists a finite prefix $\hat{\sigma}$ s.t.

$$P \cap \left\{ \sigma' \in (2^{AP})^\omega \mid \hat{\sigma} \text{ is a finite prefix of } \sigma' \right\} = \emptyset$$

$\hat{\sigma}$ is a bad prefix of P

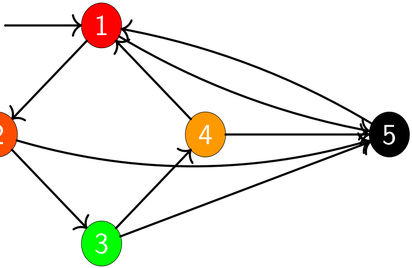
- minimal bad prefix; set of bad prefixes $BadPref(P)$
- any invariant is a safety property
- however, not the opposite (logical formulae can only express state properties)

Linear-Time Properties: Safety



- P = "a green light is always preceded by an amber one"
- P is a safety property
- however, P is not an invariant
- (in this instance $TS \models P$)

Linear-Time Properties: Safety



- P = "a green light is always preceded by an amber one"
- P is a safety property
- however, P is not an invariant
- (in this instance $TS \models P$)
- can you find an LT property that is not a safety one?

Linear-Time Properties: Safety

Theorem

Consider TS and safety property P;
$TS \models P \Leftrightarrow Traces_{fin}(TS) \cap BadPref(P) = \emptyset$
(safety properties are requirements over finite traces)

Theorem

$Traces_{fin}(TS) \subseteq Traces_{fin}(TS') \Leftrightarrow$
for any safety property P, $TS' \models P \Rightarrow TS \models P$

Theorem

$Traces_{fin}(TS) = Traces_{fin}(TS') \Leftrightarrow$
for any safety property P, $TS' \models P \Leftrightarrow TS \models P$
that is, TS *and* TS' *satisfy the same safety properties*

Linear-Time Properties: Safety (alternative definition)

- for trace $\sigma \in (2^{AP})^\omega$,

$$pref(\sigma) = \{\hat{\sigma} \in (2^{AP})^* \mid \hat{\sigma} \text{ is a finite prefix of } \sigma\}$$

- for LT property P, $pref(P) = \cup_{\sigma \in P} pref(\sigma)$
- closure of LT property P:

$$closure(P) = \{\sigma \in (2^{AP})^\omega \mid pref(\sigma) \subseteq pref(P)\}$$

Definition

Let P be an LT property over AP.
Then P is a safety property iff $closure(P) = P$

Linear-Time Properties: Liveness

- something good eventually happens
- property does rules not out any finite prefix,
  namely finite traces cannot elucidate property,
  i.e. any finite prefix can be extended to satisfy property

Definition

LT property P over AP is a liveness property whenever
$\mathit{pref}(P) = (2^{AP})^*$

- eventually; repeated eventually (infinitely often)

- duality of safety and liveness, or
  is there an LT property that is both safe and live?

Linear-Time Properties: Fairness

- used to exclude possible infinite behaviours
- employed to characterise liveness properties
- usually established fairness constraints
  1. *unconditional fairness:* "every transition is infinitely often taken"
  2. *strong fairness:* "if a transition is infinitely often enabled, then it is infinitely often taken"
  3. *weak fairness:* "if a transition is continuously enabled from a certain point in time, then it is infinitely often taken"

Today's reading material

- Sections 3.2–3.5 of
  - Christel Baier and Joost-Pieter Katoen, *Principles of Model Checking*. The MIT Press. Cambridge, MA, USA. 2008.