Safety and Liveness Properties

Lecture #6 of Model Checking

Joost-Pieter Katoen

Lehrstuhl 2: Software Modeling and Verification

E-mail: `katoen@cs.rwth-aachen.de`

November 5, 2008

Overview Lecture #6

⇒ Safety Properties

- Liveness Properties
- Safety versus Liveness Properties
- Classification of Linear-Time Properties

Linear-time properties

- Linear-time properties specify desired traces of a TS
- A *linear-time property* (LT property) over *AP* is a subset of $\left(2^{AP}\right)^{\omega}$
  - i.e., infinite words $A_0 A_1 A_2 \ldots$ with $A_i \subseteq AP$
  - finite words are not needed, as it is assumed that *TS* has no terminal states
- *TS* (over *AP*) *satisfies* LT property P (over *AP*):

$$TS \models P \quad \text{if and only if} \quad Traces(TS) \subseteq P$$

  - *TS* satisfies the LT property P if all its “observable” behaviors are admissible

Invariants

- LT property P_{inv} over *AP* is an *invariant* if it has the form:

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

  - where Φ is a propositional logic formula Φ over *AP*
  - Φ is called an *invariant condition* of P_{inv}

- Note that

$$\begin{array}{lll} TS \models P_{inv} & \text{iff} & \mathit{trace}(\pi) \in P_{inv} \text{ for all paths } \pi \text{ in } TS \\ & \text{iff} & L(s) \models \Phi \text{ for all states } s \text{ that belong to a path of } TS \\ & \text{iff} & L(s) \models \Phi \text{ for all states } s \in \mathit{Reach}(TS) \end{array}$$

- Φ has to be fulfilled by all initial states and
  - satisfaction of Φ is invariant under all transitions in the reachable fragment of *TS*

Safety properties

- Safety properties may impose requirements on finite path fragments
  - and cannot be verified by considering the reachable states only

- A safety property which is not an invariant:
  - consider a cash dispenser, also known as automated teller machine (ATM)
  - property “money can only be withdrawn once a correct PIN has been provided”

  ⇒ not an invariant, since it is not a state property

- But a safety property:
  - any infinite run violating the property has a finite prefix that is “bad”
  - i.e., in which money is withdrawn without issuing a PIN before

Safety properties

- LT property P_{safe} over *AP* is a *safety property* if
  - for all $\sigma \in \left(2^{AP}\right)^{\omega} \setminus P_{safe}$ there exists a finite prefix $\widehat{\sigma}$ of σ such that:

$$P_{safe} \cap \left\{ \sigma' \in \left(2^{AP}\right)^{\omega} \mid \widehat{\sigma} \text{ is a prefix of } \sigma' \right\} \;=\; \varnothing$$

- Path fragment $\widehat{\sigma}$ is a bad prefix of P_{safe}
  - let *BadPref*(P_{safe}) denote the set of bad prefixes of P_{safe}
- Path fragment $\widehat{\sigma}$ is a minimal bad prefix for P_{safe}:
  - if $\widehat{\sigma} \in$ *BadPref*(P_{safe}) and no proper prefix of $\widehat{\sigma}$ is in *BadPref*(P_{safe})

Example safety properties

Safety properties and finite traces

For transition system *TS* without terminal states

and safety property P_{safe}:

$$\mathit{TS} \models P_{safe} \text{ if and only if } \mathit{Traces}_{\mathit{fin}}(\mathit{TS}) \cap \mathit{BadPref}(P_{safe}) = \varnothing$$

Closure

- For trace $\sigma \in \left(2^{AP}\right)^{\omega}$, let *pref*$(\sigma)$ be the set of *finite prefixes* of σ:

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

  - if $\sigma = A_0 A_1 \ldots$ then $\mathit{pref}(\sigma) = \left\{ \varepsilon, A_0, A_0 A_1, A_0 A_1 A_2, \ldots \right\}$

- For property P this is lifted as follows: $\mathit{pref}(P) = \bigcup_{\sigma \in P} \mathit{pref}(\sigma)$

- The *closure* of LT property P:

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

  - the set of infinite traces whose finite prefixes are also prefixes of P, or
  - infinite traces in the closure of P do not have a prefix that is not a prefix of P

Safety properties and closures

For any LT property P over *AP*:

P is a safety property if and only if *closure*$(P) \; = \; P$

Finite trace equivalence and safety properties

For TS and TS' be transition systems (over AP) without terminal states:

$$\mathit{Traces}_{\mathit{fin}}(\mathit{TS}) \;\subseteq\; \mathit{Traces}_{\mathit{fin}}(\mathit{TS}')$$

if and only if

for any safety property P_{safe} : $\mathit{TS}' \models P_{safe} \;\Rightarrow\; \mathit{TS} \models P_{safe}$

$$\mathit{Traces}_{\mathit{fin}}(\mathit{TS}) = \mathit{Traces}_{\mathit{fin}}(\mathit{TS}')$$

if and only if

TS and TS' satisfy the same safety properties

Finite vs. infinite traces

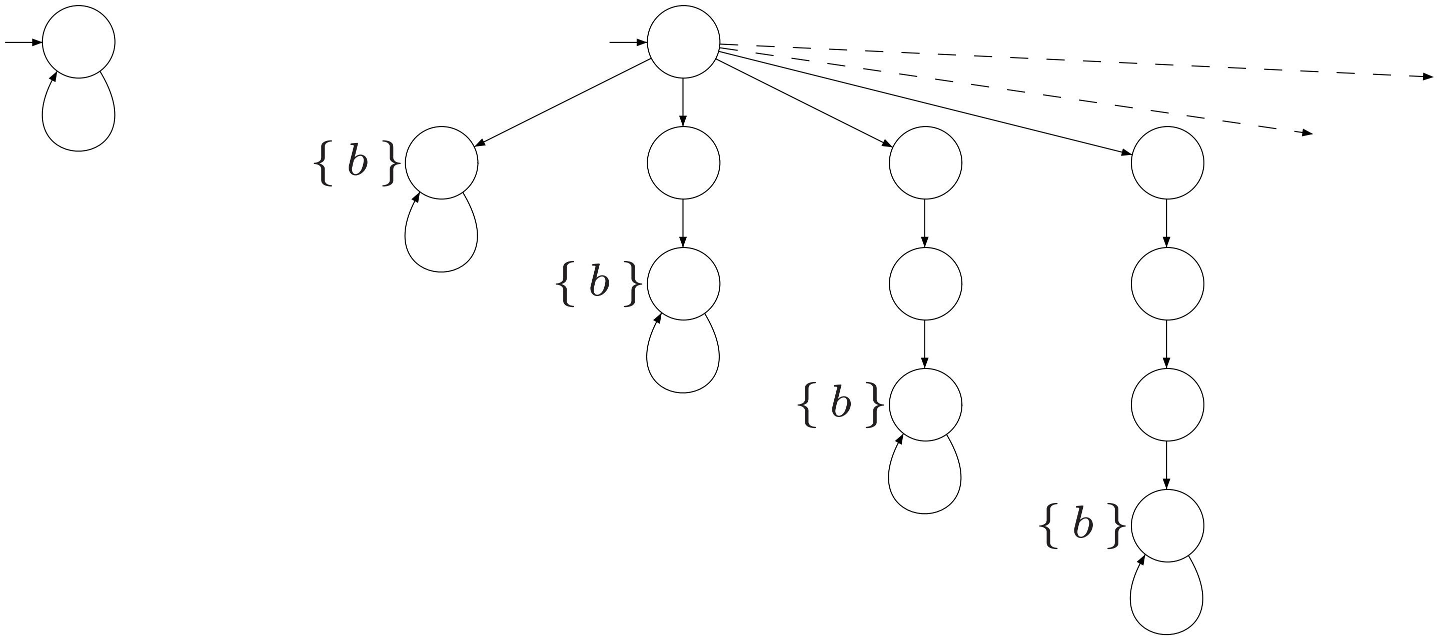
For *TS* without terminal states and finite *TS*′:

$$\mathit{Traces}(TS) \subseteq \mathit{Traces}(TS') \quad \text{iff} \quad \mathit{Traces}_{\mathit{fin}}(TS) \subseteq \mathit{Traces}_{\mathit{fin}}(TS')$$

this does not hold for infinite TS′ (cf. next slide)

but also holds for image-finite TS′

Trace inclusion $\neq$ finite trace inclusion



$\mathit{Traces}(\mathit{TS}) \not\subseteq \mathit{Traces}(\mathit{TS}')$ and $\mathit{Traces}_{\mathit{fin}}(\mathit{TS}) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathit{TS}')$

Why liveness?

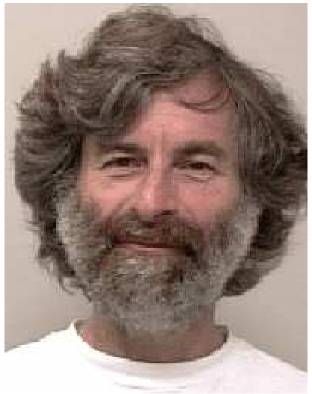
- Safety properties specify that “something bad never happens”
- Doing nothing easily fulfills a safety property
  - as this will never lead to a “bad” situation

⇒ Safety properties are complemented by liveness properties

  - that require some progress

- Liveness properties assert that:
  - ”something good” will happen eventually [Lamport 1977]

The meaning of liveness



[Lamport 2000]

The question of whether a real system satisfies a liveness property is meaningless; it can be answered only by observing the system for an infinite length of time, and real systems don't run forever.

Liveness is always an approximation to the property we really care about. We want a program to terminate within 100 years, but proving that it does would require addition of distracting timing assumptions.

So, we prove the weaker condition that the program eventually terminates. This doesn't prove that the program will terminate within our lifetimes, but it does demonstrate the absence of infinite loops.

Liveness properties

LT property P_{live} over *AP* is a *liveness* property whenever

$$\textit{pref}(P_{live}) = \left(2^{\textit{AP}}\right)^*$$

- A liveness property is an LT property
  - that *does not rule out any prefix*
- Liveness properties are violated in “infinite time”
  - whereas safety properties are violated in finite time
  - finite traces are of no use to decide whether P holds or not
  - any finite prefix can be extended such that the resulting infinite trace satisfies P

Example liveness properties

- “If the tank is empty, the outlet valve will eventually be closed”
- “If the outlet valve is open and the request signal disappears, the outlet valve will eventually be closed”
- “If the tank is full and a request is present, the outlet valve will eventually be opened”
- “The program terminates within 31 computational steps”

$\Rightarrow$ a finite trace may violate this; this is a safety property!

- “The program eventually terminates”

Liveness properties for mutual exclusion

- Eventually:
  - each process will eventually enter its critical section
- Repeated eventually:
  - each process will enter ist critical section infinitely often
- Starvation freedom:
  - each waiting process will eventually enter its critical section

how to formalize these properties?

Liveness properties for mutual exclusion

$P = \{ A_0 A_1 A_2 \ldots \mid A_j \subseteq AP \wedge \ldots \}$ and $AP = \{wait_1, crit_1, wait_2, crit_2\}$

- Eventually:

$$(\exists j \geqslant 0.\ crit_1 \in A_j) \ \wedge \ (\exists j \geqslant 0.\ crit_2 \in A_j)$$

- Repeated eventually:

$$\left(\overset{\infty}{\exists}\ j \geqslant 0.\ crit_1 \in A_j\right) \ \wedge \ \left(\overset{\infty}{\exists}\ j \geqslant 0.\ crit_2 \in A_j\right)$$

- Starvation freedom:

$$\begin{aligned} &\forall j \geqslant 0.\ (wait_1 \in A_j \ \Rightarrow \ (\exists k > j.\ crit_1 \in A_k)) \ \wedge \\ &\forall j \geqslant 0.\ (wait_2 \in A_j \ \Rightarrow \ (\exists k > j.\ crit_2 \in A_k)) \end{aligned}$$

Safety vs. liveness

- Are safety and liveness properties disjoint? Yes
- Is any linear-time property a safety or liveness property? No
- But:

for any LT property P an equivalent LT property P' exists

which is a conjunction of a safety and a liveness property

⇒ *safety and liveness provide an essential characterization of LT properties*

Basic properties

If P (over *AP*) is both a safety and a liveness property then:

$$P \; = \; \left(2^{\mathit{AP}}\right)^{\omega}$$

For any LT properties P and P':

$$\mathit{closure}(P \cup P') = \mathit{closure}(P) \, \cup \, \mathit{closure}(P')$$

let's consider the proofs of these facts

A non-safety and non-liveness property

“the machine provides infinitely often beer
after initially providing sprite three times in a row”

- This property consists of *two* parts:
  - it requires beer to be provided infinitely often

  ⇒ as any finite trace fulfills this, it is a liveness property
  - the first three drinks it provides should all be sprite

  ⇒ bad prefix = one of first three drinks is beer; this is a safety property

- Property is thus a conjunction of a safety *and* a liveness property

does this apply to all such properties?

Decomposition theorem

For any LT property P over *AP* there exists

a safety property P_{safe} and a liveness property P_{live}

(both over *AP*) such that:

$$P = P_{safe} \cap P_{live}$$

Proposal: $P = \underbrace{\mathit{closure}(P)}_{=P_{safe}} \cap \underbrace{\left(P \cup \left(\left(2^{AP}\right)^{\omega} \setminus \mathit{closure}(P)\right)\right)}_{=P_{live}}$

Proof

"Sharpest" decomposition theorem

Let P be an LT property and $P = P_{safe} \cap P_{live}$

where P_{safe} is a safety property and P_{live} a liveness property.

Then:

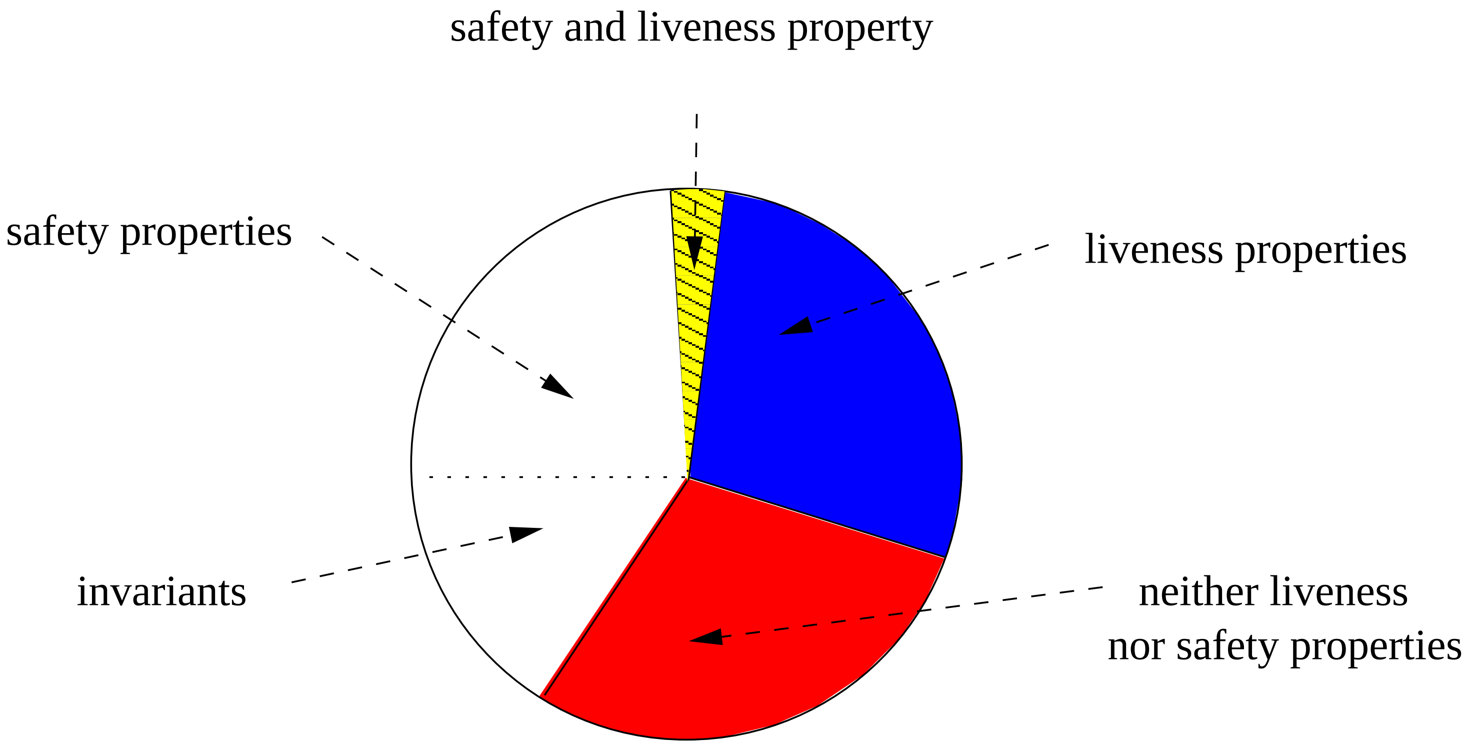
1. *closure*$(P) \subseteq P_{safe}$

2. $P_{live} \subseteq P \cup \left(\left(2^{AP} \right)^{\omega} \setminus \mathit{closure}(P) \right)$

closure(P) is the <u>strongest</u> safety property and

$\left(\left(2^{AP} \right)^{\omega} \setminus \mathit{closure}(P) \right)$ the <u>weakest</u> liveness property

Classification of LT properties



Summary LT properties

- LT properties are finite sets of infinite words over 2^{AP} (= traces)

- An invariant requires a condition Φ to hold in any reachable state

- Each trace refuting a safety property has a finite prefix causing this
  - invariants are safety properties with bad prefix $\Phi^*(\neg\Phi)$
  - a safety property is regular iff its set of bad prefixes is a regular language

  $\Rightarrow$ safety properties constrain finite behaviors

- A liveness property does not rule out finite behaviour

  $\Rightarrow$ liveness properties constrain infinite behaviors

- Any LT property is equivalent to a conjunction of a safety and a liveness property