Overview

Introduction

Modelling parallel systems

Linear Time Properties

- state-based and linear time view
- definition of linear time properties
- invariants and safety ⟵
- liveness and fairness

Regular Properties

Linear Temporal Logic

Computation-Tree Logic

Equivalences and Abstraction

Invariant

Invariant

Let E be an LT property over AP.

E is called an invariant if there exists a propositional formula Φ over AP such that

$$E = \left\{ A_0 A_1 A_2 \ldots \in \left(2^{AP}\right)^{\omega} : \forall i \geq 0.\, A_i \models \Phi \right\}$$

Let E be an LT property over AP.

E is called an invariant if there exists a propositional formula Φ over AP such that

$$E = \left\{ A_0 A_1 A_2 \ldots \in \left(2^{AP}\right)^{\omega} : \forall i \geq 0.\, A_i \models \Phi \right\}$$

Φ is called the invariant condition of E.

Safety properties

state that "nothing bad will happen"

Safety properties

state that "nothing bad will happen"

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: e.g., for dining philosophers
  never $\bigwedge_{0 \leq i < n} wait_i$

Safety properties

state that "nothing bad will happen"

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: e.g., for dining philosophers *never* $\bigwedge_{0 \leq i < n} wait_i$
- German traffic lights:
  every red phase is preceded by a yellow phase

Safety properties

state that "nothing bad will happen"

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: e.g., for dining philosophers
  never $\bigwedge_{0 \leq i < n} wait_i$
- German traffic lights:
  every red phase is preceded by a yellow phase
- beverage machine:
  no drink must be released if the user did not enter a coin before

  the total number of entered coins is never less than the total number of released drinks

Safety properties

state that "nothing bad will happen"

invariants:

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: *never* $\bigwedge_{0 \leq i < n} wait_i$

other safety properties:

- German traffic lights:
  every red phase is preceded by a yellow phase
- beverage machine:
  the total number of entered coins is never less than the total number of released drinks

Safety properties

state that "nothing bad will happen"

invariants: ⟵ "no **bad state** will be reached"

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: *never* $\bigwedge_{0 \leq i < n} wait_i$

---

other safety properties:

- German traffic lights:

  every red phase is preceded by a yellow phase

- beverage machine:

  the total number of entered coins is never less than the total number of released drinks

Safety properties

state that "nothing bad will happen"

invariants: ⟵ "no **bad state** will be reached"

- mutual exclusion: *never* $crit_1 \wedge crit_2$
- deadlock freedom: *never* $\bigwedge_{0 \le i < n} wait_i$

other safety properties: ⟵ "no **bad prefix**"

- German traffic lights:
  every red phase is preceded by a yellow phase
- beverage machine:
  the total number of entered coins is never less than the total number of released drinks

Bad prefixes of safety properties

- traffic lights:

  every red phase is preceded by a yellow phase

Bad prefixes of safety properties

- traffic lights:

every red phase is preceded by a yellow phase

$\uparrow$

bad prefix: finite trace fragment where a red phase appears without being preceded by a yellow phase
e.g., $\ldots \{●\}\{●\}$

Bad prefixes of safety properties

- traffic lights:

  every red phase is preceded by a yellow phase

  $\uparrow$

  bad prefix: finite trace fragment where a red phase appears without being preceded by a yellow phase

  e.g., $\ldots \{\color{green}\bullet\}\,\{\color{red}\bullet\}$

- beverage machine:

  the total number of entered coins is never less than the total number of released drinks

Bad prefixes of safety properties

- traffic lights:

every red phase is preceded by a yellow phase

$\uparrow$

bad prefix: finite trace fragment where a red phase appears without being preceded by a yellow phase
e.g., $\ldots$ {●} {●}

- beverage machine:

the total number of entered coins is never less than the total number of released drinks

$\uparrow$

bad prefix, e.g., {*pay*} {*drink*} {*drink*}

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma = A_0 A_1 A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0 A_1 \ldots A_n$ of σ such that *none* of the words $A_0 A_1 \ldots A_n B_{n+1} B_{n+2} B_{n+3} \ldots$ belongs to E

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma = A_0 A_1 A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0 A_1 \ldots A_n$ of σ such that *none* of the words $A_0 A_1 \ldots A_n B_{n+1} B_{n+2} B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma = A_0 A_1 A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0 A_1 \ldots A_n$ of σ such that *none* of the words $A_0 A_1 \ldots A_n B_{n+1} B_{n+2} B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Such words $A_0 A_1 \ldots A_n$ are called bad prefixes for E.

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma \;=\; A_0\, A_1\, A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0\, A_1 \ldots A_n$ of σ such that *none* of the words $A_0\, A_1 \ldots A_n\, B_{n+1}\, B_{n+2}\, B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Such words $A_0\, A_1 \ldots A_n$ are called bad prefixes for E.

$E =$ set of all infinite words that do *not* have a bad prefix

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma \;=\; A_0\,A_1\,A_2\ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0\,A_1\ldots A_n$ of σ such that *none* of the words $A_0\,A_1\ldots A_n\,B_{n+1}\,B_{n+2}\,B_{n+3}\ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0\ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Such words $A_0\,A_1\ldots A_n$ are called bad prefixes for E.

$$BadPref_E \stackrel{\text{def}}{=} \text{set of bad prefixes for } E$$

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma = A_0 A_1 A_2 \ldots \in (2^{AP})^\omega \setminus E$$

there exists a finite prefix $A_0 A_1 \ldots A_n$ of σ such that *none* of the words $A_0 A_1 \ldots A_n B_{n+1} B_{n+2} B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{ \sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma' \right\} = \varnothing$$

Such words $A_0 A_1 \ldots A_n$ are called bad prefixes for E.

$$BadPref_E \stackrel{\text{def}}{=} \text{set of bad prefixes for } E \subseteq \left(2^{AP}\right)^+$$

Definition of safety properties

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma \; = \; A_0\, A_1\, A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

there exists a finite prefix $A_0\, A_1 \ldots A_n$ of σ such that *none* of the words $A_0\, A_1 \ldots A_n\, B_{n+1}\, B_{n+2}\, B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Such words $A_0\, A_1 \ldots A_n$ are called bad prefixes for E.

$BadPref_E \stackrel{\text{def}}{=}$ set of bad prefixes for $E \;\subseteq \left(2^{AP}\right)^+$

$\uparrow$

briefly: $BadPref$

Let E be a LT property over AP, i.e., $E \subseteq (2^{AP})^\omega$.

E is called a safety property if for all words

$$\sigma \;=\; A_0\, A_1\, A_2 \ldots \in \left(2^{AP}\right)^\omega \setminus E$$

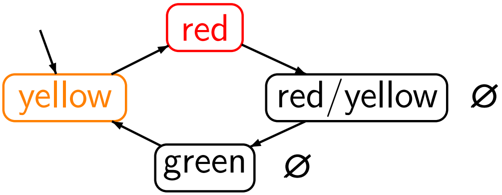
there exists a finite prefix $A_0\, A_1 \ldots A_n$ of σ such that *none* of the words $A_0\, A_1 \ldots A_n\, B_{n+1}\, B_{n+2}\, B_{n+3} \ldots$ belongs to E, i.e.,

$$E \cap \left\{\sigma' \in (2^{AP})^\omega : A_0 \ldots A_n \text{ is a prefix of } \sigma'\right\} = \varnothing$$

Such words $A_0\, A_1 \ldots A_n$ are called bad prefixes for E.

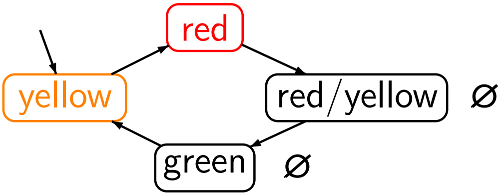
minimal bad prefixes: any word $A_0 \ldots A_i \ldots A_n \in \mathit{BadPref}$ s.t. no proper prefix $A_0 \ldots A_i$ is a bad prefix for E

Safety property for a traffic light



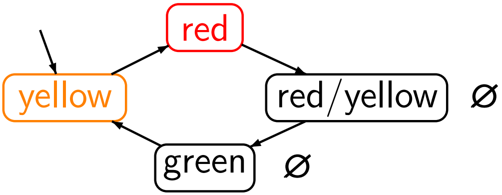
$AP = \{red, yellow\}$

Safety property for a traffic light



"every red phase is preceded by a yellow phase"

Safety property for a traffic light



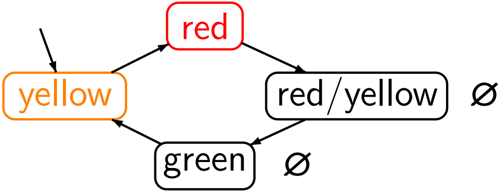
"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

E = set of all infinite words $A_0\, A_1\, A_2 \ldots$ over 2^{AP} such that for all $i \in \mathbb{N}$:

$$red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$

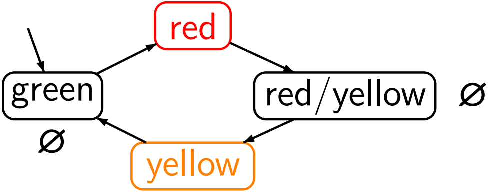
Safety property for a traffic light



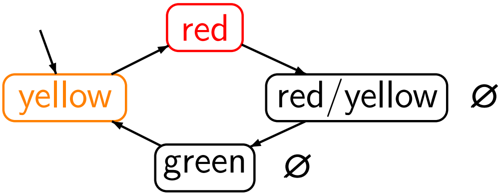
"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

$$
\begin{array}{rcl}
E & = & \text{set of all infinite words } A_0\, A_1\, A_2 \ldots \\
 & & \text{over } 2^{AP} \text{ such that for all } i \in \mathbb{N}\text{:} \\
 & & \quad \mathit{red} \in A_i \implies i \geq 1 \text{ and } \mathit{yellow} \in A_{i-1}
\end{array}
$$



Safety property for a traffic light

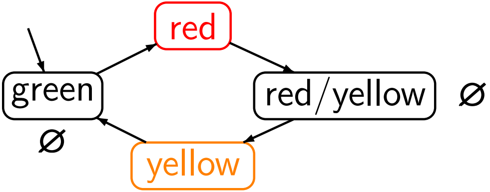


"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

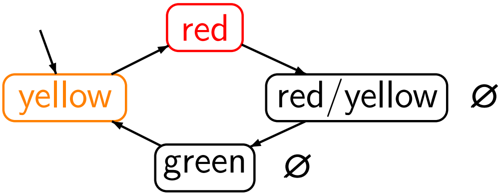
E = set of all infinite words $A_0\, A_1\, A_2 \ldots$ over 2^{AP} such that for all $i \in \mathbb{N}$:

$$red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$



"there is a red phase that is not preceded by a yellow phase"

Safety property for a traffic light

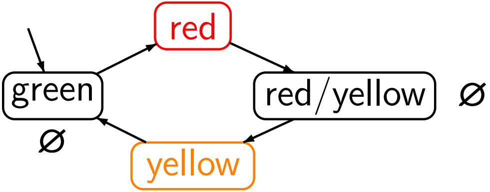


"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

E = set of all infinite words $A_0\, A_1\, A_2 \ldots$ over 2^{AP} such that for all $i \in \mathbb{N}$:

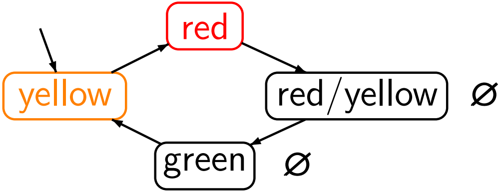
$$red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$



"there is a red phase that is not preceded by a yellow phase"

hence: $\mathcal{T} \not\models E$

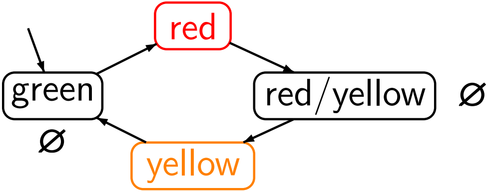
Safety property for a traffic light



"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

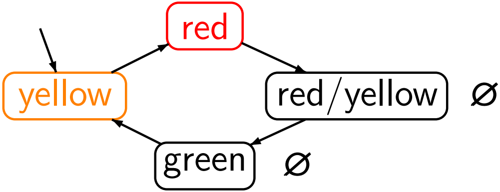
$$E \;=\; \text{set of all infinite words } A_0\,A_1\,A_2\ldots \text{ over } 2^{AP} \text{ such that for all } i \in \mathbb{N}\text{:}\quad red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$



$\mathcal{T} \not\models E$

bad prefix, e.g., $\varnothing\,\{red\}\,\varnothing\,\{yellow\}$

Safety property for a traffic light

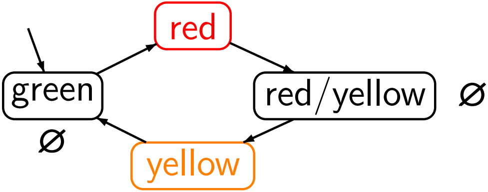


"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

E = set of all infinite words $A_0\, A_1\, A_2 \ldots$ over 2^{AP} such that for all $i \in \mathbb{N}$:

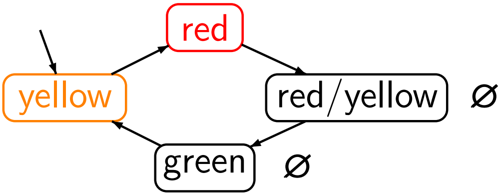
$$red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$



$\mathcal{T} \not\models E$

minimal bad prefix: $\varnothing\,\{red\}$

Safety property for a traffic light



"every red phase is preceded by a yellow phase"

hence: $\mathcal{T} \models E$

$$E = \text{set of all infinite words } A_0 A_1 A_2 \ldots \text{ over } 2^{AP} \text{ such that for all } i \in \mathbb{N}\text{:}$$
$$red \in A_i \implies i \geq 1 \text{ and } yellow \in A_{i-1}$$

is a safety property over $AP = \{red, yellow\}$ with

$$BadPref = \text{set of all finite words } A_0 A_1 \ldots A_n \text{ over } 2^{AP} \text{ s.t. for some } i \in \{0, \ldots, n\}\text{:}$$
$$red \in A_i \wedge (i{=}0 \vee yellow \notin A_{i-1})$$

Satisfaction of safety properties

IS2.5-11A

Satisfaction of safety properties

Let $E \subseteq (2^{AP})^\omega$ be a safety property, $\mathcal{T}$ a TS over AP.

$$\boxed{\mathcal{T} \models E \quad \text{iff} \quad \mathit{Traces}(\mathcal{T}) \subseteq E}$$

$\mathit{Traces}(\mathcal{T}) \quad = \quad$ set of traces of $\mathcal{T}$

Let $E \subseteq (2^{AP})^\omega$ be a safety property, $\mathcal{T}$ a TS over AP.

$$\boxed{\begin{array}{lll} \mathcal{T} \models E & \text{iff} & \mathit{Traces}(\mathcal{T}) \subseteq E \\ & \text{iff} & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathit{BadPref} = \varnothing \end{array}}$$

$$\begin{array}{lll} \mathit{BadPref} & = & \text{set of all bad prefixes of } E \\ & & \\ \mathit{Traces}(\mathcal{T}) & = & \text{set of traces of } \mathcal{T} \\ \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) & = & \text{set of finite traces of } \mathcal{T} \end{array}$$

$= \{\ \mathit{trace}(\widehat{\pi}) : \widehat{\pi}$ is an initial, finite path fragment of $\mathcal{T} \}$

Satisfaction of safety properties

Let $E \subseteq (2^{AP})^\omega$ be a safety property, $\mathcal{T}$ a TS over AP.

$$\boxed{\begin{array}{rll} \mathcal{T} \models E & \text{iff} & \mathit{Traces}(\mathcal{T}) \subseteq E \\ & \text{iff} & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathit{BadPref} = \varnothing \\ & \text{iff} & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathit{MinBadPref} = \varnothing \end{array}}$$

$\mathit{BadPref}$ = set of all bad prefixes of E

$\mathit{MinBadPref}$ = set of all minimal bad prefixes of E

$\mathit{Traces}(\mathcal{T})$ = set of traces of $\mathcal{T}$

$\mathit{Traces}_{\mathit{fin}}(\mathcal{T})$ = set of finite traces of $\mathcal{T}$

$= \{\ \mathit{trace}(\widehat{\pi}) : \widehat{\pi}$ is an initial, finite path fragment of $\mathcal{T}\}$

Correct or wrong?

Every invariant is a safety property.

Correct or wrong?

Every invariant is a safety property.

correct.

Correct or wrong?

Every invariant is a safety property.

correct.

Let E be an invariant with invariant condition Φ.

Every invariant is a safety property.

correct.

Let E be an invariant with invariant condition Φ.

- bad prefixes for E: finite words $A_0 \ldots A_i \ldots A_n$ s.t.

$$A_i \not\models \Phi \text{ for some } i \in \{0, 1, \ldots, n\}$$

Every invariant is a safety property.

correct.

Let E be an invariant with invariant condition Φ.

- bad prefixes for E: finite words $A_0 \ldots A_i \ldots A_n$ s.t.

$$A_i \not\models \Phi \text{ for some } i \in \{0, 1, \ldots, n\}$$

- minimal bad prefixes for E:
finite words $A_0\, A_1 \ldots A_{n-1}\, A_n$ such that

$$A_i \models \Phi \text{ for } i = 0, 1, \ldots, n{-}1, \text{ and } A_n \not\models \Phi$$

Correct or wrong?

$\varnothing$ is a safety property

Correct or wrong?

$\boxed{\varnothing \text{ is a safety property}}$

correct

Correct or wrong?

$\varnothing$ is a safety property

correct

- all finite words $A_0 \dots A_n \in (2^{AP})^+$ are bad prefixes

Correct or wrong?

$\boxed{\varnothing \text{ is a safety property}}$

correct

- all finite words $A_0 \ldots A_n \in (2^{AP})^+$ are bad prefixes
- $\varnothing$ is even an invariant (invariant condition ***false***)

Correct or wrong?

$\varnothing$ is a safety property

correct

- all finite words $A_0 \ldots A_n \in (2^{AP})^+$ are bad prefixes
- $\varnothing$ is even an invariant (invariant condition ***false***)

$(2^{AP})^\omega$ is a safety property

$\varnothing$ is a safety property

correct

- all finite words $A_0 \ldots A_n \in (2^{AP})^+$ are bad prefixes
- $\varnothing$ is even an invariant (invariant condition ***false***)

$(2^{AP})^\omega$ is a safety property

correct

$\varnothing$ is a safety property

correct

- all finite words $A_0 \ldots A_n \in (2^{AP})^+$ are bad prefixes
- $\varnothing$ is even an invariant (invariant condition ***false***)

$(2^{AP})^\omega$ is a safety property

correct

$$\text{“For all words} \in \underbrace{(2^{AP})^\omega \setminus (2^{AP})^\omega}_{=\varnothing} \ldots\text{”}$$

Prefix closure

Prefix closure

For a given infinite word $\sigma = A_0\, A_1\, A_2\, \ldots$, let

$$pref(\sigma) \stackrel{\text{def}}{=} \text{set of all nonempty, finite prefixes of } \sigma$$

For a given infinite word $\sigma = A_0\,A_1\,A_2\,\ldots$, let

$$\begin{aligned} \mathit{pref}(\sigma) &\stackrel{\text{def}}{=} \text{set of all nonempty, finite prefixes of } \sigma \\ &= \{\, A_0\,A_1\,\ldots A_n : n \geq 0 \,\} \end{aligned}$$

Prefix closure

For a given infinite word $\sigma = A_0\, A_1\, A_2\, \ldots$, let

$$\begin{aligned} \mathit{pref}(\sigma) &\stackrel{\text{def}}{=} \text{set of all nonempty, finite prefixes of } \sigma \\ &= \{\, A_0\, A_1 \ldots A_n : n \geq 0 \,\} \end{aligned}$$

Prefix closure

For a given infinite word $\sigma = A_0\,A_1\,A_2\,\ldots$, let

$$\begin{aligned} \mathit{pref}(\sigma) &\stackrel{\text{def}}{=} \text{set of all nonempty, finite prefixes of } \sigma \\ &= \big\{ A_0\,A_1\,\ldots A_n \,:\, n \geq 0 \big\} \end{aligned}$$

For $E \subseteq \big(2^{AP}\big)^{\omega}$, let $\mathit{pref}(E) \stackrel{\text{def}}{=} \bigcup\limits_{\sigma \in E} \mathit{pref}(\sigma)$

Prefix closure

For a given infinite word $\sigma = A_0\, A_1\, A_2\, \ldots$, let

$$pref(\sigma) \overset{\text{def}}{=} \text{set of all nonempty, finite prefixes of } \sigma$$
$$= \{ A_0\, A_1\, \ldots A_n : n \geq 0 \}$$

For $E \subseteq (2^{AP})^\omega$, let $pref(E) \overset{\text{def}}{=} \bigcup\limits_{\sigma \in E} pref(\sigma)$

Given an LT property E, the prefix closure of E is:

$$cl(E) \overset{\text{def}}{=} \{\sigma \in (2^{AP})^\omega : pref(\sigma) \subseteq pref(E)\}$$

Prefix closure and safety

For any infinite word $\sigma \in \left(2^{AP}\right)^{\omega}$, let

$$pref(\sigma) \;=\; \text{set of all nonempty, finite prefixes of } \sigma$$

For any LT property $E \subseteq \left(2^{AP}\right)^{\omega}$, let

$$\begin{aligned} pref(E) &= \bigcup_{\sigma \in E} pref(\sigma) \text{ and} \\ cl(E) &= \left\{\sigma \in (2^{AP})^{\omega} : pref(\sigma) \subseteq pref(E)\right\} \end{aligned}$$

Prefix closure and safety

For any infinite word $\sigma \in \left(2^{AP}\right)^{\omega}$, let

$$pref(\sigma) \;=\; \text{set of all nonempty, finite prefixes of } \sigma$$

For any LT property $E \subseteq \left(2^{AP}\right)^{\omega}$, let

$$pref(E) \;=\; \bigcup_{\sigma \in E} pref(\sigma) \text{ and}$$

$$cl(E) \;=\; \left\{\sigma \in (2^{AP})^{\omega} : pref(\sigma) \subseteq pref(E)\right\}$$

Theorem:

E is a safety property iff $cl(E) = E$

Safety and finite trace inclusion

remind: LT properties and trace inclusion:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces(\mathcal{T}_1) \subseteq Traces(\mathcal{T}_2)$$

iff for all LT properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Safety and finite trace inclusion

remind: LT properties and trace inclusion:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces(\mathcal{T}_1) \subseteq Traces(\mathcal{T}_2)$$

iff for all LT properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

safety properties and finite trace inclusion:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longrightarrow$": obvious, as for safety property E:

$$\mathcal{T} \models E \quad \text{iff} \quad Traces_{fin}(\mathcal{T}) \cap BadPref = \varnothing$$

$$\mathit{Traces}_{fin}(\mathcal{T}_1) \subseteq \mathit{Traces}_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longrightarrow$": obvious, as for safety property E:

$$\mathcal{T} \models E \quad \text{iff} \quad \mathit{Traces}_{fin}(\mathcal{T}) \cap \mathit{BadPref} = \varnothing$$

Hence:

If $\mathcal{T}_2 \models E$ and $\mathit{Traces}_{fin}(\mathcal{T}_1) \subseteq \mathit{Traces}_{fin}(\mathcal{T}_2)$ then:

Safety and finite trace inclusion

$$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}_1) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longrightarrow$": obvious, as for safety property E:

$$\mathcal{T} \models E \quad \text{iff} \quad \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \cap \mathit{BadPref} = \varnothing$$

Hence:

If $\mathcal{T}_2 \models E$ and $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}_1) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}_2)$ then:

$$\begin{aligned} & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}_1) \cap \mathit{BadPref} \\ \subseteq \ & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}_2) \cap \mathit{BadPref} = \varnothing \end{aligned}$$

Safety and finite trace inclusion

$$\mathit{Traces}_{fin}(\mathcal{T}_1) \subseteq \mathit{Traces}_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longrightarrow$": obvious, as for safety property E:

$$\mathcal{T} \models E \quad \text{iff} \quad \mathit{Traces}_{fin}(\mathcal{T}) \cap \mathit{BadPref} = \varnothing$$

Hence:

If $\mathcal{T}_2 \models E$ and $\mathit{Traces}_{fin}(\mathcal{T}_1) \subseteq \mathit{Traces}_{fin}(\mathcal{T}_2)$ then:

$$\boxed{\begin{aligned} &\mathit{Traces}_{fin}(\mathcal{T}_1) \cap \mathit{BadPref} \\ \subseteq\ &\mathit{Traces}_{fin}(\mathcal{T}_2) \cap \mathit{BadPref} = \varnothing \end{aligned}}$$

and therefore $\mathcal{T}_1 \models E$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2))$$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

for each transition system $\mathcal{T}$:

$$pref(Traces(\mathcal{T})) = Traces_{fin}(\mathcal{T})$$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property

as $cl(E) = E$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property

$\uparrow$

as $cl(E) = E$

set of bad prefixes: $(2^{AP})^+ \setminus Traces_{fin}(\mathcal{T}_2)$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$

Safety and finite trace inclusion

$$\mathit{Traces_{fin}}(\mathcal{T}_1) \subseteq \mathit{Traces_{fin}}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = \mathit{cl}(\mathit{Traces}(\mathcal{T}_2)) = \big\{\sigma : \mathit{pref}(\sigma) \subseteq \mathit{Traces_{fin}}(\mathcal{T}_2)\big\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$ and therefore $\mathit{Traces}(\mathcal{T}_1) \subseteq E$.

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$ and therefore $Traces(\mathcal{T}_1) \subseteq E$.

Hence: $Traces_{fin}(\mathcal{T}_1) = pref(Traces(\mathcal{T}_1))$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$ and therefore $Traces(\mathcal{T}_1) \subseteq E$.

Hence: $Traces_{fin}(\mathcal{T}_1) = pref(Traces(\mathcal{T}_1))$

$$\subseteq pref(E)$$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$ and therefore $Traces(\mathcal{T}_1) \subseteq E$.

Hence: $Traces_{fin}(\mathcal{T}_1) = pref(Traces(\mathcal{T}_1))$

$$\subseteq pref(E) = pref(cl(Traces(\mathcal{T}_2)))$$

Safety and finite trace inclusion

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Proof "$\Longleftarrow$": consider the LT property

$$E = cl(Traces(\mathcal{T}_2)) = \{\sigma : pref(\sigma) \subseteq Traces_{fin}(\mathcal{T}_2)\}$$

Then, E is a safety property and $\mathcal{T}_2 \models E$.

By assumption: $\mathcal{T}_1 \models E$ and therefore $Traces(\mathcal{T}_1) \subseteq E$.

Hence:
$$\begin{aligned} Traces_{fin}(\mathcal{T}_1) &= pref(Traces(\mathcal{T}_1)) \\ &\subseteq pref(E) = pref(cl(Traces(\mathcal{T}_2))) \\ &= Traces_{fin}(\mathcal{T}_2) \end{aligned}$$

Safety and finite trace equivalence

Safety and finite trace equivalence

safety properties and finite trace inclusion:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

Safety and finite trace equivalence

safety properties and finite trace inclusion:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces_{fin}(\mathcal{T}_1) \subseteq Traces_{fin}(\mathcal{T}_2)$$

iff for all safety properties E: $\mathcal{T}_2 \models E \implies \mathcal{T}_1 \models E$

safety properties and finite trace equivalence:

If $\mathcal{T}_1$ and $\mathcal{T}_2$ are TS over AP then:

$$Traces_{fin}(\mathcal{T}_1) = Traces_{fin}(\mathcal{T}_2)$$

iff $\mathcal{T}_1$ and $\mathcal{T}_2$ satisfy the same safety properties

Summary: trace relations and properties

trace inclusion

$Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$ iff

for all LT properties E: $\mathcal{T}' \models E \Longrightarrow \mathcal{T} \models E$

finite trace inclusion

$Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$ iff

for all safety properties E: $\mathcal{T}' \models E \Longrightarrow \mathcal{T} \models E$

trace equivalence

$\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ iff

$\mathcal{T}$ and $\mathcal{T}'$ satisfy the same LT properties

finite trace equivalence

$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) = \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$ iff

$\mathcal{T}$ and $\mathcal{T}'$ satisfy the same safety properties

correct or wrong?

If $\mathit{Traces}(\mathcal{T}) \subseteq \mathit{Traces}(\mathcal{T}')$
then $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$.

If $\mathit{Traces}(\mathcal{T}) \subseteq \mathit{Traces}(\mathcal{T}')$
then $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$.

correct, since

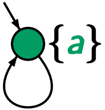
$$\begin{aligned} \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) &= \text{set of all finite nonempty prefixes of words in } \mathit{Traces}(\mathcal{T}) \\ &= \mathit{pref}(\mathit{Traces}(\mathcal{T})) \end{aligned}$$

correct or wrong?

If $\mathit{Traces}(\mathcal{T}) \subseteq \mathit{Traces}(\mathcal{T}')$
then $\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$.

correct, since

$$\begin{aligned} \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) &= \text{set of all finite nonempty prefixes} \\ & \text{of words in } \mathit{Traces}(\mathcal{T}) \\ &= \mathit{pref}(\mathit{Traces}(\mathcal{T})) \end{aligned}$$



$$\begin{aligned} \mathit{Traces}(\mathcal{T}) &= \{ \{a\}^{\omega} \} \\ \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) &= \{ \{a\}^{n} : n \geq 1 \} \end{aligned}$$

is trace equivalence the same as
finite trace equivalence **?**

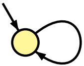
Finite trace relations versus trace relations

is trace equivalence the same as
finite trace equivalence **?**

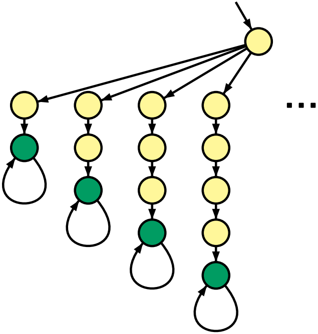
answer: **no**

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$



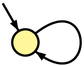
$\bigcirc \ \widehat{=}\ \varnothing \qquad \bullet \ \widehat{=}\ \{b\}$

set of propositions

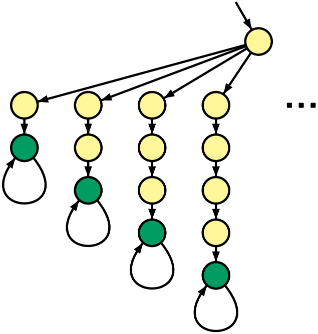
$AP = \{b\}$

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$



$$Traces(\mathcal{T}) \quad = \{\varnothing^\omega\}$$

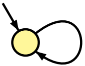
 $\widehat{=} \varnothing$  $\widehat{=} \{b\}$

set of propositions

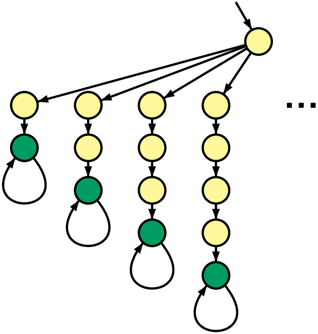
$$AP = \{b\}$$

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$

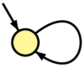


$$Traces(\mathcal{T}) = \{\varnothing^\omega\}$$
$$Traces_{fin}(\mathcal{T}) = \{\varnothing^n : n \geq 0\}$$

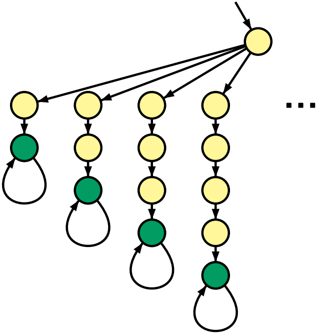
○ $\widehat{=}\ \varnothing$ ● $\widehat{=}\ \{b\}$

set of propositions
$AP = \{b\}$

$\mathcal{T}$



$\mathcal{T}'$



$$
\begin{aligned}
Traces(\mathcal{T}) &= \{\varnothing^\omega\} \\
Traces_{fin}(\mathcal{T}) &= \{\varnothing^n : n \geq 0\} \\
Traces(\mathcal{T}') &= \{\varnothing^n \{b\}^\omega : n \geq 2\}
\end{aligned}
$$

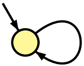
 $\hat{=}\ \varnothing$  $\hat{=}\ \{b\}$

set of propositions

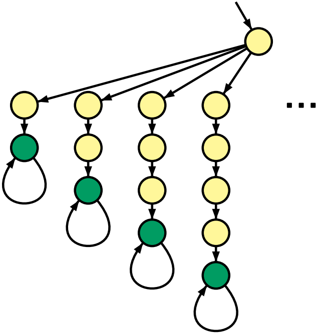
$AP = \{b\}$

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$



$$Traces(\mathcal{T}) = \{\varnothing^\omega\}$$

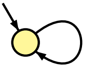
$$Traces_{fin}(\mathcal{T}) = \{\varnothing^n : n \geq 0\}$$

$$Traces(\mathcal{T}') = \{\varnothing^n \{b\}^\omega : n \geq 2\}$$

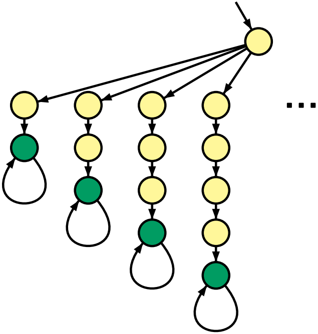
$$Traces_{fin}(\mathcal{T}') = \{\varnothing^n : n \geq 0\} \ \cup \ \{\varnothing^n \{b\}^m : n \geq 2 \wedge m \geq 1\}$$

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$

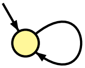


$$
\begin{aligned}
\mathit{Traces}(\mathcal{T}) &= \{\varnothing^\omega\} \\
\mathit{Traces}_{fin}(\mathcal{T}) &= \{\varnothing^n : n \geq 0\} \\
\mathit{Traces}(\mathcal{T}') &= \{\varnothing^n \{b\}^\omega : n \geq 2\} \\
\mathit{Traces}_{fin}(\mathcal{T}') &= \{\varnothing^n : n \geq 0\} \ \cup \\
& \quad \{\varnothing^n \{b\}^m : n \geq 2 \wedge m \geq 1\}
\end{aligned}
$$

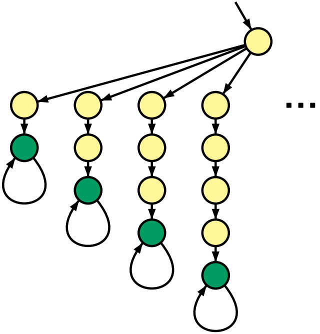
$\mathit{Traces}(\mathcal{T}) \not\subseteq \mathit{Traces}(\mathcal{T}')$, but
$\mathit{Traces}_{fin}(\mathcal{T}) \subseteq \mathit{Traces}_{fin}(\mathcal{T}')$

Finite trace relations versus trace relations

$\mathcal{T}$



$\mathcal{T}'$



$$
\begin{aligned}
Traces(\mathcal{T}) &= \{\varnothing^\omega\} \\
Traces_{fin}(\mathcal{T}) &= \{\varnothing^n : n \geq 0\} \\
Traces(\mathcal{T}') &= \{\varnothing^n \{b\}^\omega : n \geq 2\} \\
Traces_{fin}(\mathcal{T}') &= \{\varnothing^n : n \geq 0\} \ \cup \\
&\quad\ \{\varnothing^n \{b\}^m : n \geq 2 \wedge m \geq 1\}
\end{aligned}
$$

$Traces(\mathcal{T}) \not\subseteq Traces(\mathcal{T}')$, but

$Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

LT property

$E \mathrel{\widehat{=}}$ "eventually b"

$\mathcal{T} \not\models E, \quad \mathcal{T}' \models E$

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over $\boldsymbol{AP}$ such that

(1) $\mathcal{T}$ has no terminal states,

(2) $\mathcal{T}'$ is finite.

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states,
i.e., all paths of $\mathcal{T}$ are infinite

(2) $\mathcal{T}'$ is finite.

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states,
i.e., all paths of $\mathcal{T}$ are infinite

(2) $\mathcal{T}'$ is finite.

Then:

$$\begin{aligned} & \mathit{Traces}(\mathcal{T}) \subseteq \mathit{Traces}(\mathcal{T}') \\ \text{iff}\quad & \mathit{Traces}_{\mathit{fin}}(\mathcal{T}) \subseteq \mathit{Traces}_{\mathit{fin}}(\mathcal{T}') \end{aligned}$$

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states,
i.e., all paths of $\mathcal{T}$ are infinite

(2) $\mathcal{T}'$ is finite.

Then:

$$Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$$
$$\text{iff} \quad Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$$

"$\Longrightarrow$": holds for all transition systems,
no matter whether (1) and (2) hold

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states,
i.e., all paths of $\mathcal{T}$ are infinite

(2) $\mathcal{T}'$ is finite.

Then:
$$Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$$
$$\text{iff} \quad Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$$

"$\Longrightarrow$": holds for all transition systems

"$\Longleftarrow$": suppose that (1) and (2) hold and that

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Show that $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite

(3) $\mathit{Traces}_{fin}(\mathcal{T}) \subseteq \mathit{Traces}_{fin}(\mathcal{T}')$

Then $\mathit{Traces}(\mathcal{T}) \subseteq \mathit{Traces}(\mathcal{T}')$

Proof:

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Then $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

Proof: Pick some path $\pi = s_0\, s_1\, s_2 \ldots$ in $\mathcal{T}$ and show that there exists a path

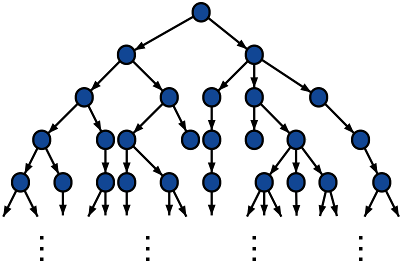
$$\pi' = t_0\, t_1\, t_2 \ldots \text{ in } \mathcal{T}'$$

such that $trace(\pi) = trace(\pi')$

Tracesfin versus traces

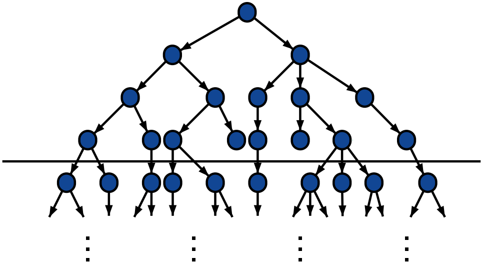
finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)



finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)



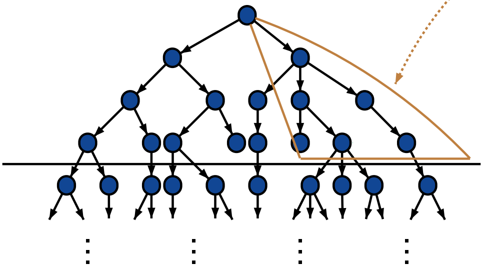
finite until
depth $\leq n$

Tracesfin versus traces

finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)

contains all path fragments with trace $A_0\,A_1 \ldots A_n$



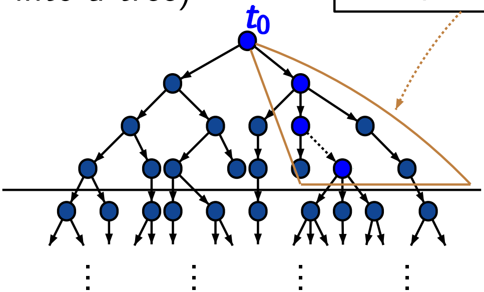
finite until depth $\leq n$

Tracesfin versus traces

finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)

contains all path fragments with trace $A_0\,A_1 \ldots A_n$
in particular: $t_0\,t_1 \ldots t_n$



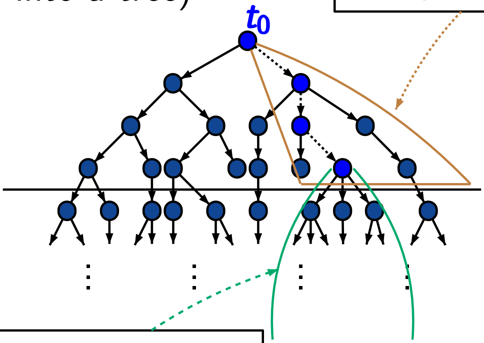
finite until depth $\leq n$

Tracesfin versus traces

finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)

contains all path fragments
with trace $A_0\, A_1 \ldots A_n$
in particular: $t_0\, t_1 \ldots t_n$



finite until
depth $\leq n$

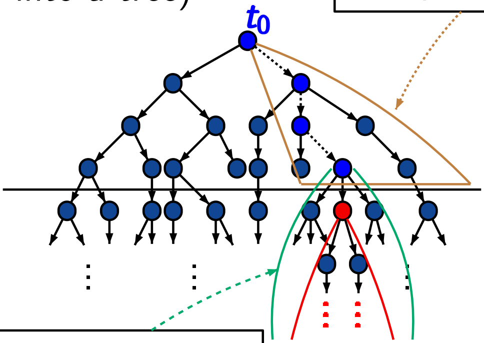
contains infinitely
many path fragments
$t_n\, s^m_{n+1} \ldots s^m_m$

Tracesfin versus traces

finite TS $\mathcal{T}'$

paths from state t_0
(unfolded into a tree)

contains all path fragments
with trace $A_0\, A_1 \ldots A_n$
in particular: $t_0\, t_1 \ldots t_n$



finite until
depth $\leq n$

contains infinitely
many path fragments
$t_n\, s^m_{n+1} \ldots s^m_m$

there exists $t_{n+1} \in Post(t_n)$
s.t. $t_{n+1} = s^m_{n+1}$ for
infinitely many m

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite $\longleftarrow$ image-finiteness is sufficient

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Then $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite $\longleftarrow$ image-finiteness is sufficient

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Then $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

image-finiteness of $\mathcal{T}' = (S', Act, \rightarrow, S'_0, AP, L')$:

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite $\longleftarrow$ image-finiteness is sufficient

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Then $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

image-finiteness of $\mathcal{T}' = (S', Act, \rightarrow, S'_0, AP, L')$:

- for each $A \in 2^{AP}$ and state $s \in S'$:

  $\{t \in Post(s) : L'(t) = A\}$ is finite

Finite trace and trace inclusion

Suppose that $\mathcal{T}$ and $\mathcal{T}'$ are TS over AP such that

(1) $\mathcal{T}$ has no terminal states

(2) $\mathcal{T}'$ is finite ⟵ image-finiteness is sufficient

(3) $Traces_{fin}(\mathcal{T}) \subseteq Traces_{fin}(\mathcal{T}')$

Then $Traces(\mathcal{T}) \subseteq Traces(\mathcal{T}')$

image-finiteness of $\mathcal{T}' = (S', Act, \rightarrow, S'_0, AP, L')$:

- for each $A \in 2^{AP}$ and state $s \in S'$:

  $\{t \in Post(s) : L'(t) = A\}$ is finite

- for each $A \in 2^{AP}$: $\{s_0 \in S'_0 : L'(s_0) = A\}$ is finite

Trace equivalence vs. finite trace equivalence

Whenever $\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ then
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) = \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$

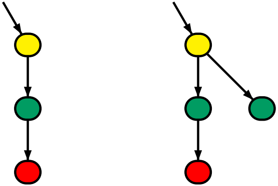
Trace equivalence vs. finite trace equivalence

Whenever $\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ then
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) = \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$

while the reverse direction does not hold in general
(even not for finite transition systems)

Whenever $\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ then
$\mathit{Traces}_{fin}(\mathcal{T}) = \mathit{Traces}_{fin}(\mathcal{T}')$

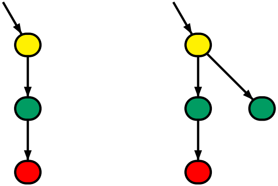
while the reverse direction does not hold in general
(even not for finite transition systems)



Trace equivalence vs. finite trace equivalence

Whenever $\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ then
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) = \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$

while the reverse direction does not hold in general
(even not for finite transition systems)



finite trace equivalent,

but *not* trace equivalent

Whenever $\mathit{Traces}(\mathcal{T}) = \mathit{Traces}(\mathcal{T}')$ then
$\mathit{Traces}_{\mathit{fin}}(\mathcal{T}) = \mathit{Traces}_{\mathit{fin}}(\mathcal{T}')$

The reverse implication holds under additional assumptions, e.g.,

- if $\mathcal{T}$ and $\mathcal{T}'$ are finite and have no terminal states
- or, if $\mathcal{T}$ and $\mathcal{T}'$ are AP-deterministic

Overview

Liveness

“liveness: something good will happen.”

Liveness

"liveness: something good will happen."

"event a will occur eventually"

"liveness: something good will happen."

"event a will occur eventually"

e.g., termination for sequential programs

"liveness: something good will happen."

"event a will occur eventually"

e.g., termination for sequential programs

"event a will occur infinitely many times"

e.g., starvation freedom for dining philosophers

Liveness

"liveness: something good will happen."

"event a will occur eventually"

e.g., termination for sequential programs

"event a will occur infinitely many times"

e.g., starvation freedom for dining philosophers

"whenever event b occurs then event a will occur sometimes in the future"

Liveness

"liveness: something good will happen."

"event ***a*** will occur eventually"

e.g., termination for sequential programs

"event ***a*** will occur infinitely many times"

e.g., starvation freedom for dining philosophers

"whenever event ***b*** occurs then event ***a***
will occur sometimes in the future"

e.g., every waiting process enters eventually
its critical section

which property type?

- Each philosopher thinks infinitely often.

which property type?

- Each philosopher thinks infinitely often.

liveness

which property type? LF2.6-2

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

  safety

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

  safety

- Whenever a philosopher eats then he will think some time afterwards.

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

  safety

- Whenever a philosopher eats then he will think some time afterwards.

  liveness

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

  safety

- Whenever a philosopher eats then he will think some time afterwards.

  liveness

- Between two eating phases of philosopher $\boldsymbol{i}$ lies at least one eating phase of philosopher $\boldsymbol{i{+}1}$.

which property type?

- Each philosopher thinks infinitely often.

  liveness

- Two philosophers next to each other never eat at the same time.

  invariant

- Whenever a philosopher eats then he has been thinking at some time before.

  safety

- Whenever a philosopher eats then he will think some time afterwards.

  liveness

- Between two eating phases of philosopher $\boldsymbol{i}$ lies at least one eating phase of philosopher $\boldsymbol{i+1}$.

  safety

Liveness

many different formal definitions of liveness
have been suggested in the literature

many different formal definitions of liveness
have been suggested in the literature

here: one just example for a formal definition
of liveness

Definition of liveness properties

Definition of liveness properties

Let E be an LT property over AP, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$.

E is called a liveness property if each finite word over AP can be extended to an infinite word in E

Definition of liveness properties

Let E be an LT property over AP, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$.

E is called a liveness property if each finite word over AP can be extended to an infinite word in E, i.e., if

$$pref(E) \ = \ \left(2^{AP}\right)^{+}$$

recall: $pref(E) =$ set of all finite, nonempty prefixes of words in E

Definition of liveness properties

Let E be an LT property over AP, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$.

> E is called a liveness property if each finite word over AP can be extended to an infinite word in E, i.e., if
>
> $$pref(E) \ = \ \left(2^{AP}\right)^{+}$$

Examples:

- each process will eventually enter its critical section
- each process will enter its critical section infinitely often
- whenever a process has requested its critical section then it will eventually enter its critical section

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = (2^{AP})^+$

Examples for $AP = \{crit_i : i = 1, \ldots, n\}$:

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) \ = \ \left(2^{AP}\right)^+$

Examples for $AP = \{crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = \left(2^{AP}\right)^+$

Examples for $AP = \{crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section

E = set of all infinite words $A_0\, A_1\, A_2 \ldots$ s.t.

$$\forall i \in \{1, \ldots, n\}\ \exists k \geq 0.\ crit_i \in A_k$$

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = (2^{AP})^+$

Examples for $AP = \{crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section
- each process will enter its critical section infinitely often

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = (2^{AP})^+$

Examples for $AP = \{crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section
- each process will enter its critical section infinitely often

E = set of all infinite words $A_0 A_1 A_2 \ldots$ s.t.

$$\forall i \in \{1, \ldots, n\} \ \stackrel{\infty}{\exists} k \geq 0.\ crit_i \in A_k$$

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = (2^{AP})^+$

Examples for $AP = \{wait_i, crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section
- each process will enter its crit. section inf. often
- whenever a process is waiting then it will eventually enter its critical section

Examples for liveness properties

An LT property E over AP is called a liveness property if $pref(E) = (2^{AP})^+$

Examples for $AP = \{wait_i, crit_i : i = 1, \ldots, n\}$:

- each process will eventually enter its critical section
- each process will enter its crit. section inf. often
- whenever a process is waiting then it will eventually enter its critical section

E = set of all infinite words $A_0 A_1 A_2 \ldots$ s.t.

$$\forall i \in \{1, \ldots, n\}\ \forall j \geq 0.\ wait_i \in A_j \longrightarrow \exists k > j.\ crit_i \in A_k$$

Recall: safety properties, prefix closure

Let E be an LT-property, i.e., $E \subseteq \left(2^{AP}\right)^\omega$

Recall: safety properties, prefix closure

Let E be an LT-property, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$

> E is a safety property
>
> iff $\forall \sigma \in \left(2^{AP}\right)^{\omega} \setminus E \;\; \exists A_0 A_1 \ldots A_n \in \mathit{pref}(\sigma)$ s.t.
>
> $$\left\{\sigma' \in E : A_0 A_1 \ldots A_n \in \mathit{pref}(\sigma')\right\} = \varnothing$$

Recall: safety properties, prefix closure

Let E be an LT-property, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$

E is a safety property

iff $\forall \sigma \in \left(2^{AP}\right)^{\omega} \setminus E \;\; \exists A_0 A_1 \ldots A_n \in \mathit{pref}(\sigma)$ s.t.

$$\left\{\sigma' \in E : A_0 A_1 \ldots A_n \in \mathit{pref}(\sigma')\right\} = \varnothing$$

remind:

$$\mathit{pref}(\sigma) = \text{set of all finite, nonempty prefixes of } \sigma$$

$$\mathit{pref}(E) = \bigcup_{\sigma \in E} \mathit{pref}(\sigma)$$

Recall: safety properties, prefix closure

Let E be an LT-property, i.e., $E \subseteq \left(2^{AP}\right)^{\omega}$

E is a safety property

iff $\forall \sigma \in \left(2^{AP}\right)^{\omega} \setminus E \;\; \exists A_0 A_1 \dots A_n \in \mathit{pref}(\sigma)$ s.t.

$$\left\{\sigma' \in E : A_0 A_1 \dots A_n \in \mathit{pref}(\sigma')\right\} = \varnothing$$

iff $cl(E) = E$

remind: $cl(E) = \left\{\sigma \in \left(2^{AP}\right)^{\omega} : \mathit{pref}(\sigma) \subseteq \mathit{pref}(E)\right\}$

$\mathit{pref}(\sigma)$ = set of all finite, nonempty prefixes of σ

$$\mathit{pref}(E) = \bigcup_{\sigma \in E} \mathit{pref}(\sigma)$$

Decomposition theorem

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E \;=\; SAFE \cap LIVE$$

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E = SAFE \cap LIVE$$

Proof:

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E = SAFE \cap LIVE$$

Proof: Let $SAFE \stackrel{\text{def}}{=} cl(E)$

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E = SAFE \cap LIVE$$

Proof: Let $SAFE \stackrel{\text{def}}{=} cl(E)$

---

remind: $cl(E) = \left\{ \sigma \in \left(2^{AP}\right)^{\omega} : pref(\sigma) \subseteq pref(E) \right\}$

$pref(\sigma)$ = set of all finite, nonempty prefixes of σ

$$pref(E) = \bigcup_{\sigma \in E} pref(\sigma)$$

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E = SAFE \cap LIVE$$

Proof: Let

$$SAFE \stackrel{\text{def}}{=} cl(E)$$

$$LIVE \stackrel{\text{def}}{=} E \cup \left(\left(2^{AP}\right)^{\omega} \setminus cl(E) \right)$$

---

remind: $cl(E) = \left\{ \sigma \in \left(2^{AP}\right)^{\omega} : pref(\sigma) \subseteq pref(E) \right\}$

$pref(\sigma)$ = set of all finite, nonempty prefixes of σ

$$pref(E) = \bigcup_{\sigma \in E} pref(\sigma)$$

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E \;=\; SAFE \cap LIVE$$

Proof: Let $SAFE \stackrel{\text{def}}{=} cl(E)$

$$LIVE \stackrel{\text{def}}{=} E \cup \left(\left(2^{AP}\right)^{\omega} \setminus cl(E) \right)$$

Show that:

- $E = SAFE \cap LIVE$
- $SAFE$ is a safety property
- $LIVE$ is a liveness property

Decomposition theorem

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E = SAFE \cap LIVE$$

Proof: Let $SAFE \stackrel{\text{def}}{=} cl(E)$

$$LIVE \stackrel{\text{def}}{=} E \cup \big((2^{AP})^{\omega} \setminus cl(E) \big)$$

Show that:

- $E = SAFE \cap LIVE$ $\quad \surd$
- $SAFE$ is a safety property
- $LIVE$ is a liveness property

Decomposition theorem

For each LT-property E, there exists a safety property *SAFE* and a liveness property *LIVE* s.t.

$$E \;=\; \mathit{SAFE} \cap \mathit{LIVE}$$

Proof: Let $\mathit{SAFE} \stackrel{\text{def}}{=} cl(E)$

$$\mathit{LIVE} \stackrel{\text{def}}{=} E \cup \big((2^{AP})^{\omega} \setminus cl(E) \big)$$

Show that:

- $E = \mathit{SAFE} \cap \mathit{LIVE}$ $\quad\surd$
- *SAFE* is a safety property as $cl(\mathit{SAFE}) = \mathit{SAFE}$
- *LIVE* is a liveness property

For each LT-property E, there exists a safety property $SAFE$ and a liveness property $LIVE$ s.t.

$$E \;=\; SAFE \cap LIVE$$

Proof: Let $SAFE \stackrel{\text{def}}{=} cl(E)$

$$LIVE \stackrel{\text{def}}{=} E \cup \left(\left(2^{AP}\right)^{\omega} \setminus cl(E) \right)$$

Show that:

- $E = SAFE \cap LIVE$ $\surd$
- $SAFE$ is a safety property as $cl(SAFE) = SAFE$
- $LIVE$ is a liveness property, i.e., $pref(LIVE) = \left(2^{AP}\right)^{+}$

Being safe and live

Which LT properties are both
a safety and a liveness property?

Being safe and live

Which LT properties are both
a safety and a liveness property?

answer: The set $(2^{AP})^{\omega}$ is the only LT property which is a safety property and a liveness property

Which LT properties are both a safety and a liveness property?

answer: The set $(2^{AP})^\omega$ is the only LT property which is a safety property and a liveness property

- $(2^{AP})^\omega$ is a safety and a liveness property: $\surd$

Being safe and live

Which LT properties are both
a safety and a liveness property?

answer: The set $(2^{AP})^{\omega}$ is the only LT property which is a safety property and a liveness property

- $(2^{AP})^{\omega}$ is a safety and a liveness property: $\surd$
- If E is a liveness property then

$$pref(E) = (2^{AP})^{+}$$

Which LT properties are both a safety and a liveness property?

answer: The set $(2^{AP})^\omega$ is the only LT property which is a safety property and a liveness property

- $(2^{AP})^\omega$ is a safety and a liveness property: $\surd$
- If E is a liveness property then

$$\begin{aligned} & \mathit{pref}(E) = (2^{AP})^+ \\ \Longrightarrow \quad & cl(E) = (2^{AP})^\omega \end{aligned}$$

Being safe and live

> Which LT properties are both
> a safety and a liveness property?

answer: The set $(2^{AP})^\omega$ is the only LT property which is a safety property and a liveness property

- $(2^{AP})^\omega$ is a safety and a liveness property: $\surd$
- If E is a liveness property then

$$pref(E) = (2^{AP})^+$$
$$\Longrightarrow \quad cl(E) = (2^{AP})^\omega$$

If E is a safety property too, then $cl(E) = E$.

Being safe and live

> Which LT properties are both a safety and a liveness property?

answer: The set $(2^{AP})^\omega$ is the only LT property which is a safety property and a liveness property

- $(2^{AP})^\omega$ is a safety and a liveness property: $\surd$
- If E is a liveness property then

$$pref(E) = (2^{AP})^+$$
$$\Longrightarrow \quad cl(E) = (2^{AP})^\omega$$

If E is a safety property too, then $cl(E) = E$.
Hence $E = cl(E) = (2^{AP})^\omega$.