Outline

Static and dynamic properties

Example: Static properties

Mutual exclusion

Safety properties are often static.

They can be reduced to reachability.

Example: Dynamic properties

Every elevator request should be eventually granted.

The elevator should not cross a level for which a call is pending without stopping.

Temporal Structures

Definition: Flows of time

A *flow of time* is a strict order $(\mathbb{T}, <)$ where $\mathbb{T}$ is the nonempty set of *time points* and $<$ is an irreflexive transitive relation on $\mathbb{T}$.

Example: Flows of time

- $(\{0, \ldots, n\}, <)$: Finite runs of sequential systems.
- $(\mathbb{N}, <)$: Infinite runs of sequential systems.
- $(\mathbb{R}, <)$: runs of real-time sequential systems.
- Trees: Finite or infinite run-trees of sequential systems.
- Mazurkiewicz traces: runs of distributed systems (rendez-vous).
- Message sequence charts: runs of distributed systems (FIFO).
- and also $(\mathbb{Z}, <)$ or $(\mathbb{Q}, <)$ or $(\omega^2, <)$, ...

Definition: Temporal Structures

Let AP be a set of atoms (atomic propositions) and let $\mathcal{C}$ be a class of time flows. A *temporal structure* over $(\mathcal{C}, \mathrm{AP})$ is a triple $(\mathbb{T}, <, \lambda)$ where $(\mathbb{T}, <)$ is a time flow in $\mathcal{C}$ and $\lambda \colon \mathbb{T} \to 2^{\mathrm{AP}}$ labels time points with atomic propositions.
The temporal structure $(\mathbb{T}, <, \lambda)$ is also denoted $(\mathbb{T}, <, h)$ where $h : \mathrm{AP} \to 2^{\mathbb{T}}$ assigns time points to atomic propositions: $h(p) = \{t \in \mathbb{T} \mid p \in \lambda(t)\}$ for $p \in \mathrm{AP}$.

Linear behaviors and specifications

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure (we omit actions: $T \subseteq S \times S$).

Definition: Runs as temporal structures

An infinite run $\sigma = s_0 s_1 s_2 \cdots$ of M with $(s_i, s_{i+1}) \in T$ for all $i \geq 0$ defines a *linear* temporal structure $\ell(\sigma) = (\mathbb{N}, <, \lambda)$ where $\lambda(i) = \ell(s_i)$ for $i \in \mathbb{N}$.

Such a temporal structure can be seen as an infinite word over $\Sigma = 2^{\mathrm{AP}}$:
$\ell(\sigma) = \ell(s_0)\ell(s_1)\ell(s_2) \cdots \in \Sigma^\omega$

Linear specifications only depend on runs.

Example: The printer manager is starvation free.

On each run, whenever some process requests the printer, it eventually gets it.

Remark:

Two Kripke structures having the same linear temporal structures satisfy the same linear specifications.

Branching behaviors and specifications

The system has an infinite active run, along which it may always reach an inactive state.

Definition: Computation-tree or run-tree : unfolding of the TS

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure. Wlog. $I = \{s_0\}$ is a singleton.

Let D be a finite set with $|D|$ the outdegree of the transition relation T.

The computation-tree of M is an *unordered* tree $t : D^* \to S$ (partial map) s.t.

- $t(\varepsilon) = s_0$,
- For every node $u \in \mathrm{dom}(t)$ labelled $s = t(u)$, if $T(s) = \{s_1, \ldots, s_k\}$ then u has exactly k children which are labelled $s_1, \ldots, s_k$

Associated temporal structure $\ell(t) = (\mathrm{dom}(t), <, \lambda)$ where

- $<$ is the strict prefix relation over D^*,
- and $\lambda(u) = \ell(t(u))$ for $u \in \mathrm{dom}(t)$.

(Linear) runs of M are branches of the computation-tree t.

First-order Specifications

Definition: Syntax of $\mathrm{FO}(\mathrm{AP}, <)$

Let $\mathrm{Var} = \{x, y, \ldots\}$ be first-order variables.

$$\varphi ::= \bot \mid p(x) \mid x = y \mid x < y \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x\, \varphi$$

where $p \in \mathrm{AP}$.

Definition: Semantics of $\mathrm{FO}(\mathrm{AP}, <)$

Let $w = (\mathbb{T}, <, \lambda)$ be a temporal structure over AP.
Let $\nu : \mathrm{Var} \to \mathbb{T}$ be an assignment of first-order variables to time points.

$$\begin{array}{lll} w, \nu \models p(x) & \text{if} & p \in \lambda(\nu(x)) \\ w, \nu \models x = y & \text{if} & \nu(x) = \nu(y) \\ w, \nu \models x < y & \text{if} & \nu(x) < \nu(y) \\ w, \nu \models \exists x\, \varphi & \text{if} & w, \nu[x \mapsto t] \models \varphi \text{ for some } t \in \mathbb{T} \end{array}$$

where $\nu[x \mapsto t]$ maps x to t and $y \neq x$ to $\nu(y)$.

Previous specifications can be written in $\mathrm{FO}(<)$ (except the branching one).

First-order vs Temporal

First-order logic

- $\mathrm{FO}(<)$ has a good expressive power
  . . . but $\mathrm{FO}(<)$-formulae are not easy to write and to understand.
- $\mathrm{FO}(<)$ is decidable
  . . . but satisfiability and model checking are non elementary.

Temporal logics

- no variables: time is implicit.
- quantifications and variables are replaced by modalities.
- Usual specifications are easy to write and read.
- Good complexity for satisfiability and model checking problems.
- Good expressive power.

Linear Temporal Logic (LTL) over $(\mathbb{N}, <, \mathrm{AP})$ introduced by Pnueli (1977) as a convenient specification language for verification of systems.

Temporal Specifications

Definition: Syntax of $\mathrm{TL}(\mathrm{AP}, \mathsf{SU}, \mathsf{SS})$

$$\varphi ::= \bot \mid p \ (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \varphi \,\mathsf{SU}\, \varphi \mid \varphi \,\mathsf{SS}\, \varphi$$

Definition: Semantics: $w = (\mathbb{T}, <, \lambda)$ temporal structure and $i \in \mathbb{T}$

$$\begin{array}{lll}
w, i \models p & \text{if} & p \in \lambda(i) \\
w, i \models \neg\varphi & \text{if} & w, i \not\models \varphi \\
w, i \models \varphi \vee \psi & \text{if} & w, i \models \varphi \text{ or } w, i \models \psi \\
\\
w, i \models \varphi \,\mathsf{SU}\, \psi & \text{if} & \exists k \ i < k \text{ and } w, k \models \psi \text{ and } \forall j \ (i < j < k \rightarrow w, j \models \varphi) \\
\\
w, i \models \varphi \,\mathsf{SS}\, \psi & \text{if} & \exists k \ i > k \text{ and } w, k \models \psi \text{ and } \forall j \ (i > j > k \rightarrow w, j \models \varphi)
\end{array}$$

Previous specifications can be written in $\mathrm{TL}(\mathrm{AP}, \mathsf{SU}, \mathsf{SS})$
(except the branching one).

Theorem: $\mathrm{TL} \subseteq \mathrm{FO}^3$

For each $\varphi \in \mathrm{TL}(\mathrm{AP}, \mathsf{SU}, \mathsf{SS})$ we can construct an equivalent formula with one free variable $\widetilde{\varphi}(x) \in \mathrm{FO}^3(\mathrm{AP}, <)$.

Temporal Specifications

Definition: non-strict versions of until and since

$$\varphi \,\mathsf{U}\, \psi \stackrel{\text{def}}{=} \psi \vee (\varphi \wedge \varphi \,\mathsf{SU}\, \psi) \qquad \varphi \,\mathsf{S}\, \psi \stackrel{\text{def}}{=} \psi \vee (\varphi \wedge \varphi \,\mathsf{SS}\, \psi)$$

$$w, i \models \varphi \,\mathsf{U}\, \psi \quad \text{if} \quad \exists k\ i \leq k \text{ and } w, k \models \psi \text{ and } \forall j\ (i \leq j < k \rightarrow w, j \models \varphi)$$
$$w, i \models \varphi \,\mathsf{S}\, \psi \quad \text{if} \quad \exists k\ i \geq k \text{ and } w, k \models \psi \text{ and } \forall j\ (i \geq j > k \rightarrow w, j \models \varphi)$$

Definition: Derived modalities

$$\mathsf{X}\,\varphi \stackrel{\text{def}}{=} \bot \,\mathsf{SU}\, \varphi \quad \text{Next} \qquad \mathsf{Y}\,\varphi \stackrel{\text{def}}{=} \bot \,\mathsf{SS}\, \varphi \quad \text{Yesterday}$$

$$w, i \models \mathsf{X}\,\varphi \quad \text{if} \quad \exists k\ i < k \text{ and } w, k \models \varphi \text{ and } \neg\exists j\ (i < j < k)$$
$$w, i \models \mathsf{Y}\,\varphi \quad \text{if} \quad \exists k\ i > k \text{ and } w, k \models \varphi \text{ and } \neg\exists j\ (i > j > k)$$

$$\begin{aligned} \mathsf{SF}\,\varphi &\stackrel{\text{def}}{=} \top \,\mathsf{SU}\, \varphi \\ \mathsf{F}\,\varphi &\stackrel{\text{def}}{=} \top \,\mathsf{U}\, \varphi \\ \mathsf{G}\,\varphi &\stackrel{\text{def}}{=} \neg\,\mathsf{F}\,\neg\varphi \end{aligned} \qquad \begin{aligned} \mathsf{SP}\,\varphi &\stackrel{\text{def}}{=} \top \,\mathsf{SS}\, \varphi \\ \mathsf{P}\,\varphi &\stackrel{\text{def}}{=} \top \,\mathsf{S}\, \varphi \\ \mathsf{H}\,\varphi &\stackrel{\text{def}}{=} \neg\,\mathsf{P}\,\neg\varphi \end{aligned}$$

$$\begin{aligned} \varphi \,\mathsf{W}\, \psi &\stackrel{\text{def}}{=} (\mathsf{G}\,\varphi) \vee (\varphi \,\mathsf{U}\, \psi) && \text{Weak Until} \\ \varphi \,\mathsf{R}\, \psi &\stackrel{\text{def}}{=} (\mathsf{G}\,\psi) \vee (\psi \,\mathsf{U}\, (\varphi \wedge \psi)) && \text{Release} \end{aligned}$$

Temporal Specifications

Example: Specifications on the time flow $(\mathbb{N}, <)$

- Safety: $\mathsf{G}\,\mathrm{good}$
- MutEx: $\neg\,\mathsf{F}(\mathrm{crit}_1 \wedge \mathrm{crit}_2)$
- Liveness: $\mathsf{G}\,\mathsf{F}\,\mathrm{active}$
- Response: $\mathsf{G}(\mathrm{request} \rightarrow \mathsf{F}\,\mathrm{grant})$
- Response': $\mathsf{G}(\mathrm{request} \rightarrow (\neg\mathrm{request}\ \mathsf{SU}\ \mathrm{grant}))$
- Release: $\mathrm{reset}\ \mathsf{R}\ \mathrm{alarm}$
- Strong fairness: $(\mathsf{G}\,\mathsf{F}\,\mathrm{request}) \rightarrow (\mathsf{G}\,\mathsf{F}\,\mathrm{grant})$
- Weak fairness: $(\mathsf{F}\,\mathsf{G}\,\mathrm{request}) \rightarrow (\mathsf{G}\,\mathsf{F}\,\mathrm{grant})$
- Stability: $\mathsf{G}\,\neg p \vee (\neg p\ \mathsf{U}\ \mathsf{G}\,p)$

Discrete linear time flows

Definition: discrete linear time flows $(\mathbb{T}, <)$

A linear time flow is discrete if $\mathsf{SF}\,\top \to \mathsf{X}\,\top$ and $\mathsf{SP}\,\top \to \mathsf{Y}\,\top$ are valid formulae.

$(\mathbb{N}, <)$ and $(\mathbb{Z}, <)$ are discrete.

$(\mathbb{Q}, <)$ and $(\mathbb{R}, <)$ are not discrete.

Exercise: For discrete linear time flows $(\mathbb{T}, <)$

$$\varphi\,\mathsf{SU}\,\psi \equiv \mathsf{X}(\varphi\,\mathsf{U}\,\psi) \qquad \neg\,\mathsf{X}\,\varphi \equiv \neg\,\mathsf{X}\,\top \vee \mathsf{X}\,\neg\varphi$$

$$\varphi\,\mathsf{SS}\,\psi \equiv \mathsf{Y}(\varphi\,\mathsf{S}\,\psi) \qquad \neg\,\mathsf{Y}\,\varphi \equiv \neg\,\mathsf{Y}\,\top \vee \mathsf{Y}\,\neg\varphi$$

$$\begin{aligned} \neg(\varphi\,\mathsf{U}\,\psi) &\equiv (\mathsf{G}\,\neg\psi) \vee (\neg\psi\,\mathsf{U}\,(\neg\varphi \wedge \neg\psi)) \\ &\equiv \neg\psi\,\mathsf{W}\,(\neg\varphi \wedge \neg\psi) \\ &\equiv \neg\varphi\,\mathsf{R}\,\neg\psi \end{aligned}$$

Remark: Dense time flow $\mathbb{T} = \mathbb{Q}$ or $\mathbb{T} = \mathbb{R}$

$\neg(\varphi\,\mathsf{U}\,\psi)$ does not imply $\neg\varphi\,\mathsf{R}\,\neg\psi$.

For instance, $w = (\mathbb{T}, <, \ell)$ with $\mathbb{T} = \{0\} \cup \{\frac{1}{n} \mid n \in \mathbb{N}\}$ with $\ell(0) = \{p\}$, $\ell(\frac{1}{2n}) = \{p\}$ and $\ell(\frac{1}{2n+1}) = \{q\}$. Then, $w, 0 \models \neg(p\,\mathsf{U}\,q)$ and $w, 0 \not\models \neg p\,\mathsf{R}\,\neg q$.

Model checking for linear behaviors

Definition: Model checking problem

Input: A Kripke structure $M = (S, T, I, \mathrm{AP}, \ell)$
A formula $\varphi \in \mathrm{LTL}(\mathrm{AP}, \mathsf{SU}, \mathsf{SS})$

Question: Does $M \models \varphi$?

- Universal MC: $M \models_\forall \varphi$ if $\ell(\sigma), 0 \models \varphi$ for all initial infinite runs σ of M.
- Existential MC: $M \models_\exists \varphi$ if $\ell(\sigma), 0 \models \varphi$ for some initial infinite run σ of M.

$$M \models_\forall \varphi \quad \text{iff} \quad M \not\models_\exists \neg\varphi$$

Theorem [11, Sistla, Clarke 85], [10, Lichtenstein & Pnueli 85]

The Model checking problem for LTL is PSPACE-complete. Proof later