# Hoare logic and Model checking

**Part II: Model checking**

**Lecture 9: Temporal logic (continued)**

**Christopher Pulte** cp526
University of Cambridge

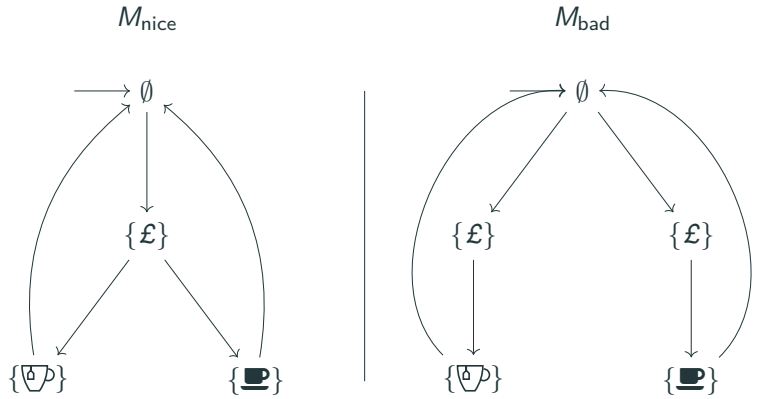
CST Part II – 2023/24

In the last lecture we saw LTL and how we can use it to specify safety and liveness properties of temporal models. LTL can express properties of linear paths in a temporal model, but cannot express properties of the different possible paths out of a state/the branching structure in the temporal model.

In this lecture we will define CTL, which allows expressing such properties, and CTL* that combines the expressive power of LTL and CTL.



## Tea & coffee machines




A good property about $M_{\text{nice}}$: "Following payment, it is **possible** to receive coffee in the next state". We cannot say this in LTL.



## CTL: computation tree logic

Formally stating this property requires the ability to specify properties of the **tree** of possible paths out of a state of the model, not just an individual linear path.

In CTL time is "tree-shaped": it considers for each state/point in time the set of possible futures. Properties of temporal models are specified in formulas that can quantify over possible futures.

## Syntax of CTL

Given a fixed set of atomic propositions $AP$,

| $\psi \in \text{StateProp} ::=$ | | | $\phi \in \text{PathProp} ::=$ | | |
|---|---|---|---|---|---|
| $\bot$ | \| | false | X $\psi$ | \| | neXt |
| $\top$ | \| | true | F $\psi$ | \| | future |
| $\neg\psi$ | \| | negation | G $\psi$ | \| | generally |
| $\psi_1 \wedge \psi_2$ | \| | conjunction | $\psi_1$ U $\psi_2$ | \| | until |
| $\psi_1 \vee \psi_2$ | \| | disjunction | | | |
| $\psi_1 \to \psi_2$ | \| | implication | | | |
| injp $p$ | \| | atomic proposition | | | |
| A $\phi$ | \| | universal | | | |
| E $\phi$ | \| | existential | | | |

We usually omit injp.



## Informal semantics of CTL

- A CTL formula is a state property.
- A temporal model satisfies a CTL formula, if all its initial states satisfy the formula.



## Informal semantics of CTL

State properties:

- $\bot$ holds for no state
- $\top$ holds for any state
- $\neg\psi$: $\psi$ does not hold
- $\psi_1 \wedge \psi_2$: $\psi_1$ holds and $\psi_2$ holds
- $\psi_1 \vee \psi_2$: $\psi_1$ holds or $\psi_2$ holds
- $\psi_1 \to \psi_2$: $\psi_1$ does not hold or $\psi_2$ holds
- injp $p$: the current state satisfies atomic proposition $p$
- A $\phi$: every outgoing path satisfies **path property** $\phi$
- E $\phi$: some outgoing path satisfies **path property** $\phi$



## Informal semantics of CTL. Changed wrt handout.

Path properties:

- X $\psi$: the next state along the current path satisfies **state property** $\psi$
- F $\psi$: some state along the current path satisfies **state property** $\psi$
- G $\psi$: every state along the current path satisfies **state property** $\psi$
- $\psi_1$ U $\psi_2$: some state along the current path satisfies **state property** $\psi_2$, and all states along the current path before that satisfy **state property** $\psi_1$

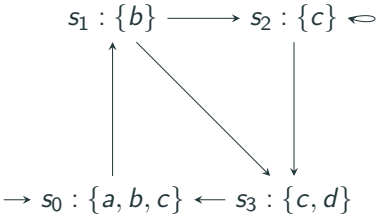
## Notation

Note: the literature sometimes uses alternative notation for the temporal operators:

- $\forall$ instead of A
- $\exists$ instead of E
- $\bigcirc$ instead of X
- $\diamond$ instead of F
- $\Box$ instead of G



## Semantics of CTL – Examples

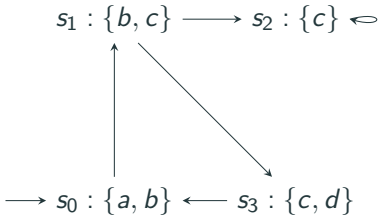



Consider the state $s_1$. Then:

- $s_1$ satisfies E X $d$
- $s_1$ satisfies A X $c$
- $s_1$ does not satisfy A F $a$
- $s_1$ does not satisfy E G $c$
- $s_1$ satisfies A G E F $a$



## Semantics of CTL – Examples




Consider the state $s_1$. Then:

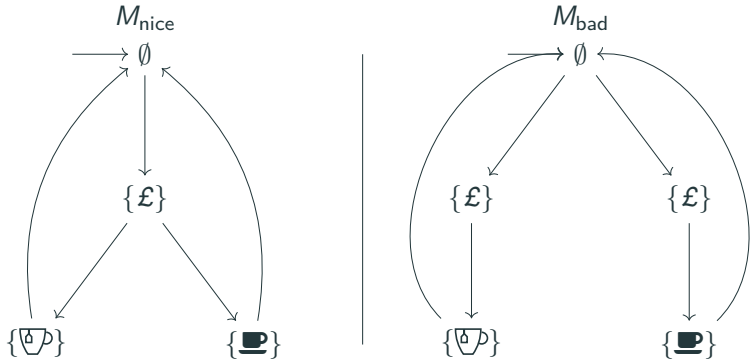
- $s_1$ satisfies E F A G $c$
- $s_1$ satisfies A F $b$
- $s_1$ does not satisfy A ($b$ U $d$)
- $s_1$ satisfies A G A F $c$
- $s_1$ satisfies A ($\bot$ U (E X $d$))



## Semantics of CTL

What does it mean for a temporal model to satisfy a CTL formula? Instead of defining CTL now, we later define the semantics of CTL* formally.

## Tea & coffee machines




A G ($£ \to$ E X ☕): "For all paths, at every state (along the path) in which money has been paid, there exists a path from this state in which, in the next state, the machine has produced coffee."

LTL cannot distinguish between the good and the bad coffee machine.



## CTL Examples: Elevator

- "If it is possible to answer a call to level 2 in the next step, then the elevator does that."

$$\mathsf{A\ G\ ((Call_2 \wedge E\ X\ Loc_2) \to A\ X\ Loc_2)}$$

- "From every state the elevator can reach the second floor."

$$\mathsf{A\ G\ E\ F\ Loc_2}$$

(This does not mean the elevator will necessarily go there.)



## LTL vs. CTL

The examples of the coffee and tea machine and the elevator show some properties CTL can express that LTL cannot: properties related to the set of possible paths out of some state.

Is CTL strictly more expressive than LTL?



## LTL vs. CTL

No: the LTL formula (F $p$) $\to$ (F $q$) has no CTL equivalent. E.g. these are not equivalent to the LTL formula:

- (A F $p$) $\to$ (A F $q$)
- A G ($p \to$ A F $q$)

CTL formulas are state properties; embedding path properties always requires path quantification using A or E .

(See Huth and Ryan. "Logic in Computer Science – Modelling and Reasoning about Systems", Chapter 3.5)

## CTL*

CTL* combines the expressive power of LTL and CTL, by dropping from CTL the requirement that path properties have to be associated with path quantification, which allows us to express properties such as the following, that have no LTL or CTL equivalent:

E G F $p$: "there exists a path such that $p$ holds infinitely often"



## Syntax of CTL*. Updated wrt handout.

Given a fixed set of atomic propositions $AP$,

| $\psi, \ldots \in$ StateProp ::= | | | $\phi, \ldots \in$ PathProp ::= | | |
|---|---|---|---|---|---|
| $\bot$ | \| | false | $\phi_1 \wedge^{\mathsf{p}} \phi_2$ | \| | conjunction |
| $\top$ | \| | true | $\phi_1 \vee^{\mathsf{p}} \phi_2$ | \| | disjunction |
| $\psi_1 \wedge^{\mathsf{s}} \psi_2$ | \| | conjunction | $\phi_1 \rightarrow^{\mathsf{p}} \phi_2$ | \| | implication |
| $\psi_1 \vee^{\mathsf{s}} \psi_2$ | \| | disjunction | injs $\psi$ | \| | state property |
| $\psi_1 \rightarrow^{\mathsf{s}} \psi_2$ | \| | implication | X $\phi$ | \| | next |
| injp $p$ | \| | atomic proposition | F $\phi$ | \| | future |
| A $\phi$ | \| | universal | G $\phi$ | \| | generally |
| E $\phi$ | \| | existential | $\phi_1$ U $\phi_2$ | \| | until |

We almost always omit injp and injs. We encode negation:

- $\neg^{\mathsf{s}} \psi \stackrel{def}{=} \psi \rightarrow^{\mathsf{s}} \bot$
- $\neg^{\mathsf{p}} \phi \stackrel{def}{=} \phi \rightarrow^{\mathsf{p}} (\text{injs } \bot)$



## Informal semantics of CTL*

- injp $p$: the current state satisfies atomic proposition $p$
- A $\phi$: all paths starting from the current state satisfy $\phi$
- E $\phi$: some path starting from the current state satisfies $\phi$
- injs $\psi$: the first state of the current path satisfies $\psi$
- G $\phi$: every suffix of the current path satisfies $\phi$
- F $\phi$: some suffix of the current path satisfies $\phi$
- X $\phi$: the tail of the current path satisfies $\phi$
- $\phi_1$ U $\phi_2$: some suffix of the current path satisfies $\phi_2$, and all the suffixes of the current path of which that path is a suffix satisfy $\phi_1$



## Example propositions in CTL*: path quantification

- E G F $p$: "there exists a path on which $p$ holds infinitely often"
- E G A F $p$: "there exists a path such that all paths starting from any state along that path eventually reach a state in which $p$ holds"
- E G E F $p$: "there exists a path such that from every state along this path a state satisfying $p$ can be reached"

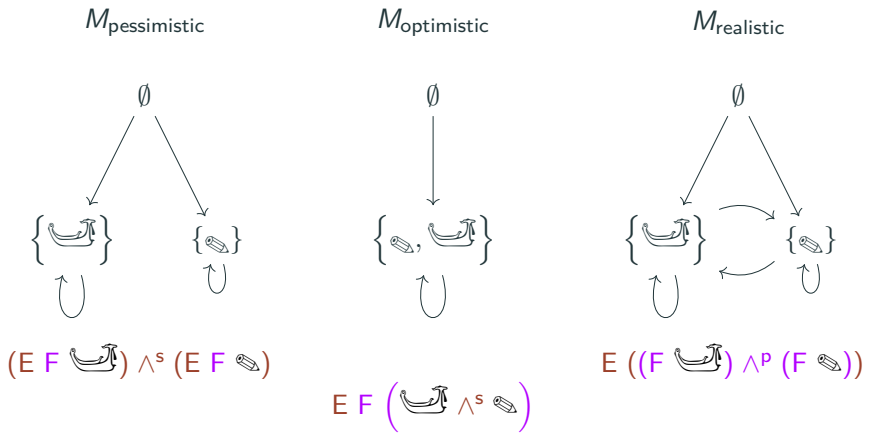
## Example propositions in CTL*: conjunctions

- $\mathsf{E}\ \mathsf{F}\ (p \wedge^{\mathsf{s}} q)$: there is a state reachable from the current state that satisfies both $p$ and $q$
- $\mathsf{E}\ ((\mathsf{F}\ p) \wedge^{\mathsf{p}} (\mathsf{F}\ q))$: there is a path from the current state, along which there is a state satisfying $p$ and a state satisfying $q$
- $(\mathsf{E}\ \mathsf{F}\ p) \wedge^{\mathsf{s}} (\mathsf{E}\ \mathsf{F}\ q)$: there is a state reachable from the current state that satisfies $p$ and a reachable state that satisfies $q$



## Example of path conjunction vs. state conjunction

"At Cambridge, you can row and study"






## Semantics of CTL*

We define whether $M$ satisfies $\psi$,

$$\begin{aligned} &① \vDash ② \quad \in \mathsf{TModel} \to \mathsf{StateProp} \to \mathbb{B} \\ &M \vDash \psi \stackrel{def}{=} \forall s \in M.S.\ M.S_0\ s \Rightarrow s \vDash^{\mathsf{s}}_{M} \psi \end{aligned}$$

using two auxiliary mutually inductive predicates

$$\begin{aligned} &② \vDash^{\mathsf{s}}_{①} ③ \in (M \in \mathsf{TModel}) \to M.S \to \mathsf{StateProp} \to \mathbb{B} \\ &② \vDash^{\mathsf{p}}_{①} ③ \in (M \in \mathsf{TModel}) \to \mathsf{stream}\ M.S \to \mathsf{PathProp} \to \mathbb{B} \end{aligned}$$

We write the arguments that remain constant through recursive calls in this shade of grey blue.



## Semantics of CTL*: state properties

$$\begin{aligned}
s \vDash^{\mathsf{s}}_{M} \top &\stackrel{def}{=} \top \\
s \vDash^{\mathsf{s}}_{M} \bot &\stackrel{def}{=} \bot \\
s \vDash^{\mathsf{s}}_{M} \psi_1 \wedge^{\mathsf{s}} \psi_2 &\stackrel{def}{=} \left(s \vDash^{\mathsf{s}}_{M} \psi_1\right) \wedge \left(s \vDash^{\mathsf{s}}_{M} \psi_2\right) \\
s \vDash^{\mathsf{s}}_{M} \psi_1 \vee^{\mathsf{s}} \psi_2 &\stackrel{def}{=} \left(s \vDash^{\mathsf{s}}_{M} \psi_1\right) \vee \left(s \vDash^{\mathsf{s}}_{M} \psi_2\right) \\
s \vDash^{\mathsf{s}}_{M} \psi_1 \to^{\mathsf{s}} \psi_2 &\stackrel{def}{=} \neg \left(s \vDash^{\mathsf{s}}_{M} \psi_1\right) \vee \left(s \vDash^{\mathsf{s}}_{M} \psi_2\right) \\
s \vDash^{\mathsf{s}}_{M} \mathsf{inj}\ p &\stackrel{def}{=} M.\ell\ s\ p \quad (\text{"}M.\ell\ s \text{ includes } p\text{"}) \\
s \vDash^{\mathsf{s}}_{M} \mathsf{A}\ \phi &\stackrel{def}{=} \left(\begin{array}{l} \forall \pi \in \mathsf{stream}\ M.S. \\ \quad \mathsf{IsPath}\ M\ \pi \Rightarrow \pi\ 0 = s \Rightarrow \pi \vDash^{\mathsf{p}}_{M} \phi \end{array}\right) \\
s \vDash^{\mathsf{s}}_{M} \mathsf{E}\ \phi &\stackrel{def}{=} \left(\begin{array}{l} \exists \pi \in \mathsf{stream}\ M.S. \\ \quad \mathsf{IsPath}\ M\ \pi \wedge \pi\ 0 = s \wedge \\ \quad \pi \vDash^{\mathsf{p}}_{M} \phi \end{array}\right)
\end{aligned}$$

## Semantics of CTL*: path properties

$$
\begin{array}{ll}
\pi \vDash^{\mathsf{p}}_M \mathsf{injs}\ \psi & \stackrel{def}{=} (\pi\ 0) \vDash^{\mathsf{s}}_M \psi \\
\pi \vDash^{\mathsf{p}}_M \phi_1\ \wedge^{\mathsf{p}}\ \phi_2 & \stackrel{def}{=} (\pi \vDash^{\mathsf{p}}_M \phi_1) \wedge (\pi \vDash^{\mathsf{p}}_M \phi_2) \\
\pi \vDash^{\mathsf{p}}_M \phi_1\ \vee^{\mathsf{p}}\ \phi_2 & \stackrel{def}{=} (\pi \vDash^{\mathsf{p}}_M \phi_1) \vee (\pi \vDash^{\mathsf{p}}_M \phi_2) \\
\pi \vDash^{\mathsf{p}}_M \phi_1\ \rightarrow^{\mathsf{p}}\ \phi_2 & \stackrel{def}{=} \neg (\pi \vDash^{\mathsf{p}}_M \phi_1) \vee (\pi \vDash^{\mathsf{p}}_M \phi_2) \\
\pi \vDash^{\mathsf{p}}_M \mathsf{X}\ \phi & \stackrel{def}{=} (\mathsf{tailn}\ M.S\ 1\ \pi) \vDash^{\mathsf{p}}_M \phi \\
\pi \vDash^{\mathsf{p}}_M \mathsf{F}\ \phi & \stackrel{def}{=} \exists n \in \mathbb{N}.\, (\mathsf{tailn}\ M.S\ n\ \pi) \vDash^{\mathsf{p}}_M \phi \\
\pi \vDash^{\mathsf{p}}_M \mathsf{G}\ \phi & \stackrel{def}{=} \forall n \in \mathbb{N}.\, (\mathsf{tailn}\ M.S\ n\ \pi) \vDash^{\mathsf{p}}_M \phi \\
\pi \vDash^{\mathsf{p}}_M \phi_1\ \mathsf{U}\ \phi_2 \stackrel{def}{=} & \\
\exists n \in \mathbb{N}. \begin{pmatrix} (\forall k \in \mathbb{N}.\, 0 \leq k < n \Rightarrow (\mathsf{tailn}\ M.S\ k\ \pi) \vDash^{\mathsf{p}}_M \phi_1) \wedge \\ (\mathsf{tailn}\ M.S\ n\ \pi) \vDash^{\mathsf{p}}_M \phi_2 \end{pmatrix} &
\end{array}
$$



CTL* is very expressive, but model checking CTL* is computationally expensive.



## CTL* fragments

- CTL: by forcing all uses of temporal operators to come with a unique path quantifier
- LTL: (roughly) by enforcing that there is no path quantification, just one implicit leading A quantifier at the start of the formula
- ACTL*: the universal fragment of CTL*, where all A are under an even number of negations, all E under an odd number of negations
- ECTL*: the existential fragment of CTL*, dual to ACTL*



## Quantifiers

Unlike in Hoare logic, there are no "normal"/non-path quantifiers, as they would make it difficult to mechanically check properties.

To partly make up for this, we can use property schemas with big operators or bounded quantifiers, and indexed atomic propositions, which stand for the expanded property.

For example $\bigwedge_{i \in S} p_i$, for $S = \{1, 2, 3\}$, is expanded to $p_1 \wedge p_2 \wedge p_3$.

This can done "by preprocessing", without changing the language of properties. As a result, this is not as general as quantifiers, as the value of $S$ has to be constant (it cannot depend on properties of states, for instance).

## Summary

Temporal logics can be used to specify temporal models. LTL specifies temporal models in terms of properties of linear paths through the model.

CTL can express properties about the sets of possible paths out of states of a temporal model. The two logics are incomparable; CTL* combines their expressive power.