Possibility is not expressible in LTL

Example:
\[ \varphi \text{: Whenever } p \text{ holds, it is possible to reach a state where } q \text{ holds.} \]
\[ \varphi \text{ cannot be expressed in LTL.} \]

We need quantifications on runs:
- \( E \) for some infinite run
- \( A \) for all infinite runs

CTL* (Emerson & Halpern 86)

Definition: Syntax of the Computation Tree Logic CTL*
\[ \varphi ::= \bot \mid p \mid \neg \varphi \mid \varphi \lor \varphi \mid X \varphi \mid \varphi U \varphi \mid E \varphi \mid A \varphi \]

In this chapter, temporal modalities U, F, G, . . . are non-strict.
We may also add past modalities Y and S

Definition: Semantics of CTL*
Let \( M = (S, T, I, AP, t) \) be a Kripke structure.
Let \( \sigma = s_0s_1s_2 \cdots \) be an infinite run of \( M \).
\[ M, \sigma, i \models E \varphi \text{ if } M, \sigma', i \models \varphi \text{ for some infinite run } \sigma' \text{ such that } \sigma'[i] = \sigma[i] \]
\[ M, \sigma, i \models A \varphi \text{ if } M, \sigma', i \models \varphi \text{ for all infinite runs } \sigma' \text{ such that } \sigma'[i] = \sigma[i] \]
where \( \sigma[i] = s_0 \cdots s_i \).

Remark:
- \( A \varphi \equiv \neg E \neg \varphi \)
- \( \sigma'[i] = \sigma[i] \) means that future is branching but past is not.
CTL* (Emerson & Halpern 86)

Example: Some specifications
- EF $\varphi$: $\varphi$ is possible
- AG $\varphi$: $\varphi$ is an invariant
- AF $\varphi$: $\varphi$ is unavoidable
- EG $\varphi$: $\varphi$ holds globally along some path

State formulae and path formulae

Definition: State formulae
$\varphi \in \text{CTL}^*$ is a state formula if for all $M, \sigma, \sigma', i, j$ such that $\sigma(i) = \sigma'(j)$ we have:

$M, \sigma, i \models \varphi \iff M, \sigma', j \models \varphi$

If $\varphi$ is a state formula and $M = (S, T, I, AP, \ell)$, define:

$[\varphi]^M = \{ s \in S \mid M, s \models \varphi \}$

Example: State formulae
Atomic propositions are state formulae:
$\lbrack p \rbrack = \{ s \in S \mid p \in \ell(s) \}$

State formulae are closed under boolean connectives:
$\lbrack \neg \varphi \rbrack = S \setminus \lbrack \varphi \rbrack$
$\lbrack \varphi_1 \lor \varphi_2 \rbrack = \lbrack \varphi_1 \rbrack \cup \lbrack \varphi_2 \rbrack$

Formulate of the form $E \varphi$ or $A \varphi$ are state formulae, provided $\varphi$ is future.

Definition: Alternative syntax
- State formulae $\varphi ::= \bot \mid p (p \in AP) \mid \neg \varphi \mid \varphi \lor \varphi \mid E \psi \mid A \psi$
- Path formulae $\psi ::= \varphi \mid \neg \varphi \mid \psi \lor \psi \mid X \psi \mid \psi U \psi$

Model checking of CTL*

Definition: Existential and universal model checking
Let $M = (S, T, I, AP, \ell)$ be a Kripke structure and $\varphi \in \text{CTL}^*$ a formula.

$M \models_\exists \varphi$ if $M, \sigma, 0 \models \varphi$ for some initial infinite run $\sigma$ of $M$.

$M \models_\forall \varphi$ if $M, \sigma, 0 \models \varphi$ for all initial infinite run $\sigma$ of $M$.

Remark:

$M \models_\exists \varphi$ if $I \cap \lbrack E \varphi \rbrack \neq \emptyset$

$M \models_\forall \varphi$ if $I \subseteq \lbrack A \varphi \rbrack$

$M \models_\forall \neg \varphi$ if $M \not \models_\exists \neg \varphi$

Definition: Model checking problems $\text{MC}^\exists_{\text{CTL}^*}$ and $\text{MC}^\forall_{\text{CTL}^*}$

Input: A Kripke structure $M = (S, T, I, AP, \ell)$ and a formula $\varphi \in \text{CTL}^*$

Question: Does $M \models \forall \varphi$? or Does $M \models \exists \varphi$?

Complexity of CTL*

Definition: Syntax of the Computation Tree Logic CTL*

$\varphi ::= \bot \mid p (p \in AP) \mid \neg \varphi \mid \varphi \lor \varphi \mid X \varphi \mid \varphi U \varphi \mid E \varphi \mid A \varphi$

Theorem
The model checking problem for CTL* is PSPACE-complete.

Proof:
PSPACE-hardness: follows from $\text{LTL} \subseteq \text{CTL}^*$.
PSPACE-easiness: reduction to LTL-model checking by inductive eliminations of path quantifications.
**MC^3_{\text{CTL}*} in PSPACE**

Proof:

For $\psi \in \text{LTL}$, let $MC^3_{\text{LTL}}(M, t, \psi)$ be the function which computes in polynomial space whether $M, t \models \psi$, i.e., if $M, t \models E \psi$.

Let $M = (S, T, I, \text{AP}, \ell)$ be a Kripke structure, $s \in S$ and $\phi \in \text{CTL}^*$. Replacing $A \psi$ by $\neg E \neg \psi$ we assume $\phi$ only contains the existential path quantifier.

$MC^3_{\text{CTL}*}(M, s, \phi)$

If $E$ does not occur in $\phi$ then return $MC^3_{\text{LTL}}(M, s, \phi)$.

Let $E \psi$ be a subformula of $\phi$ with $\psi \in \text{LTL}$.

Let $e_\psi$ be a new propositional variable.

Define $\ell'(t) \cap \text{AP} = \ell(t)$ and $e_\psi \in \ell'(t)$ if $MC^3_{\text{LTL}}(M, t, \psi)$.

Let $M' = (S, T, I, \text{AP}', \ell')$.

Let $\phi' = \phi[e_\psi / E \psi]$ be obtained from $\phi$ by replacing each $E \psi$ by $e_\psi$.

Return $MC^3_{\text{CTL}*}(M', s, \phi')$.

---

**Satisfiability for CTL***

Definition: SAT(CTL*)

Input: A formula $\varphi \in \text{CTL}^*$

Question: Existence of a model $M$ and a run $\sigma$ such that $M, \sigma, 0 \models \varphi$?

Theorem

The satisfiability problem for $\text{CTL}^*$ is 2-EXPTIME-complete.

---

**CTL (Clarke & Emerson 81)**

Definition: Computation Tree Logic (CTL)

Syntax:

$\varphi ::= \bot \mid p \ (p \in \text{AP}) \mid \neg \varphi \mid \varphi \lor \varphi \mid EX \varphi \mid AX \varphi \mid E \varphi U \varphi \mid A \varphi U \varphi$

The semantics is inherited from $\text{CTL}^*$.

Remark: All CTL formulae are state formulae.

$[\varphi]^M = \{s \in S \mid M, s \models \varphi\}$

Examples: Macros

- $EF \varphi = E T U \varphi$ and $AF \varphi = A T U \varphi$
- $EG \varphi = \neg AF \neg \varphi$ and $AG \varphi = \neg EF \neg \varphi$
- $AG(req \rightarrow EF\ grant)$
- $AG(req \rightarrow AF\ grant)$
**Definition: Semantics**

All CTL-formulae are state formulae. Hence, we have a simpler semantics. Let $M = (S, T, I, AP, t)$ be a Kripke structure without deadlocks and let $s \in S$.

- $s \models p$ if $p \in \ell(s)$
- $s \models \text{EX} \varphi$ if $\exists s' \text{ with } s' \models \varphi$
- $s \models \text{AX} \varphi$ if $\forall s \text{ with } s' \models \varphi$
- $s \models \text{E} \varphi \text{ U } \psi$ if $\exists s = s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \cdots s_j \text{ finite path, with } s_j \models \psi \text{ and } s_k \models \varphi \text{ for all } 0 \leq k < j$
- $s \models \text{A} \varphi \text{ U } \psi$ if $\forall s = s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \cdots \text{ infinite path, } \exists j \geq 0 \text{ with } s_j \models \psi \text{ and } s_k \models \varphi \text{ for all } 0 \leq k < j$

**Remark: Equivalent formulae**

- $\text{AX} \varphi = \neg \text{EX} \neg \varphi$,
- $\neg (\varphi \text{ U } \psi) = \text{G} \neg \psi \lor (\neg \psi \text{ U } (\neg \varphi \land \neg \psi))$
- $\text{A} \varphi \text{ U } \psi = \neg \text{EG} \neg \psi \land \neg \text{E} (\neg \psi \text{ U } (\neg \varphi \land \neg \psi))$
- $\text{AG} (\text{req} \rightarrow \text{F grant}) = \text{AG} (\text{req} \rightarrow \text{AF grant})$
- $\text{AGF} \varphi = \text{AGAF} \varphi$ (infinitely often)
- $\text{EFG} \varphi = \text{EFEG} \varphi$ (ultimately)
- $\text{GEF} \varphi \neq \text{EGF} \varphi$
- $\text{AFAG} \varphi \neq \text{AFG} \varphi$
- $\text{EGEX} \varphi \neq \text{EGX} \varphi$

**CTL (Clarke & Emerson 81)**

**Example:**

![Kripke structure](image)

- $[\text{EX} \varphi] = \{5, 6\}$
- $[\text{AX} \varphi] = \{2\}$
- $[\text{EF} \varphi] = \{2, 6\}$
- $[\text{AF} \varphi] = \{2\}$
- $[\text{EQ} \text{ U } r] = \{5, 6\}$
- $[\text{AQ} \text{ U } r] = \{5, 6\}$

**Model checking of CTL**

**Definition: Existential and universal model checking**

Let $M = (S, T, I, AP, t)$ be a Kripke structure and $\varphi \in \text{CTL}$ a formula.

- $M \models_{\exists} \varphi$ if $M, s \models \varphi$ for some $s \in I$.
- $M \models_{\forall} \varphi$ if $M, s \models \varphi$ for all $s \in I$.

**Remark:**

- $M \models_{\exists} \varphi$ iff $I \cap [\varphi] \neq \emptyset$
- $M \models_{\forall} \varphi$ iff $I \subseteq [\varphi]$
- $M \models_{\exists} \varphi$ iff $M \models_{\forall} \neg \varphi$

**Definition: Model checking problems $\text{MC}^\forall_{\text{CTL}}$ and $\text{MC}^\exists_{\text{CTL}}$**

**Input:** A Kripke structure $M = (S, T, I, AP, t)$ and a formula $\varphi \in \text{CTL}$.

**Question:** Does $M \models_{\forall} \varphi$? or Does $M \models_{\exists} \varphi$?
Model checking of CTL

**Theorem**

Let $M = (S, T, I, AP, \ell)$ be a Kripke structure and $\varphi \in \text{CTL}$ a formula. The model checking problem $M \models \varphi$ is decidable in time $O(|M| + |\varphi|)$.

**Proof:**

Compute $\llbracket \varphi \rrbracket = \{ s \in S \mid M, s \models \varphi \}$ by induction on the formula.

The set $\llbracket \varphi \rrbracket$ is represented by a boolean array: $L[s][\varphi] = T$ if $s \in \llbracket \varphi \rrbracket$.

The labelling $\ell$ is encoded in $L$: for $p \in AP$ we have $L[s][p] = T$ if $p \in \ell(s)$.

---

**Definition: procedure semantics($\varphi$)**

\[
\begin{align*}
\text{case } \varphi &= \neg \varphi_1 \\
\quad \text{semantics}(\varphi_1) &\quad O(|S|) \\
\llbracket \varphi \rrbracket &:= S \setminus \llbracket \varphi_1 \rrbracket \\
\text{case } \varphi &= \varphi_1 \lor \varphi_2 \\
\quad \text{semantics}(\varphi_1), \text{semantics}(\varphi_2) &\quad O(|S|) \\
\llbracket \varphi \rrbracket &:= \llbracket \varphi_1 \rrbracket \cup \llbracket \varphi_2 \rrbracket \\
\text{case } \varphi &= \text{EX} \varphi_1 \\
\quad \text{semantics}(\varphi_1) &\quad O(|S|) \\
\llbracket \varphi \rrbracket &:= 0 \\
\text{for all } (s, t) \in T \text{ do } &\quad O(|T|) \\
\text{if } t \in \llbracket \varphi_1 \rrbracket &\quad \llbracket \varphi \rrbracket := \llbracket \varphi \rrbracket \cup \{s\} \\
\text{case } \varphi &= \text{AX} \varphi_1 \\
\quad \text{semantics}(\varphi_1) &\quad O(|S|) \\
\llbracket \varphi \rrbracket &:= S \\
\text{for all } (s, t) \in T \text{ do } &\quad O(|T|) \\
\text{if } t \notin \llbracket \varphi_1 \rrbracket &\quad \llbracket \varphi \rrbracket := \llbracket \varphi \rrbracket \setminus \{s\} \\
\end{align*}
\]

$Z$ is only used to make the invariant clear. It can be replaced by $\llbracket \varphi \rrbracket$.
Complexity of CTL

Definition: SAT(CTL)
Input: A formula $\varphi \in \text{CTL}$
Question: Existence of a model $M$ and a state $s$ such that $M, s \models \varphi$?

Theorem: Complexity
- The model checking problem for CTL is PTIME-complete.
- The satisfiability problem for CTL is EXPTIME-complete.

Outline

Introduction
Models
Specifications
Satisfiability and Model Checking for LTL

Branching Time Specifications
- CTL$^*$
- CTL
- Fair CTL

fairness

Example: Fairness
Only fair runs are of interest
- Each process is enabled infinitely often: $\bigwedge_i \text{GF run}_i$
- No process stays ultimately in the critical section: $\bigwedge_i \neg \text{FG CS}_i = \bigwedge_i \text{GF} \neg \text{CS}_i$

Definition: Fair Kripke structure
$M = (S, T, I, \text{AP}, \ell, F_1, \ldots, F_n)$ with $F_i \subseteq S$.
An infinite run $\sigma$ is fair if it visits infinitely often each $F_i$

fair CTL

Definition: Syntax of fair-CTL
$\varphi ::= \bot | p \ (p \in \text{AP}) | \neg \varphi | \varphi \lor \varphi | \text{EF} \varphi | \text{AF} \varphi | \text{EF} \varphi \text{U} \varphi | \text{AF} \varphi \text{U} \varphi$

Definition: Semantics as a fragment of CTL$^*$
Let $M = (S, T, I, \text{AP}, \ell, F_1, \ldots, F_n)$ be a fair Kripke structure.

Then,
$\text{EF} \varphi = E(\text{fair} \land \varphi)$ and $\text{AF} \varphi = A(\text{fair} \rightarrow \varphi)$

where $\text{fair} = \bigwedge_i \text{GF} F_i$

Lemma: CTL$^f$ cannot be expressed in CTL

fair CTL

Proof: CTL\(_f\) cannot be expressed in CTL

Consider the Kripke structure \(M_k\) defined by:

\[
\begin{array}{cccccccc}
\frac{2k}{p} & \frac{2k-1}{p} & \frac{2k-2}{p} & \frac{2k-3}{p} & \cdots & \frac{4}{p} & \frac{3}{p} & \frac{2}{p} & \frac{1}{p}
\end{array}
\]

- \(M_k, 2k \models EG F p\) but \(M_k, 2k - 2 \nless EG F p\)
- If \(\varphi \in \text{CTL}\) and \(|\varphi| \leq m \leq k\) then
  - \(M_k, 2k \models \varphi\) iff \(M_k, 2m \models \varphi\)
  - \(M_k, 2k - 1 \models \varphi\) iff \(M_k, 2m - 1 \models \varphi\)

If the fairness condition is \(\ell^{-1}(p)\) then \(E_f T\) cannot be expressed in CTL.

Model checking of CTL\(_f\)

**Theorem**
The model checking problem for CTL\(_f\) is decidable in time \(O(|M| \cdot |\varphi|)\)

**Proof:** Computation of \(\text{Fair} = \{ s \in S \mid M, s \models E_f T \}\)

Compute the SCC of \(M\) with Tarjan’s algorithm (in time \(O(|M|)\)).
Let \(S'\) be the union of the (non trivial) SCCs which intersect each \(F_i\).
Then, \(\text{Fair}\) is the set of states that can reach \(S'\).
Note that reachability can be computed in linear time.

Proof: Reductions

\[
E_f X \varphi = E X (\text{Fair} \land \varphi) \quad \text{and} \quad E_f \varphi U \psi = E \varphi U (\text{Fair} \land \psi)
\]

It remains to deal with \(A_f \varphi U \psi\).
We have

\[
A_f \varphi U \psi = \neg E_f G \neg \psi \land \neg E_f (\neg \psi U (\neg \varphi \land \neg \psi))
\]

Hence, we only need to compute the semantics of \(E_f G \varphi\).

Proof: Computation of \(E_f G \varphi\)

Let \(M_\varphi\) be the restriction of \(M\) to \([\varphi]_f\).
Compute the SCC of \(M_\varphi\) with Tarjan’s algorithm (in linear time).
Let \(S'\) be the union of the (non trivial) SCCs of \(M_\varphi\) which intersect each \(F_i\).
Then, \(M, s \models E_f G \varphi\) iff \(M, s \models E \varphi U S'\) iff \(M_\varphi, s \models EF S'\).
This is again a reachability problem which can be solved in linear time.