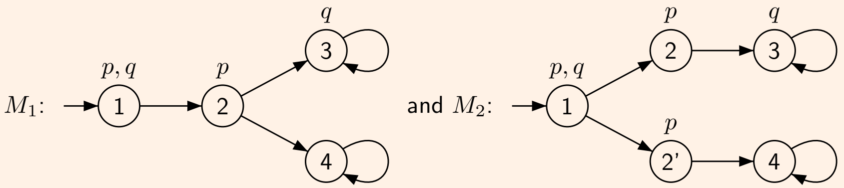
# Possibility is not expressible in LTL

Example:

$\varphi$: Whenever $p$ holds, it is possible to reach a state where $q$ holds.
$\varphi$ cannot be expressed in LTL.

Consider the two models:




$M_1 \models \varphi$ but $M_2 \not\models \varphi$
$M_1$ and $M_2$ satisfy the same LTL formulae.

We need quantifications on runs: $\varphi = \mathsf{AG}(p \to \mathsf{EF}\, q)$

- E: for some infinite run
- A: for all infinite runs



# $\mathrm{CTL}^*$ (Emerson & Halpern 86)

Definition: Syntax of the Computation Tree Logic $\mathrm{CTL}^*$

$$\varphi ::= \bot \mid p \; (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{X}\,\varphi \mid \varphi\,\mathsf{U}\,\varphi \mid \mathsf{E}\,\varphi \mid \mathsf{A}\,\varphi$$

Definition: Semantics:

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure and $\sigma$ an infinte run of $M$.

$M, \sigma, i \models \mathsf{E}\varphi$ if $M, \sigma', 0 \models \varphi$ for some infinite run $\sigma'$ such that $\sigma'(0) = \sigma(i)$
$M, \sigma, i \models \mathsf{A}\varphi$ if $M, \sigma', 0 \models \varphi$ for all infinite runs $\sigma'$ such that $\sigma'(0) = \sigma(i)$

Example: Some specifications

- $\mathsf{EF}\,\varphi$: $\varphi$ is possible
- $\mathsf{AG}\,\varphi$: $\varphi$ is an invariant
- $\mathsf{AF}\,\varphi$: $\varphi$ is unavoidable
- $\mathsf{EG}\,\varphi$: $\varphi$ holds globally along some path

Remark: $\mathsf{A}\,\varphi \equiv \neg\,\mathsf{E}\,\neg\varphi$



# State formulae and path formulae

Definition: State formulae

$\varphi \in \mathrm{CTL}^*$ is a state formula if $\forall M, \sigma, \sigma', i, j$ 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 $M = (S, T, I, \mathrm{AP}, \ell)$, define

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

Example: State formulae

Formulae of the form $p$ or $\mathsf{E}\,\varphi$ or $\mathsf{A}\,\varphi$ are state formulae.
State formulae are closed under boolean connectives.

$$[\![p]\!] = \{s \in S \mid p \in \ell(s)\} \qquad [\![\neg\varphi]\!] = S \setminus [\![\varphi]\!] \qquad [\![\varphi_1 \vee \varphi_2]\!] = [\![\varphi_1]\!] \cup [\![\varphi_2]\!]$$

Definition: Alternative syntax

State formulae $\varphi ::= \bot \mid p \; (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{E}\,\psi \mid \mathsf{A}\,\psi$
Path formulae $\psi ::= \varphi \mid \neg\psi \mid \psi \vee \psi \mid \mathsf{X}\,\psi \mid \psi\,\mathsf{U}\,\psi$



# Model checking of $\mathrm{CTL}^*$

Definition: Existential and universal model checking

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure and $\varphi \in \mathrm{CTL}^*$ 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:

$M \models_\exists \varphi$ iff $I \cap [\![\mathsf{E}\,\varphi]\!] \neq \emptyset$
$M \models_\forall \varphi$ iff $I \subseteq [\![\mathsf{A}\,\varphi]\!]$
$M \models_\forall \varphi$ iff $M \not\models_\exists \neg\varphi$

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

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

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

## Complexity of $\mathrm{CTL}^*$

**Definition: Syntax of the Computation Tree Logic $\mathrm{CTL}^*$**

$$\varphi ::= \bot \mid p\ (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{X}\,\varphi \mid \varphi\,\mathsf{U}\,\varphi \mid \mathsf{E}\,\varphi \mid \mathsf{A}\,\varphi$$

**Theorem**

The model checking problem for $\mathrm{CTL}^*$ is PSPACE-complete

**Proof:**

PSPACE-hardness: follows from $\mathrm{LTL} \subseteq \mathrm{CTL}^*$.

PSPACE-easiness: reduction to LTL-model checking by inductive eliminations of path quantifications.



## $\mathrm{MC}^{\forall}_{\mathrm{CTL}^*}$ in PSPACE

**Proof:**

For $\mathcal{Q} \in \{\exists, \forall\}$ and $\psi \in \mathrm{LTL}$, let $\mathrm{MC}^{\mathcal{Q}}_{\mathrm{LTL}}(M, t, \psi)$ be the function which computes in polynomial space whether $M, t \models_{\mathcal{Q}} \psi$, i.e., if $M, t \models \mathcal{Q}\psi$.

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure, $s \in S$ and $\varphi \in \mathrm{CTL}^*$.

$\mathrm{MC}^{\forall}_{\mathrm{CTL}^*}(M, s, \varphi)$
- If $E, \mathsf{A}$ do not occur in $\varphi$ then return $\mathrm{MC}^{\forall}_{\mathrm{LTL}}(M, s, \varphi)$ fi
- Let $\mathcal{Q}\psi$ be a subformula of $\varphi$ with $\psi \in \mathrm{LTL}$ and $\mathcal{Q} \in \{\mathsf{E}, \mathsf{A}\}$
- Let $p_{\mathcal{Q}\psi}$ be a new propositional variable
- Define $\ell' : S \to 2^{\mathrm{AP}'}$ with $\mathrm{AP}' = \mathrm{AP} \uplus \{p_{\mathcal{Q}\psi}\}$ by
  - $\ell'(t) \cap \mathrm{AP} = \ell(t)$ and $p_{\mathcal{Q}\psi} \in \ell'(t)$ iff $\mathrm{MC}^{\mathcal{Q}}_{\mathrm{LTL}}(M, t, \psi)$
- Let $M' = (S, T, I, \mathrm{AP}', \ell')$
- Let $\varphi' = \varphi[p_{\mathcal{Q}\psi}/\mathcal{Q}\psi]$ be obtained from $\varphi$ by replacing each $\mathcal{Q}\psi$ by $p_{\mathcal{Q}\psi}$
- Return $\mathrm{MC}^{\forall}_{\mathrm{CTL}^*}(M', s, \varphi')$



## Satisfiability for $\mathrm{CTL}^*$

**Definition: $\mathrm{SAT}(\mathrm{CTL}^*)$**

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

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

**Theorem**

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



## $\mathrm{CTL}$ (Clarke & Emerson 81)

**Definition: Computation Tree Logic $(\mathrm{CTL})$**

Syntax:

$$\varphi ::= \bot \mid p\ (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{EX}\,\varphi \mid \mathsf{AX}\,\varphi \mid \mathsf{E}\,\varphi\,\mathsf{U}\,\varphi \mid \mathsf{A}\,\varphi\,\mathsf{U}\,\varphi$$

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

**Remark: All CTL formulae are state formulae**

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

**Examples: Macros**

- $\mathsf{EF}\,\varphi = \mathsf{E}\,\top\,\mathsf{U}\,\varphi$ and $\mathsf{AF}\,\varphi = \mathsf{A}\,\top\,\mathsf{U}\,\varphi$
- $\mathsf{EG}\,\varphi = \neg\,\mathsf{AF}\,\neg\varphi$ and $\mathsf{AG}\,\varphi = \neg\,\mathsf{EF}\,\neg\varphi$
- $\mathsf{AG}(\mathrm{req} \to \mathsf{EF}\,\mathrm{grant})$
- $\mathsf{AG}(\mathrm{req} \to \mathsf{AF}\,\mathrm{grant})$

## CTL (Clarke & Emerson 81)

**Definition: Semantics**

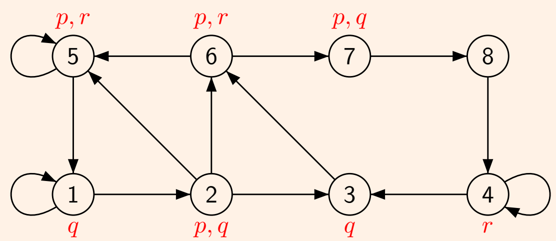
All CTL-formulae are state formulae. Hence, we have a simpler semantics.
Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure without deadlocks and let $s \in S$.

| | | |
|---|---|---|
| $s \models p$ | if | $p \in \ell(s)$ |
| $s \models \mathsf{EX}\,\varphi$ | if | $\exists s \to s'$ with $s' \models \varphi$ |
| $s \models \mathsf{AX}\,\varphi$ | if | $\forall s \to s'$ we have $s' \models \varphi$ |
| $s \models \mathsf{E}\,\varphi\,\mathsf{U}\,\psi$ | if | $\exists s = s_0 \to s_1 \to s_2 \to \cdots s_j$ finite path, with $s_j \models \psi$ and $s_k \models \varphi$ for all $0 \le k < j$ |
| $s \models \mathsf{A}\,\varphi\,\mathsf{U}\,\psi$ | if | $\forall s = s_0 \to s_1 \to s_2 \to \cdots$ infinite path, $\exists j \ge 0$ with $s_j \models \psi$ and $s_k \models \varphi$ for all $0 \le k < j$ |



## CTL (Clarke & Emerson 81)

Example:



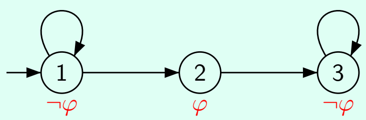

$$[\![\mathsf{EX}\,p]\!] = \{1,2,3,5,6\}$$
$$[\![\mathsf{AX}\,p]\!] = \{3,6\}$$
$$[\![\mathsf{EF}\,p]\!] = \{1,2,3,4,5,6,7,8\}$$
$$[\![\mathsf{AF}\,p]\!] = \{2,3,5,6,7\}$$
$$[\![\mathsf{E}\,q\,\mathsf{U}\,r]\!] = \{1,2,3,4,5,6\}$$
$$[\![\mathsf{A}\,q\,\mathsf{U}\,r]\!] = \{2,3,4,5,6\}$$



## CTL (Clarke & Emerson 81)

**Remark: Equivalent formulae**

- $\mathsf{AX}\,\varphi = \neg\,\mathsf{EX}\,\neg\varphi$,
- $\neg(\varphi\,\mathsf{U}\,\psi) = \mathsf{G}\,\neg\psi \vee (\neg\psi\,\mathsf{U}\,(\neg\varphi \wedge \neg\psi))$
- $\mathsf{A}\,\varphi\,\mathsf{U}\,\psi = \neg\,\mathsf{EG}\,\neg\psi \wedge \neg\,\mathsf{E}\,\neg\psi\,\mathsf{U}\,(\neg\varphi \wedge \neg\psi)$
- $\mathsf{AG}(\mathrm{req} \to \mathsf{F}\,\mathrm{grant}) = \mathsf{AG}(\mathrm{req} \to \mathsf{AF}\,\mathrm{grant})$
- $\mathsf{A\,G\,F}\,\varphi = \mathsf{AG\,AF}\,\varphi$ — infinitely often
- $\mathsf{E\,F\,G}\,\varphi = \mathsf{EF\,EG}\,\varphi$ — ultimately
- $\mathsf{EG\,EF}\,\varphi \ne \mathsf{E\,G\,F}\,\varphi$
- $\mathsf{AF\,AG}\,\varphi \ne \mathsf{A\,F\,G}\,\varphi$
- $\mathsf{EG\,EX}\,\varphi \ne \mathsf{E\,G\,X}\,\varphi$






## Model checking of CTL

**Definition: Existential and universal model checking**

Let $M = (S, T, I, \mathrm{AP}, \ell)$ be a Kripke structure and $\varphi \in \mathrm{CTL}$ 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 $I \cap [\![\varphi]\!] \ne \emptyset$

$M \models_\forall \varphi$ iff $I \subseteq [\![\varphi]\!]$

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

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

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

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

# Model checking of CTL

**Theorem**

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

**Proof:**

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

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

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



# Model checking of CTL

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

case $\varphi = \neg\varphi_1$
 semantics($\varphi_1$)
 $[\![\varphi]\!] := S \setminus [\![\varphi_1]\!]$ — $\mathcal{O}(|S|)$

case $\varphi = \varphi_1 \vee \varphi_2$
 semantics($\varphi_1$); semantics($\varphi_2$)
 $[\![\varphi]\!] := [\![\varphi_1]\!] \cup [\![\varphi_2]\!]$ — $\mathcal{O}(|S|)$

case $\varphi = EX\varphi_1$
 semantics($\varphi_1$)
 $[\![\varphi]\!] := \emptyset$ — $\mathcal{O}(|S|)$
 for all $(s,t) \in T$ do if $t \in [\![\varphi_1]\!]$ then $[\![\varphi]\!] := [\![\varphi]\!] \cup \{s\}$ — $\mathcal{O}(|T|)$

case $\varphi = AX\varphi_1$
 semantics($\varphi_1$)
 $[\![\varphi]\!] := S$ — $\mathcal{O}(|S|)$
 for all $(s,t) \in T$ do if $t \notin [\![\varphi_1]\!]$ then $[\![\varphi]\!] := [\![\varphi]\!] \setminus \{s\}$ — $\mathcal{O}(|T|)$



# Model checking of CTL

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

case $\varphi = E\varphi_1 \mathsf{U} \varphi_2$ — $\mathcal{O}(|S| + |T|)$
 semantics($\varphi_1$); semantics($\varphi_2$)
 $L := [\![\varphi_2]\!]$ // the set $L$ is the "todo" list — $\mathcal{O}(|S|)$
 $Z := \emptyset$ // the set $Z$ is the "done" list — $\mathcal{O}(|S|)$
 while $L \neq \emptyset$ do — $|S|$ times
 Invariant: $[\![\varphi_2]\!] \cup ([\![\varphi_1]\!] \cap T^{-1}(Z)) \subseteq Z \cup L \subseteq [\![\mathsf{E}\,\varphi_1 \mathsf{U}\, \varphi_2]\!]$
  take $t \in L$; $L := L \setminus \{t\}$; $Z := Z \cup \{t\}$ — $\mathcal{O}(1)$
  for all $s \in T^{-1}(t)$ do — $|T|$ times
   if $s \in [\![\varphi_1]\!] \setminus (Z \cup L)$ then $L := L \cup \{s\}$
 $[\![\varphi]\!] := Z$

Z is only used to make the invariant clear.
$Z \cup L$ can be replaced by $[\![\varphi]\!]$.



# Model checking of CTL

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

Replacing $Z \cup L$ by $[\![\varphi]\!]$

case $\varphi = E\varphi_1 \mathsf{U} \varphi_2$ — $\mathcal{O}(|S| + |T|)$
 semantics($\varphi_1$); semantics($\varphi_2$)
 $L := [\![\varphi_2]\!]$ // the set L is imlemented with a list — $\mathcal{O}(|S|)$
 $[\![\varphi]\!] := [\![\varphi_2]\!]$ — $\mathcal{O}(|S|)$
 while $L \neq \emptyset$ do — $|S|$ times
  take $t \in L$; $L := L \setminus \{t\}$ — $\mathcal{O}(1)$
  for all $s \in T^{-1}(t)$ do — $|T|$ times
   if $s \in [\![\varphi_1]\!] \setminus [\![\varphi]\!]$ then $L := L \cup \{s\}$; $[\![\varphi]\!] := [\![\varphi]\!] \cup \{s\}$ — $\mathcal{O}(1)$

# Model checking of CTL

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

case $\varphi = A\varphi_1 \,\mathsf{U}\, \varphi_2$ — $\mathcal{O}(|S|+|T|)$
semantics($\varphi_1$); semantics($\varphi_2$)
$L := [\![\varphi_2]\!]$ // the set $L$ is the "todo" list — $\mathcal{O}(|S|)$
$Z := \emptyset$ // the set $Z$ is the "done" list — $\mathcal{O}(|S|)$
for all $s \in S$ do $c[s] := |T(s)|$ — $\mathcal{O}(|S|)$
while $L \neq \emptyset$ do — $|S|$ times
Invariant: $\forall s \in S,\ c[s] = |T(s) \setminus Z|$ and
$[\![\varphi_2]\!] \cup ([\![\varphi_1]\!] \cap \{s \in S \mid T(s) \subseteq Z\}) \subseteq Z \cup L \subseteq [\![\mathsf{A}\,\varphi_1 \,\mathsf{U}\, \varphi_2]\!]$
take $t \in L$; $L := L \setminus \{t\}$; $Z := Z \cup \{t\}$ — $\mathcal{O}(1)$
for all $s \in T^{-1}(t)$ do — $|T|$ times
$c[s] := c[s] - 1$ — $\mathcal{O}(1)$
if $c[s] = 0 \wedge s \in [\![\varphi_1]\!] \setminus (Z \cup L)$ then $L := L \cup \{s\}$
$[\![\varphi]\!] := Z$

$Z$ is only used to make the invariant clear.
$Z \cup L$ can be replaced by $[\![\varphi]\!]$.



# Model checking of CTL

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

Replacing $Z \cup L$ by $[\![\varphi]\!]$

case $\varphi = A\varphi_1 \,\mathsf{U}\, \varphi_2$ — $\mathcal{O}(|S|+|T|)$
semantics($\varphi_1$); semantics($\varphi_2$)
$L := [\![\varphi_2]\!]$ // the set L is imlemented with a list — $\mathcal{O}(|S|)$
$[\![\varphi]\!] := [\![\varphi_2]\!]$ — $\mathcal{O}(|S|)$
for all $s \in S$ do $c[s] := |T(s)|$ — $\mathcal{O}(|S|)$
while $L \neq \emptyset$ do — $|S|$ times
take $t \in L$; $L := L \setminus \{t\}$ — $\mathcal{O}(1)$
for all $s \in T^{-1}(t)$ do — $|T|$ times
$c[s] := c[s] - 1$ — $\mathcal{O}(1)$
if $c[s] = 0 \wedge s \in [\![\varphi_1]\!] \setminus [\![\varphi]\!]$ then — $\mathcal{O}(1)$
$L := L \cup \{s\}$; $[\![\varphi]\!] := [\![\varphi]\!] \cup \{s\}$ — $\mathcal{O}(1)$



# Complexity of CTL

**Definition: SAT(CTL)**

Input: A formula $\varphi \in \mathrm{CTL}$

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.



# fairness

**Example: Fairness**

Only fair runs are of interest

- Each process is enabled infinitely often: $\bigwedge_i \mathsf{G}\,\mathsf{F}\,\mathrm{run}_i$
- No process stays ultimately in the critical section: $\bigwedge_i \neg\,\mathsf{F}\,\mathsf{G}\,\mathrm{CS}_i = \bigwedge_i \mathsf{G}\,\mathsf{F}\,\neg\mathrm{CS}_i$

**Definition: Fair Kripke structure**

$M = (S, T, I, \mathrm{AP}, \ell, F_1, \ldots, F_n)$ 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**

$\varphi ::= \bot \mid p\ (p \in \mathrm{AP}) \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{E}_f\,\mathsf{X}\,\varphi \mid \mathsf{A}_f\,\mathsf{X}\,\varphi \mid \mathsf{E}_f\,\varphi\,\mathsf{U}\,\varphi \mid \mathsf{A}_f\,\varphi\,\mathsf{U}\,\varphi$

**Definition: Semantics as a fragment of $\mathrm{CTL}^*$**

Let $M = (S, T, I, \mathrm{AP}, \ell, F_1, \ldots, F_n)$ be a fair Kripke structure.

Then, $\mathsf{E}_f\,\varphi = \mathsf{E}(\mathrm{fair} \wedge \varphi)$ and $\mathsf{A}_f\,\varphi = \mathsf{A}(\mathrm{fair} \rightarrow \varphi)$

where $\mathrm{fair} = \bigwedge_i \mathsf{G}\,\mathsf{F}\,F_i$

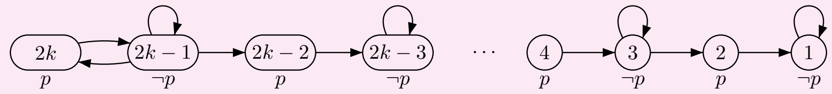
**Lemma: $\mathrm{CTL}_f$ cannot be expressed in $\mathrm{CTL}$**



# fair CTL

**Proof: $\mathrm{CTL}_f$ cannot be expressed in $\mathrm{CTL}$**

Consider the Kripke structure $M_k$ defined by:




- $M_k, 2k \models \mathsf{E}\,\mathsf{G}\,\mathsf{F}\,p$ but $M_k, 2k-2 \not\models \mathsf{E}\,\mathsf{G}\,\mathsf{F}\,p$
- If $\varphi \in \mathrm{CTL}$ and $|\varphi| \leq m \leq k$ then

$$M_k, 2k \models \varphi \text{ iff } M_k, 2m \models \varphi$$

$$M_k, 2k-1 \models \varphi \text{ iff } M_k, 2m-1 \models \varphi$$

If the fairness condition is $\ell^{-1}(p)$ then $\mathsf{E}_f\,\top$ cannot be expressed in $\mathrm{CTL}$.



# Model checking of $\mathrm{CTL}_f$

**Theorem**

The model checking problem for $\mathrm{CTL}_f$ is decidable in time $\mathcal{O}(|M| \cdot |\varphi|)$

**Proof: Computation of $\mathrm{Fair} = \{s \in S \mid M, s \models \mathsf{E}_f\,\top\}$**

Compute the SCC of $M$ with Tarjan's algorithm (in time $\mathcal{O}(|M|)$).

Let $S'$ be the union of the (non trivial) SCCs which intersect each $F_i$.

Then, $\mathrm{Fair}$ is the set of states that can reach $S'$.

Note that reachability can be computed in linear time.



# Model checking of $\mathrm{CTL}_f$

**Proof: Reductions**

$\mathsf{E}_f\,\mathsf{X}\,\varphi = \mathsf{E}\,\mathsf{X}(\mathrm{Fair} \wedge \varphi)$ and $\mathsf{E}_f\,\varphi\,\mathsf{U}\,\psi = \mathsf{E}\,\varphi\,\mathsf{U}\,(\mathrm{Fair} \wedge \psi)$

It remains to deal with $\mathsf{A}_f\,\varphi\,\mathsf{U}\,\psi$.

Recall that $\mathsf{A}\,\varphi\,\mathsf{U}\,\psi = \neg\,\mathsf{EG}\,\neg\psi \wedge \neg\,\mathsf{E}\,\neg\psi\,\mathsf{U}\,(\neg\varphi \wedge \neg\psi)$

This formula also holds for fair quantifications $\mathsf{A}_f$ and $\mathsf{E}_f$.

Hence, we only need to compute the semantics of $\mathsf{E}_f\,\mathsf{G}\,\varphi$.

**Proof: Computation of $\mathsf{E}_f\,\mathsf{G}\,\varphi$**

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, $M, s \models \mathsf{E}_f\,\mathsf{G}\,\varphi$ iff $M, s \models \mathsf{E}\,\varphi\,\mathsf{U}\,S'$ iff $M_\varphi, s \models \mathsf{EF}\,S'$.

This is again a reachability problem which can be solved in linear time.