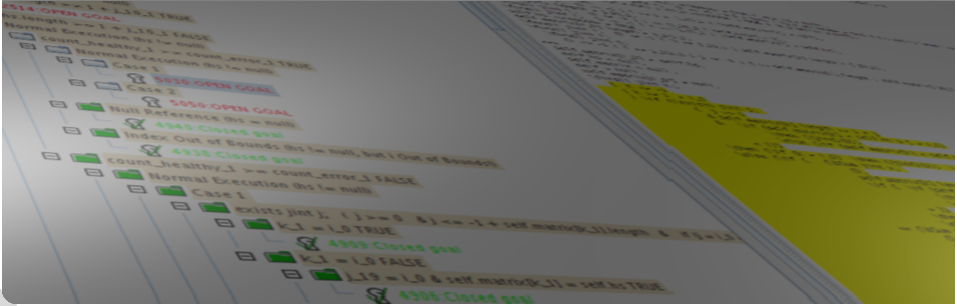
Karlsruhe Institute of Technology

Formale Systeme 2

Prof. Dr. Peter H. Schmitt

KIT – INSTITUT FÜR THEORETISCHE INFORMATIK

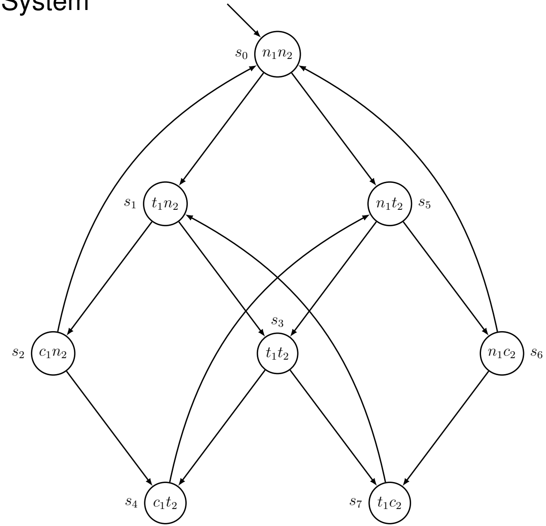


CTL

Computation Tree Logic

Motivating Example

Transition System



Motivating Example

Transition System

The Transitionsystem $\mathcal{T} = (S, R, v)$ uses propositional variables $n_1, n_2, t_1, t_2, c_1, c_2$ with the intended meaning.

$s \models n_i$	iff	in state s agent i is not trying
$s \models t_i$	iff	in state s agent i is trying
$s \models c_i$	iff	in state s agent i is in the critical section

Motivating Example

Properties

safety
: There is no state s reachable from s_0 with $s \models c_1 \wedge c_2$.

liveness
: Whenever an agent tries to enter the critical section it will eventually enter it.

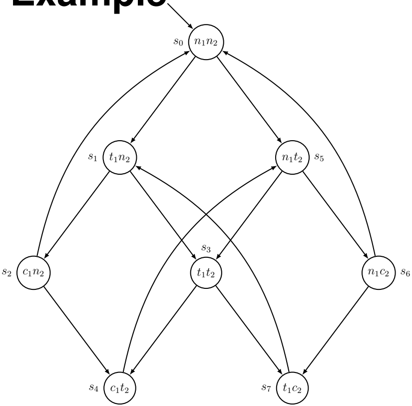
non-blocking
: An agent can always try to enter the critical section.

non-sequencing
: It is not the case that the agent who first tried will first enter the critical section.

non-alternating
: It is not the case that the two agents take alternate turns to the critical section.

Motivating Example

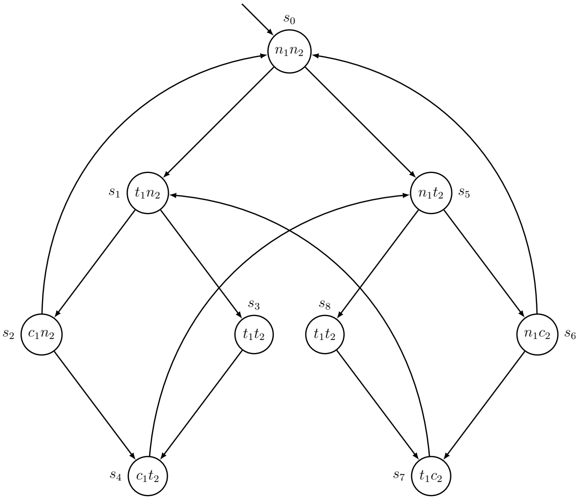
Properties



The safety property is obviously true.
There is not even a state s with $s \models c_1 \wedge c_2$

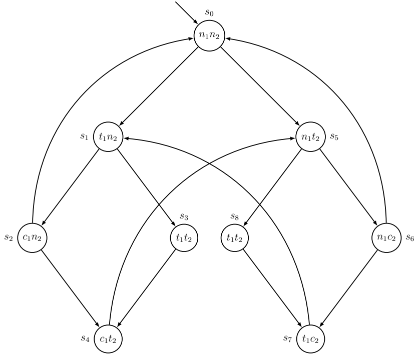
The non-blocking property can easily seen to be true.
Likewise the absence of dead ends

Modified Transition System



Modified Transition System

Properties



The liveness property is now true.

But now the non-sequencing property is violated.

Transition Systems

Definition

Let PVar be a set of propositional atoms.
A transition system $\mathcal{T} = (S, R, v)$ consists of

- a finite set S of states with one distinguished initial state s_0,
- a binary relation R and
- a function $v : S \times \text{PVar} \to \{\mathbf{1}, \mathbf{0}\}$

such that for every $s \in S$ there is $s' \in S$ with $R(s, s')$.

From a technical point of view a transition system is just a Kripke structure, whose accessability relation has no dead ends.

Computation Tree Logic (CTL)

Syntax

1. Any propositional variable $p \in \mathrm{PVar}$ is a CTL formula.
2. If F, G are CTL formulas then all propositional combinations are also CTL formulas, e.g., $\neg F$, $F \vee G$, $F \wedge G$, etc.
3. If F, G are CTL formulas then also

   $$\mathbf{AX}F, \mathbf{EX}F, \mathbf{A}(F\ \mathbf{U}\ G) \text{ and } \mathbf{E}(F\ \mathbf{U}\ G)$$

   are CTL formulas.

Note: The temporal operators **A**, **E** and **X**, **U** always occur in pairs.

Path

Let (S, R, v) be a transition system.

A path through (S, R, v) is an infinite sequence of states

$$t_1, t_2, \ldots, t_n, t_{n+1}, \ldots$$

such that t_1 is the initial state and for all n the relation $R(t_n, t_{n+1})$ is true.

CTL
Semantics

Let $\mathcal{T} = (S, R, v)$ be a transition system.
$(\mathcal{T}, s) \models \phi$,
read: formula ϕ is true in state s of $\mathcal{T}$,
will be abbreviated as $s \models \phi$.

1	$g \models p$	iff	$v(g, p) = 1$ (in case $p \in$ PVar)
2	$g \models \neg\phi$	iff	$g \not\models \phi$
3	$g \models \phi_1 \wedge \phi_2$	iff	$g \models \phi_1$ and $g \models \phi_2$
4	$g \models \mathbf{AX}\phi$	iff	$g_1 \models \phi$ is true for all g_1 with $R(g, g_1)$
5	$g \models \mathbf{EX}\phi$	iff	$g_1 \models \phi$ is true for at least one g_1 with $R(g, g_1)$

CTL

Semantics (continued)

6 $g \models \mathbf{A}(\phi_1\ \mathbf{U}\ \phi_2)$ iff for every path $g_0, g_1, \ldots$ with $g_0 = g$
there exists $i \geq 0,$ such that
$g_i \models \phi_2$ and
$g_j \models \phi_1$ for all j with $0 \leq j < i$,

7 $g \models \mathbf{E}(\phi_1\ \mathbf{U}\ \phi_2)$ iff there is a path $g_0, g_1, \ldots$ with $g_0 = g$
and there is $i \geq 0,$ such that
$g_i \models \phi_2$ and
$g_j \models \phi_1$ for all j satisfying $0 \leq j < i$,

Defined CTL Operators

Using **F** and **G** from LTL four new CTL operators can be defined:

$ua(\phi)$	$\equiv$	$\mathbf{AF}\phi$	$\equiv$	$\mathbf{A}(\mathbf{1}\ \mathbf{U}\ \phi)$	ϕ cannot be avoided
$re(\phi)$	$\equiv$	$\mathbf{EF}\phi$	$\equiv$	$\mathbf{E}(\mathbf{1}\ \mathbf{U}\ \phi)$	ϕ is reachable
$ofa(\phi)$	$\equiv$	$\mathbf{EG}\phi$	$\equiv$	$\neg\mathbf{A}(\mathbf{1}\ \mathbf{U}\ \neg\phi)$	once and for all ϕ
$aw(\phi)$	$\equiv$	$\mathbf{AG}\phi$	$\equiv$	$\neg\mathbf{E}(\mathbf{1}\ \mathbf{U}\ \neg\phi)$	always ϕ

8	$g \models \mathbf{AF}\phi$	iff	for every path $g_0, g_1, \ldots$ with $g_0 = g$ there exists $i \geq 0,$ such that $g_i \models \phi$
9	$g \models \mathbf{EF}\phi$	iff	there is a path $g_0, g_1, \ldots$ with $g_0 = g$ and there exists $i \geq 0,$ such that $g_i \models \phi$
10	$g \models \mathbf{EG}\phi$	iff	there is a path $g_0, g_1, \ldots$ with $g_0 = g$ such that $g_i \models \phi$ for all i
11	$g \models \mathbf{AG}\phi$	iff	for every path $g_0, g_1, \ldots$ with $g_0 = g$ and every i it is true that $g_i \models \phi$

CTL Tautologies

The following formulas are CTL tautologies:

1. $\textbf{AG}\ \phi \leftrightarrow \phi \land \textbf{AXAG}\ \phi$
2. $\textbf{EG}\ \phi \leftrightarrow \phi \land \textbf{EXEG}\ \phi$
3. $\textbf{AF}\ \phi \leftrightarrow \phi \lor \textbf{AXAF}\ \phi$
4. $\textbf{EF}\ \phi \leftrightarrow \phi \lor \textbf{EXEF}\ \phi$
5. $\textbf{A}(\phi\ \textbf{U}\ \psi) \leftrightarrow \psi \lor (\phi \land \textbf{AXA}(\phi\ \textbf{U}\ \psi))$
6. $\textbf{E}(\phi\ \textbf{U}\ \psi) \leftrightarrow \psi \lor (\phi \land \textbf{EXE}(\phi\ \textbf{U}\ \psi))$

CTL^*

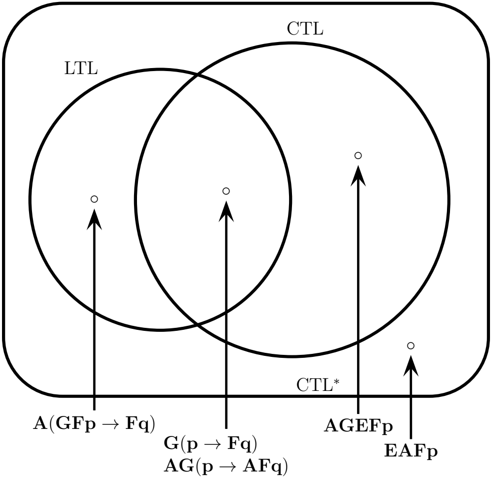
CTL* Formulas

There are two categories of CTL* formulas

- state formulas and
- path formulas.

1. any propositional variable is a state formula
2. if F, G are state formulas, so are $\neg F$, $F \vee G$, $F \wedge G$, etc.,
3. if F is a path formula, then $(\mathbf{A}F)$, $(\mathbf{E}F)$ are state formulas,
4. every state formula also is a path formula,
5. if F, G are path formulas, so are $\neg F$, $F \vee G$, $F \wedge G$,
6. if F, G are path formulas, so $\mathbf{X}F$ und $F\ \mathbf{U}\ G$.

Comparative Expressive Power



Comparing CTL* with LTL

Lemma

Let F be a CTL* state formula.

Then F is expressible in LTL iff F is equivalent to $\mathbf{A}(F^d)$.

F^d denotes the formula that arises from F by simply dropping all quantifiers.
Thus e.g., $(AFAGp)^d = FGp$.

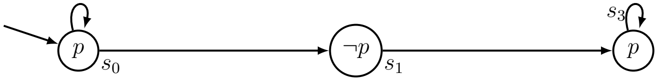
Proof: E.M.Clarke and I.A.Draghicescu, 1988

Comparing CTL with LTL

Application of previous Lemma

The formula $\phi = \mathbf{AFAG}p$ is in CTL but not in LTL.

$\phi^d = \mathbf{FG}p$



Set of all paths starting in s_0 is $\{s_0^n s_1 s_3^\omega \mid n \geq 1\} \cup \{s_0^\omega\}$.

$$s_0 \models \mathbf{AFG}p \quad \text{but} \quad s_0 \not\models \mathbf{AFAG}p.$$

Example reconsidered

Properties

safety: There is no state s reachable from s_0 with $s \models c_1 \wedge c_2$.
$s_1 \models \mathbf{AG}\neg(c_1 \wedge c_2)$

liveness: Whenever an agent tries it will eventually enter the CS.
$s_1 \models \mathbf{AG}(t_i \rightarrow \mathbf{A}(t_i \ \mathbf{U} \ c_i))$

non-blocking: An agent can always try to enter the critical section. $s_1 \models \mathbf{AG}(\neg(c_i \vee t_i) \rightarrow \mathbf{AX} t_i)$

non-sequencing: It is not the case that the agent who first tried will first enter the critical section.
$s_1 \models \neg\mathbf{AG}(t_1 \rightarrow \mathbf{A}((t_1 \wedge \neg c_2) \ \mathbf{U} \ c_1))$

non-alternating: It is not the case that the two agents take alternate turns to the critical section.
$s_1 \models \neg\mathbf{AG}(c_1 \rightarrow \mathbf{A}((\neg c_1) \ \mathbf{U}_w \ c_2))$