Linear Temporal Logic (propositional)

Amir Pnueli (1970ies Late)
$\Diamond$ "eventually"
$\Box$ "always"

relative order of eve[nts]
¬precise timi[ng]
time abstrac[tion]

Syntax

redundant if $AP \neq \emptyset$

$$\varphi ::= \text{true} \mid \underset{\in AP}{a} \mid \varphi \wedge \varphi \mid \neg \varphi \quad \text{constants and logical connectives}$$

$For_{LTL}(A)$

$$\mid \underset{\text{next}}{\bigcirc \varphi} \mid \underset{\text{until}}{\varphi \,\mathsf{U}\, \varphi} \quad \text{temporal modalities}$$

right associative

definable: false, $\rightarrow$, $\leftrightarrow$, $\oplus$ – exclusive or / parity

false ::= ¬ true

$\varphi_1 \rightarrow \varphi_2 ::= \neg \varphi_1 \vee \varphi_2$

$\varphi_1 \leftrightarrow \varphi_2 ::= (\varphi_1 \rightarrow \varphi_2) \wedge (\varphi_2 \rightarrow \varphi_1)$
$= (\neg \varphi_1 \vee \varphi_2) \wedge (\neg \varphi_2 \vee \varphi_1)$

$\varphi_1 \oplus \varphi_2 ::= (\varphi_1 \wedge \neg \varphi_2) \vee (\neg \varphi_1 \wedge \varphi_2)$

$\Diamond$, $\Box$
eventually, always

precedence: unary $\neg, \bigcirc$; binary U; $\wedge, \vee, \rightarrow$

$$\{\wedge, \vee, \rightarrow\} < \{\mathsf{U}\} < \{\neg, \bigcirc\}$$

$\neg \varphi_1 \,\mathsf{U} \bigcirc \varphi_2$ $\quad (\neg\varphi_1)\,\mathsf{U}\,(\bigcirc\varphi_2)$

$\varphi_1 \,\mathsf{U}\, \varphi_2 \,\mathsf{U}\, \varphi_3$ $\quad \varphi_1 \,\mathsf{U}\, (\varphi_2 \,\mathsf{U}\, \varphi_3)$

Intuitive semantics

An LTL formula expresses a property of an infinite path $\sigma \in 2^{AP}$
(i.e. models of LTL formulas are infinite sequences of sets $\subseteq 2^{AP}$)

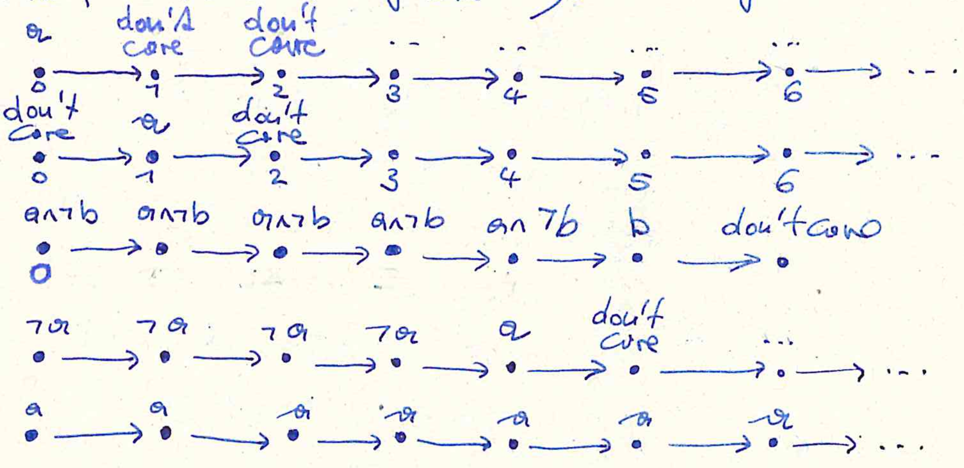
"a is true"

"$\bigcirc a$ is true"
next

"$a \,\mathsf{U}\, b$ is true"
until

"$\Diamond a$ is true"
eventually

"$\Box a$ is true"
always



Formal semantics

Words

Let $\sigma \in (2^{AP})^\omega$ and $\sigma = A_0 \ldots A_i A_{i+1}$ then $\sigma_{\geq i} = A_i A_{i+1} \ldots$
$\sigma[i] = A_i$

$\sigma \models_{LTL} \varphi$ ("σ models φ") if the statement "$\sigma \models \varphi$" follows from the following clauses:

$\sigma \models \text{true}$

$\sigma \models a$ iff $a \in \sigma[0]$

$\sigma \models \varphi_1 \wedge \varphi_2$ iff $\sigma \models \varphi_1$ and $\sigma \models \varphi_2$

$\sigma \models \neg \varphi$ iff $\sigma \not\models \varphi$ (that is, $\sigma \models \varphi$ does not hold)

$\sigma \models \bigcirc \varphi$ iff $\sigma_{\geq 1} \models \varphi$

$\sigma \models \varphi \,\mathsf{U}\, \psi$ iff $\exists i \geq 0: \sigma_{\geq i} \models \psi$ and $\forall 0 \leq j < i: \sigma_{\geq j} \models \varphi$

$\sigma \models \Diamond \varphi$ iff $\exists i \geq 0: \sigma_{\geq i} \models \varphi$

$\sigma \models \Box \varphi$ iff $\forall i \geq 0: \sigma_{\geq i} \models \varphi$

$Words ::= (2^{AP})^\omega$

$Words(\varphi) ::= \{\sigma \in Words \mid \sigma \models \varphi\}$

$\boxed{\text{Derived Modalities}}$

"eventually" $\quad \Diamond\varphi ::= true \,\mathsf{U}\, \varphi$

$\sigma \models true \,\mathsf{U}\, \varphi$ iff $\exists i \geq 0 : \sigma_{\geq i} \models \varphi$ and $\forall 0 \leq j < i : \sigma_{\geq j} \models true$

iff $\exists i \geq 0 : \sigma_{\geq i} \models \varphi$

iff $\sigma \models \Diamond\varphi$

"always" $\quad \Box\varphi ::= \neg\Diamond\neg\varphi$

$= \neg(true \,\mathsf{U}\, \neg\varphi)$

$\sigma \models \neg(true \,\mathsf{U}\, \neg\varphi)$ iff $\sigma \not\models true \,\mathsf{U}\, \neg\varphi$

iff $\neg\exists i \geq 0 : \sigma_{\geq i} \models \neg\varphi$ and $\forall 0 \leq j < i : \sigma_{\geq j} \models true$

iff $\neg\exists i \geq 0 : \sigma_{\geq i} \not\models \varphi$

iff $\forall i \geq 0 : \sigma_{\geq i} \models \varphi$

iff $\sigma \models \Box\varphi$

$\sigma \models \neg\Diamond\neg\varphi$ iff $\sigma \not\models \Diamond\neg\varphi$

iff not$(\sigma \models \Diamond\neg\varphi)$

iff not $(\exists i \geq 0 : \sigma_{\geq i} \models \neg\varphi)$

iff not $(\exists i \geq 0 : \sigma_{\geq i} \not\models \varphi)$

iff not $(\exists i \geq 0 :$ not $\sigma_{\geq i} \models \varphi)$

iff $\forall i \geq 0 : \sigma_{\geq i} \models \varphi$

iff $\sigma \models \Box\varphi$

<u>Exercise</u>: define $\Box\varphi$ see above

<u>Exercise</u>: define "infinitely often φ" $\quad \Box\Diamond\varphi$

"eventually forever φ" $\quad \Diamond\Box\varphi$

$\sigma \models \Box\Diamond\varphi$ iff $\forall i \geq 0 : \sigma_{\geq i} \models \Diamond\varphi$

iff $\forall i \geq 0 : \exists j \geq i : \sigma_{\geq j} \models \varphi$ iff $\overset{\infty}{\exists} j : \sigma_{\geq j} \models$

$\sigma \models \Diamond\Box\varphi$ iff $\exists i \geq 0 : \sigma_{\geq i} \models \Box\varphi$

iff $\exists i \geq 0 : \forall j \geq i : \sigma_{\geq j} \models \varphi$ iff $\overset{\infty}{\forall} j : \sigma_{\geq j} \models$

Exercises: Which of the following equivalences are correct?

$\boxed{5}$ a) $\Box(\varphi \to \Diamond\psi) \equiv \varphi \, U \, (\psi \wedge \neg\varphi)$ $(\times)$

counter model: let $\psi := \varphi := p$ and $\sigma_{\geq i} \not\models p$ for all $i \in \mathbb{N}$

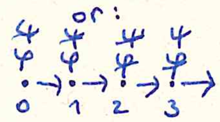
then $\sigma_{\geq i} \models \varphi \to \Diamond\psi$

and hence $\sigma \models \Box(\varphi \to \Diamond\psi)$

but $\sigma \not\models \varphi \, U \, (\psi \wedge \neg\varphi)$

because $\sigma_{\geq i} \not\models \psi \wedge \neg\varphi$ due to $\sigma_{\geq i} \not\models \psi$

or:



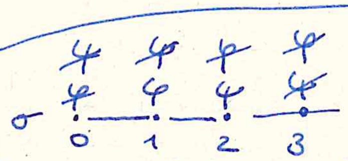
$\Box(\varphi \to \Diamond\psi)$

~~$\varphi U (\psi \wedge \neg\psi)$~~

$\boxed{3}$ b) $\bigcirc\Diamond\varphi \equiv \Diamond\bigcirc\varphi$ $(\checkmark)$

$\boxed{4}$ c) $\Box(\varphi \wedge \bigcirc\Diamond\varphi) \equiv \Box\varphi$ $(\checkmark)$

2 d) $\Diamond(\varphi \wedge \psi) \equiv \Diamond\varphi \wedge \Diamond\psi$ $(\times)$



$\sigma \models \Diamond\varphi, \ \sigma \models \Diamond\psi$

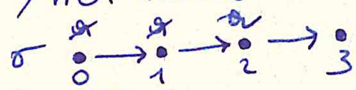
$\sigma \not\models \Diamond(\varphi \wedge \psi)$

7 e) $\Box(\varphi \wedge \psi) \equiv \Box\varphi \wedge \Box\psi$ $(\checkmark)$

$\boxed{6}$ f) $\Box\Box(\varphi \to \psi) \equiv \neg\Diamond(\neg\psi \wedge \varphi)$ $(\checkmark)$

Ex. 5.24: Check the following LTL-formulas whether they are (i) satisfiable, and/or (ii) valid.

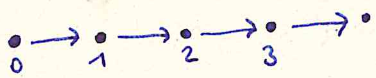
(a) $\bigcirc\bigcirc a \to \bigcirc a$ satisfiable / not valid



$\bigcirc\bigcirc a$

$\bigcirc a$

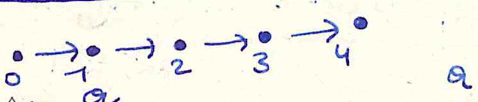
(b) $\bigcirc(a \vee \Diamond a) \to \Diamond a$ valid



$\bigcirc(a \vee \Diamond a)$ $a \vee \Diamond a$

$\Diamond a$

Case 1:



Case 2:

(c) $\Box a \to \neg\bigcirc(\neg a \wedge \Box\neg a)$ (valid)

(d) $(\Box a) \, U \, (\Diamond b) \to \Box(a \, U \, \Diamond b)$ (satisfiable, but not valid)

(e) $\Diamond b \to a \, U \, b$ (satisfiable, but not valid)

$\varphi_1 \equiv \varphi_2$ equivalent $:\Leftrightarrow$ $\text{Words}(\varphi_1) = \text{Words}(\varphi_2)$

φ satisfiable $:\Leftrightarrow$ $\text{Words}(\varphi) \neq \emptyset$

φ valid $:\Leftrightarrow$ $\text{Words}(\varphi) = (2^{AP})^\omega$

Lemma. $\varphi_1 \equiv \varphi_2 \iff \varphi_1 \leftrightarrow \varphi_2$ valid

Recall $TS = \langle S, Act, \rightarrow, I, AP, L \rangle$ transition system.

states, actions, $\rightarrow \subseteq S \times Act \times S$ transition relation, initial states, atomic propositions, $L: S \rightarrow 2^{AP}$ labeling function

$\boxed{\text{we may assume all paths and traces are infinite}}$

$\pi \in Paths(TS)$: infinite path segments $\quad \pi \models \varphi :\Leftrightarrow trace(\pi) \models \varphi$

$s \in S$: $\quad s \models \varphi :\Leftrightarrow \forall \pi \in Paths(s) : \pi \models \varphi$

$TS \models \varphi :\Leftrightarrow \forall s \in I : s \models \varphi$

---

examples

A note on negation:

$Words(\varphi) = \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \varphi \}$

$Words(\neg\varphi) = (2^{AP})^\omega \setminus Words(\varphi)$ due to $\sigma \models \varphi \Leftrightarrow \sigma \not\models \neg\varphi$

$\sigma \not\models \varphi \Leftrightarrow \sigma \models \neg\varphi$

$\boxed{\text{traces decide formulas}}$

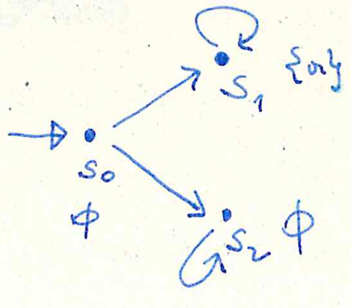
(an LTL formula is either true or false for a trace)

However: transition systems do not decide formulas

$TS \models \varphi \;\overset{\Longrightarrow}{\not\Leftarrow}\; TS \not\models \neg\varphi$

$TS \not\models \varphi \;\underset{\Longleftarrow}{\not\Rightarrow}\; TS \models \neg\varphi$

Take TS



$TS \not\models \Diamond a$ because $s_0 s_2 s_2 \ldots \not\models \Diamond a$

$TS \not\models \neg\Diamond a$ because $s_0 s_1 s_1 \ldots \not\models \neg\Diamond a$ due to $s_0 s_1 s_1 \ldots \models \Diamond a$

Exercise. Let $AP = \{a,b\}$ and $TS =$
$$\begin{tikzcd} \arrow[d] & & \\ s_1 \arrow[r, bend left] & s_2 \arrow[l, bend left] \arrow[r] & s_3 \arrow[loop above] \end{tikzcd}$$
$\{a,b\}$ $\{a,b\}$ $\{a\}$

a) Is TS deterministic? ~~no~~, neither action-deterministic nor AP-determi

b) $s_1 \models \bigcirc(a \wedge b)$? (✓)

c) $s_2 \models \bigcirc(a \wedge b)$? (×) because $s_2 s_3^\omega \not\models \bigcirc b$ and hence $s_2 s_3^\omega \not\models \bigcirc(a$

d) $TS \models \Box a$? (✓)

e) $TS \models \Box(\neg b \rightarrow a)$? (✓)

f) $TS \models \Diamond \neg b$? (×) since $s_1 s_2 s_1 s_2 \ldots \not\models$

$TS \models \Box b$ (×) since $s_1 s_2 s_3^\omega \not\models \Box b$

g) $TS \models \Diamond \Box \neg b$? (×)

$\text{Words}(\varphi) = \{ \sigma \in (2^{AP})^\omega \mid \sigma \models \varphi \}$

π infinite path fragment of TS

$\pi \models \varphi :\Leftrightarrow \text{trace}(\pi) \models \varphi$

$\Leftrightarrow \text{trace}(\pi) \in \text{Words}(\varphi)$

For all states we can define

$s \models \varphi :\Leftrightarrow \forall \pi \in \text{Paths}(s).\ \pi \models \varphi$ where $s \in S$

And finally

$TS \models \varphi :\Leftrightarrow \forall s \in I : s \models \varphi$

Input: φ LTL formula & TS finite without terminal states.

Output: "yes" if $TS \models \varphi$, otherwise counterexample

$\mathcal{A}_{\neg\varphi} :=$ NBA s.th. $\mathcal{L}(\mathcal{A}_{\neg\varphi}) = \text{Words}(\neg\varphi)$ Thm. Büchi automata accept ω-reg. languages

state explosion $\mathcal{A} := TS \otimes \mathcal{A}_{\neg\varphi}$ emptiness-checking

if $\exists \pi \in \text{paths}(\mathcal{A}) : \pi$ satisfies the acceptance condition of $\mathcal{A}$

then return (expressive) bad prefix of π

else return "yes"

fi

NBA $\mathcal{A}$ has empty language

$\Updownarrow$

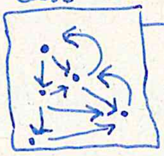
$\nexists$ reachable accepting state on a cycle

... is recognisable in linear time

Model checking in LTL

Basic Algorithm (Vardi, Wolper 1986)

MODELLING Loch TS
informal
or
formal



φ · $\Box\Diamond c \vee \Diamond \neg d$
· $a \cup \Diamond b$
...

spec

conjunction of several requirements

MC

$\times$ + counterexample $TS \not\models \varphi$, $\pi \not\models \varphi$

✓ $TS \models \varphi$

$TS \models \varphi \iff Traces(TS) \subseteq Words(\varphi)$
$\iff Traces(TS) \cap ((2^{AP})^{\omega} \setminus Words(\varphi)) = \emptyset$
$\iff Traces(TS) \cap Words(\neg\varphi) = \emptyset$
$\iff Traces(TS) \cap \mathcal{L}_{\omega}(\mathcal{A}_{\neg\varphi})$
$\iff \mathcal{L}_{\omega}(TS \otimes \mathcal{A}_{\neg\varphi}) = \emptyset$
$\iff TS \otimes \mathcal{A}_{\neg\varphi} \models \Diamond\Box\neg F$

System → Model of system → Transition system TS

Negation of property → LTL-formula $\neg\varphi$ → Generalized Büchi automaton $\mathcal{G}_{\neg\varphi}$ → Büchi automaton $\mathcal{A}_{\neg\varphi}$

Product transition system $TS \otimes \mathcal{A}_{\neg\varphi}$

$TS \otimes \mathcal{A}_{\neg\varphi} \models \underbrace{\Diamond\Box\neg F}_{Pers}$

eventually always not in a finite state

finite states

"Yes"   "No" (counterexample)

(persistence of forever no accept state)

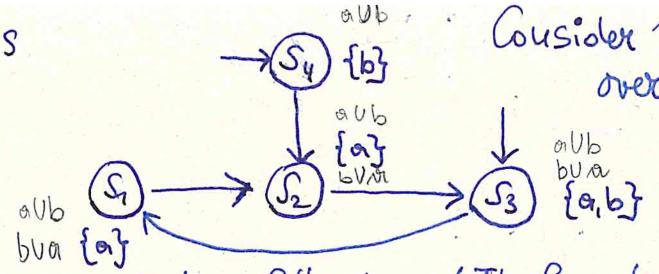
Check whether there is a reachable final state on a cycle of $TS \otimes \mathcal{A}_{\neg\varphi}$

(can be checked in linear time)

Complexity $O(|TS| \cdot 2^{|\varphi|})$ PSPACE-Complete

5.5 Exercises

Exercise 5.1.



Consider the following transition system over set $\{a,b\}$ of atomic propositions:

$J = \langle \{s_1, s_2, s_3, s_4\}, 2^{\{a,b\}}, \rightarrow, \{s_3, s_4\}, \{a,b\}, L\rangle$

Indicate for each of the following LTL-formulae the set of states for which these formulae are fulfilled:

(a) $\bigcirc a$ $\{s_1, s_2, s_3, s_4\} = \{s \in S / s \models_J \bigcirc a\}$

path: $\pi \models \varphi :\Leftrightarrow \text{trace}(\pi) \models \varphi$

$s \models \varphi :\Leftrightarrow \forall \pi \in \text{Paths}(s).\ \pi \models \varphi$

$TS \models \varphi :\Leftrightarrow (\forall \pi \in \text{Paths}(s_i), s_i \text{ initial state of } TS)\ \pi \models \varphi$

$\forall s_{i_{init}} \in TS, s_i \text{ initial state}: s_{init} \models \varphi$

$\Leftrightarrow \text{Traces}(TS) \subseteq \text{Words}(\varphi)$

$\text{Words}(\varphi) := \{\sigma \in (2^{AP})^\omega / \sigma \models \varphi\}$

$\text{Traces}(TS) := \{\text{trace}(\pi) / \pi \in \text{Path}_{TS}(s_{init}), s_{init} \in I\}$

$\pi = s_1 s_2 s_3 \cdots$

$\text{trace}(\pi) = L(s_1)L(s_2)L(s_3)\cdots$

$TS = \langle S, Act, \rightarrow, I, AP, L\rangle$

$\rightarrow \subseteq S \times Act \times S$ transition relation

$I \subseteq S$ initial states

AP atomic propositions

$L: S \rightarrow 2^{AP}$

(b) $\bigcirc\bigcirc\bigcirc a$ $\{s \in S \mid s \models \bigcirc\bigcirc\bigcirc a\} = \{s_1, s_2, s_3, s_4\}$

(c) $\square b$ $\{s \in S \mid s \models \square b\} = \{\}$

(d) $\square \lozenge a$ $\{s \in S \mid s \models \square \lozenge a\} = S$

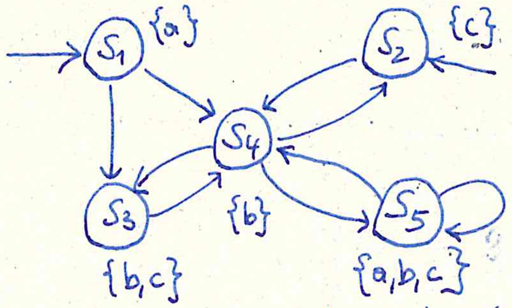
(e) $\square (b \,U\, a)$ $\{s \in S / s \models \square (b \,U\, a)\} = S$

(f) $\lozenge (a \,U\, b)$ $\{s \in S / s \models \lozenge (a \,U\, b)\} = S$

(g) $\square a$ $\{s \in S / s \models \square a\} = \{s_1, s_2, s_3\}$

Exercise 5.2

Consider the transition system over the set of atomic propositions $AP = \{a, b, c\}$: (8)



Decide for each LTL formulae φ_i below, whether $TS \models \varphi_i$ holds. Justify your answers. If $TS \not\models \varphi_i$ provide a path $\pi \in Paths(TS)$ such that $\pi \not\models \varphi_i$:

$\varphi_1 = \Diamond\Box c$ $\quad TS \not\models \varphi_1 \quad s_1 s_3 s_4 s_3 s_4 \ldots \not\models \Diamond\Box c$

$\varphi_2 = \Box\Diamond c$ $\quad TS \models \varphi_2$

$\varphi_3 = \bigcirc\neg c \to \bigcirc\bigcirc c$ $\quad TS \models \varphi_3$

$\varphi_4 = \Box a$ $\quad TS \not\models \varphi_4 \quad s_2 s_4 \ldots \not\models a$

hence $s_2 s_4 \ldots \not\models \Box a$

$\varphi_5 = a \, U \, \Box(b \vee c)$ $\quad TS \models \varphi_5$

$$\left.\begin{array}{l} s_2, \ldots, s_5 \models \Box b \vee c \\ s_1 \models a \end{array}\right\} \Rightarrow \begin{array}{l} s_2 \models a \, U \, (b \vee c) \\ s_1 \models a \, U \, (b \vee c) \\ \Rightarrow TS \models \varphi_5 \end{array}$$

$\varphi_6 = (\bigcirc\bigcirc b) \, U \, (b \vee c)$ $\quad TS \not\models \varphi_6$ because $s_1 s_4 s_2 \ldots \not\models \bigcirc\bigcirc b$, $\not\models b \vee c$

$\varphi_1' = \Diamond\Box b$ $\quad TS \not\models \varphi_1'$

$\varphi_2' = \Box\Diamond b$ $\quad TS \models \varphi_2'$

$TS = \langle S, Act, \to, I, AP, L \rangle$

$\pi \in Path(TS)$: $\quad \pi \models \varphi \;:\Leftrightarrow\; trace(\pi) \models \varphi$

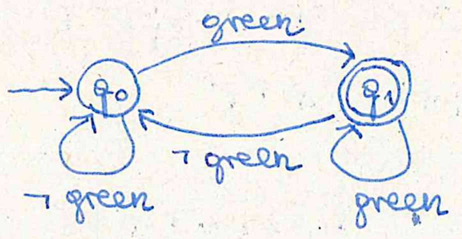
$s \in S$: $\quad s \models \varphi \;:\Leftrightarrow\; \forall \pi \in Path(s) : \pi \models \varphi$

$TS \models \varphi \;:\Leftrightarrow\; \forall s \in I. \; s \models \varphi$

$\sigma \models \varphi \, U \, \psi \;\Leftrightarrow\; \exists j \geq 0: \sigma_j \models \psi$ and $\forall 0 \leq i < j: \sigma \models \varphi$

non-deterministic Büchi automaton

NBA for $\square\lozenge$ green
infinitely often green

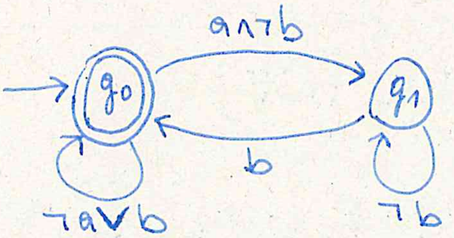


deterministic

{green} ∅ {green} ∅ ...
q_0 q_1 q_0 q_1

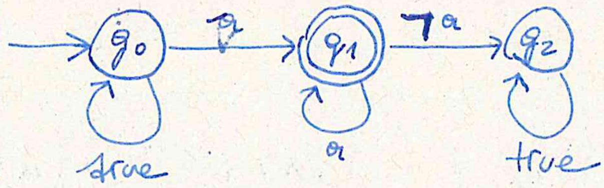
infinitely often visits end state q_1
⇓
is accepting computation.

NBA for $\square(a \to \lozenge b)$

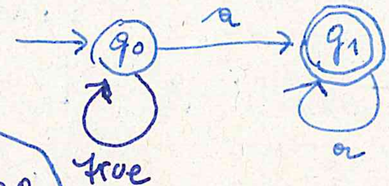


almost always a

NBAs for $\lozenge\square a$



Complete NBA



incomplete NBA $Q \times A \to \mathcal{P}(Q)$
(Q, A, δ, I, F)
$q_{i_0} q_{i_1} q_{i_2} \ldots$

$\sigma \in (2^{AP})^\omega$ an accepting run of Büchi automaton $\mathcal{A}$ if $\overset{\infty}{\exists} j \geq 0:$ $q_{i_j} \in F$

$\sigma = A_0 A_1 A_2 A_3 \ldots$

Büchi-automata accept ω-regular languages.

Julius Richard Büchi (1924–1984)

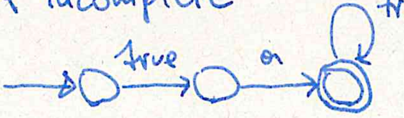
$G = E_1 . F_1^\omega + \ldots + E_n . F_n^\omega$
where $E_1, \ldots, E_n, F_1, \ldots, F_n$ are reg. expressi

$E, E_1, E_2 ::\ \emptyset \mid \hat{a} \mid E_1 + E_2 \mid E_1 . E_2 \mid E_1^*$

NBA for $\bigcirc a$

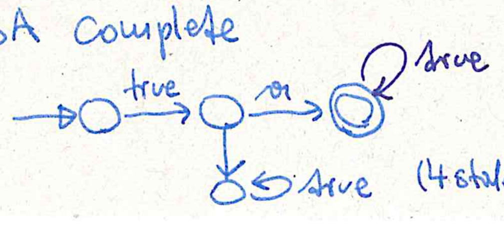
possible with 2 states?
(probably still yes with 2 initial states)

DBA incomplete



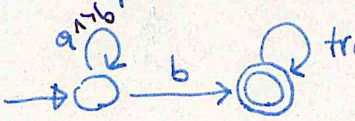
(3 states)

DBA complete



(4 states)

DBA complete
NBA for $a \mathsf{U} b$



(2 states)

Baier/Katoen, p285:
"For example, for the LTL formulae $\bigcirc a$ and $a \mathsf{U} b$ an NBA with 2 states suffices. (It is left to the reader to provide these NBAs.)"

Weak Until

$$\sigma \models \varphi W \psi \;:\Longleftrightarrow\; \left(\exists i \geq 0:\ \sigma_{\geq i} \models \psi \wedge \forall 0 \leq j < i.\ \sigma_{\geq j} \models \varphi\right)$$
$$\vee\ \forall i \geq 0:\ \sigma_{\geq i} \models \varphi \wedge \neg\psi$$

$$\neg(\varphi U \psi) \equiv \underbrace{(\varphi \wedge \neg\psi) U (\neg\varphi \wedge \neg\psi) \ \vee\ \Box(\varphi \wedge \neg\psi)}_{(\varphi \wedge \neg\psi) W (\neg\varphi \wedge \neg\psi)}$$

motivation.

Def. $\varphi W \psi := (\varphi U \psi) \vee \Box\varphi$

Then
$$\neg(\varphi U \psi) \equiv (\varphi \wedge \neg\psi) W (\neg\varphi \wedge \neg\psi)$$
$$\neg(\varphi W \psi) \equiv (\varphi \wedge \neg\psi) U (\neg\varphi \wedge \neg\psi)$$

Positive Normal Form for LTL (Weak-Until PNF)

$$\varphi ::= true/false \mid a \mid \neg a \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \varphi_1 U \varphi_2 \mid \varphi_1 W$$

Fairness in LTL

Definition. Let Φ and Ψ be propositional logic formulae over AP.

1. An unconditional LTL fairness constraint is an LTL-formula of the f[orm]

$$ufair = \square\lozenge\Psi$$

2. A strong LTL fairness condition:

$$sfair = \square\lozenge\Phi \to \square\lozenge\Psi.$$

3. A weak LTL-fairness condition:

$$wfair = \lozenge\square\Phi \to \square\lozenge\Psi.$$

"$s_0 \xrightarrow{\alpha_1} s_2 \xrightarrow{\alpha_2} \dots$"

ρ is unconditionally A-fair if $\overset{\infty}{\exists} j \geq 0 : \alpha_j \in A$

ρ is strongly A-fair if $\overset{\infty}{\exists} j \geq 0.\ A \cap Act(s_j) \neq \emptyset \Rightarrow \overset{\infty}{\exists} j \geq 0 : \alpha_j \in A$

weakly A-fair if $\overset{\infty}{\forall} j \geq 0 : A \cap Act(s_j) \neq \emptyset \Rightarrow \overset{\infty}{\exists} j \geq 0 : \alpha_j \in A$

An LTL fairness assumption is a conjunction of LTL fairness constraints of arbitrary type

$$sfair = \bigwedge_{0 < i \leq k} (\square\lozenge\Phi_i \to \square\lozenge\Psi_i)$$

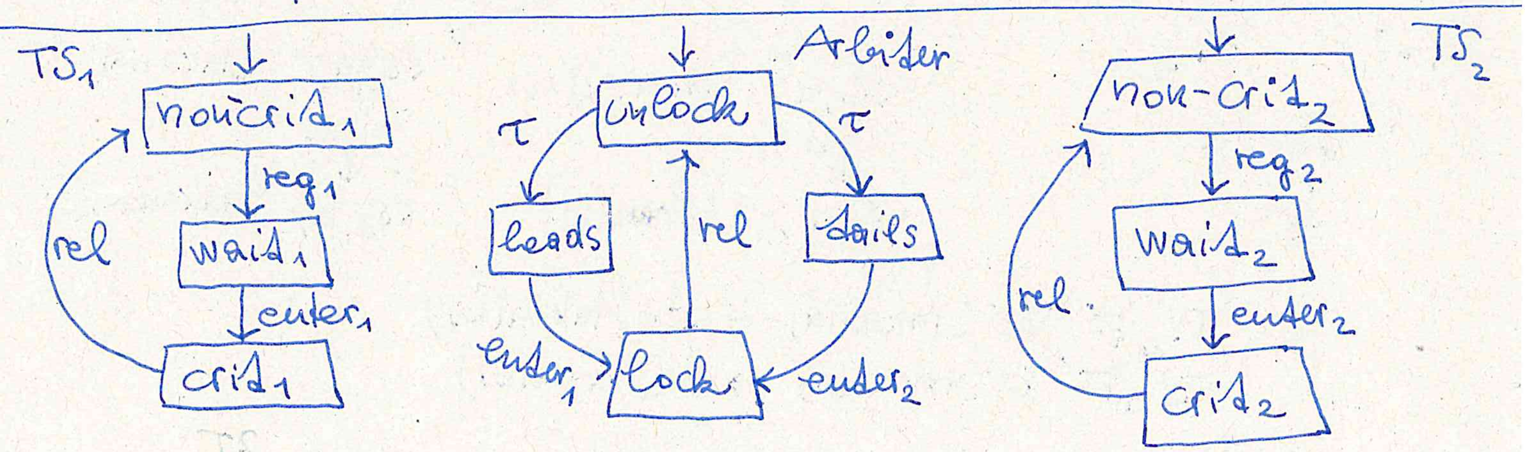
$$fair = ufair \wedge sfair \wedge wfair.$$

$$FairPaths(s) = \{\pi \in Paths(s) \mid \pi \models fair\}$$

$$FairTraces(s) = \{trace(\pi) \mid \pi \in FairPaths(s)\}$$

$$s \models_{fair} \varphi \ :\Leftrightarrow\ \forall \pi \in FairPaths(s) : \pi \models \varphi$$

$$TS \models_{fair} \varphi \ :\Leftrightarrow\ \forall s_0 \in I : s_0 \models_{fair} \varphi$$



$$TS_1 \parallel Arbiter \parallel TS_2 \not\models \square\lozenge crit_1$$

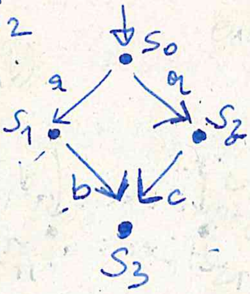
$$fair = \square\lozenge heads \wedge \square\lozenge tails$$

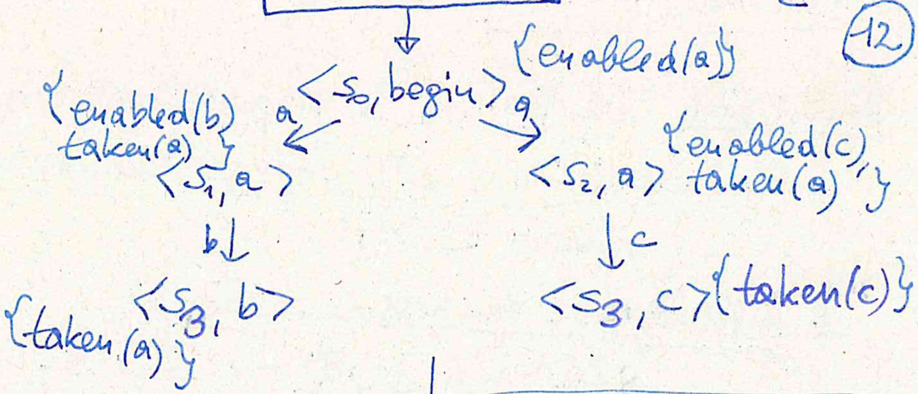
$$TS_1 \parallel Arbiter \parallel TS_2 \models_{fair} \square\lozenge crit_1 \wedge \square\lozenge crit_2$$

action-based versus state-based

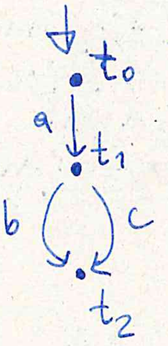
$\widetilde{TS}_2$

TS_2

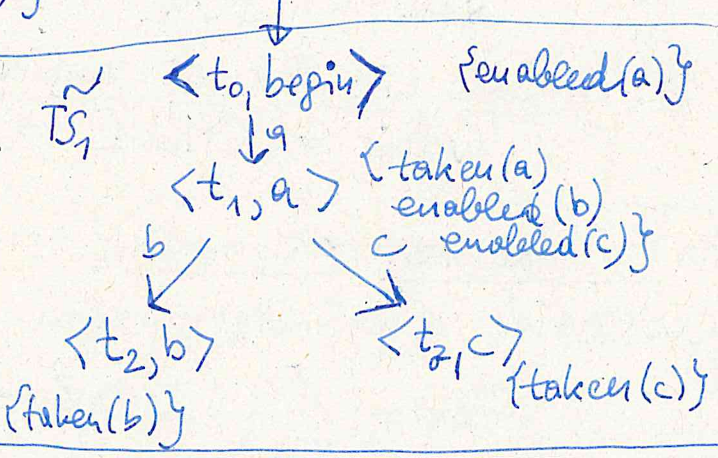




TS_1



$\widetilde{TS}_1$

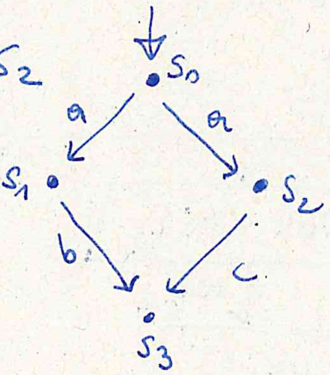


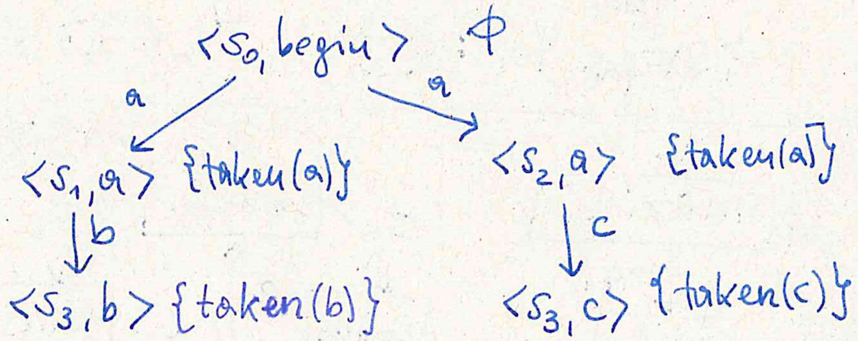
teaser for Lecture on CTL

$\varphi \equiv \forall \circ \text{ enabled}(b)$ $\qquad \varphi \equiv \exists \circ \text{ enabled}$

$\widetilde{TS}_1 \models \varphi \qquad \widetilde{TS}_2 \not\models \varphi$

TS_2

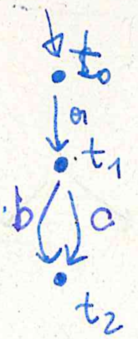


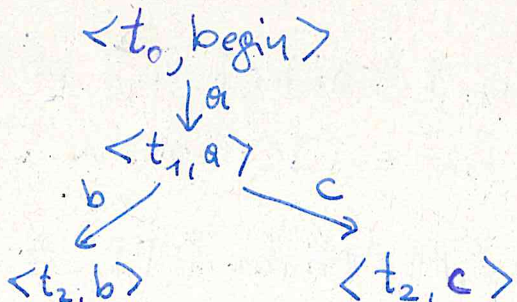


$TS_2 \not\models \forall\circ(\text{taken}(a) \rightarrow \exists\circ(\text{taken}(b))$

$TS_2 \not\models \circ(\text{taken}(a) \rightarrow \circ(\text{taken}(b))$

TS_1





$TS_1 \models \forall\circ(\text{taken}(a) \longrightarrow \exists\circ(\text{taken}(b))$

$TS_1 \not\models \circ(\text{taken}(a) \longrightarrow \circ(\text{taken}(b))$