

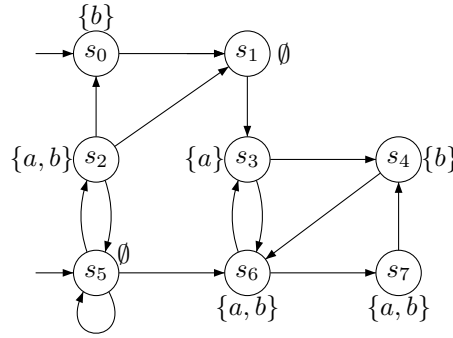
Exercise 1 (CTL-star):

(3 points)

Consider the CTL*-formula (over $AP = \{a, b\}$)

$$\Phi = \forall \Diamond \Box \exists \bigcirc (a \cup \exists \Box b)$$

and the transition system TS outlined below:



Apply the CTL* Model Checking Algorithm to compute $Sat(\Phi)$ and decide whether $TS \models \Phi$.

Hint: You may infer the satisfaction sets for LTL formulas directly.

Solution:

We consider the maximal proper state subformulas $Sub(\Phi)$:

1. $\Psi = a$: $Sat(a) = \{s_2, s_3, s_6, s_7\}$
2. $\Psi = b$: $Sat(b) = \{s_0, s_2, s_4, s_6, s_7\}$
3. $\Psi = \exists \Box b$:

The following equivalence is used to compute $Sat(\exists \Box b)$:

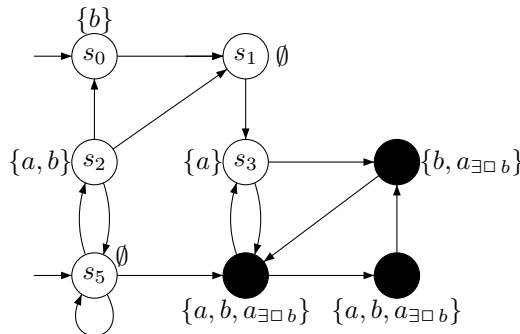
$$s \models_{CTL^*} \exists \varphi \iff s \models_{CTL^*} \neg \forall \neg \varphi \iff s \not\models_{CTL^*} \forall \neg \varphi \iff s \not\models_{LTL} \neg \varphi$$

According to the LTL semantics, we have $Sat_{LTL}(\neg \Box b) = Sat_{LTL}(\Diamond \neg b) = \{s_0, s_1, s_2, s_3, s_5\}$. Then, $S \setminus Sat_{LTL}(\neg \Box b) = \{s_4, s_6, s_7\}$ is the satisfaction set $Sat_{CTL^*}(\exists \Box b)$:

$$Sat_{CTL^*}(\exists \Box b) = \{s_4, s_6, s_7\}.$$

The labelling is extended by a fresh atomic proposition $a_{\exists \Box b}$ according to $Sat_{CTL^*}(\exists \Box b)$.

The corresponding subformula $\exists \Box b$ of Φ is replaced by $a_{\exists \Box b}$.



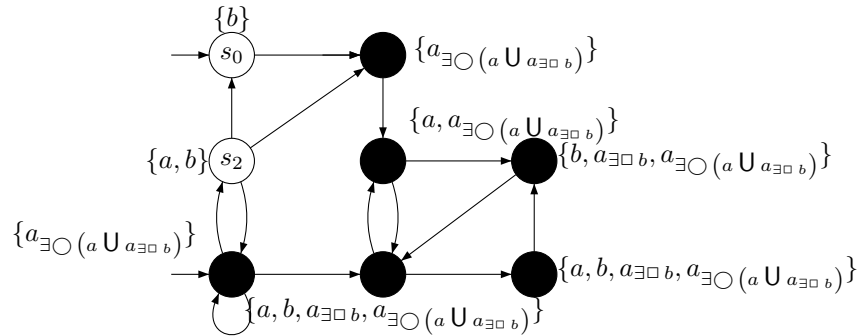
The above equivalence for existentially quantified path formulas yields:

$$s \models_{\text{CTL}^*} \exists \bigcirc (a U_{a \exists \square b}) \iff s \not\models_{\text{LTL}} \neg \bigcirc (a U_{a \exists \square b}).$$

By the equivalence $\neg \bigcirc (a U_{a \exists \square b}) \equiv \bigcirc \neg (a U_{a \exists \square b})$, the satisfaction set of $\neg (a U_{a \exists \square b})$ can be inferred:

$$\begin{aligned} \text{Sat}_{\text{LTL}}(\neg (a U_{a \exists \square b})) &= \{s_0, s_1, s_2, s_5\} \\ \text{Sat}_{\text{LTL}}(\bigcirc \neg (a U_{a \exists \square b})) &= \{s_0, s_2\} \\ \text{Sat}_{\text{CTL}^*}(\exists \bigcirc (a U_{a \exists \square b})) &= S \setminus \text{Sat}_{\text{LTL}}(\bigcirc \neg (a U_{a \exists \square b})) \\ &= S \setminus \{s_0, s_2\} \\ &= \{s_1, s_3, s_4, s_5, s_6, s_7\} \end{aligned}$$

The labelling is extended by a new atomic prop. $a_{\exists \bigcirc (a U_{a \exists \square b})}$ according to $\text{Sat}_{\text{CTL}^*}(\exists \bigcirc (a U_{a \exists \square b}))$. Again, the corresponding subformula Ψ of Φ is replaced by $a_{\exists \bigcirc (a U_{a \exists \square b})}$:



5. $\Psi = \forall \Diamond \square a_{\exists \bigcirc (a U_{a \exists \square b})}$:

In the case of universal quantification, we can directly apply the LTL-semantics:

$$\text{Sat}_{\text{LTL}}(\Diamond \square a_{\exists \bigcirc (a U_{a \exists \square b})}) = \{s_0, s_1, s_3, s_4, s_6, s_7\}.$$

Because of $s_5 \in Q_0$, but $s_5 \notin \text{Sat}(\Phi)$, this yields $TS \not\models_{\text{CTL}^*} \Phi$.

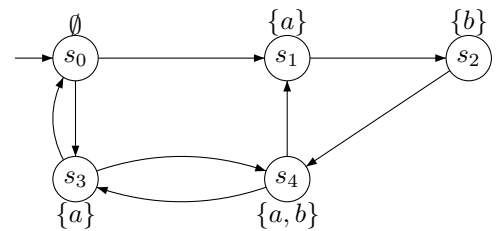
Exercise 2 (Fair-CTL Model Checking):

(3 points)

Consider the CTL-formula $\Phi = \forall \square (a \rightarrow \forall \Diamond (b \wedge \neg a))$ together with the following CTL fairness assumption

$$\begin{aligned} \text{fair} &= \square \Diamond \forall \bigcirc (a \wedge \neg b) \rightarrow \square \Diamond \forall \bigcirc (b \wedge \neg a) \\ &\quad \wedge \Diamond \square \exists \Diamond b \rightarrow \square \Diamond b. \end{aligned}$$

Check that $TS \models_{\text{fair}} \Phi$!



Solution:

First we check the state formulae of the fairness assumptions. We have:

- $\text{Sat}(\forall \bigcirc (a \wedge \neg b)) = \{s_0, s_4\}$
- $\text{Sat}(\forall \bigcirc (b \wedge \neg a)) = \{s_1\}$

- $Sat(\exists \Diamond b) = S$
- $Sat(b) = \{s_2, s_4\}$

Next we need to check which are the states where the fairness assumption is realisable, that is $s \models_{fair} \exists \Box true$. To show realisability it suffices to find a *strongly connected node set*¹ that is reachable from s and satisfies both constraints of *fair*. Let us consider the whole state space S which is a strongly connected node set (and an SCC). We check that S satisfies the first (i.e. strong) condition like in the lecture:

Treatment of $\exists \Box$ under strong fairness

CTLFAIR4.4-20A

$$fair = \bigwedge_{1 \leq i \leq k} (\Box \Diamond b_i \rightarrow \Box \Diamond c_i)$$

$s \models_{fair} \exists \Box a$ iff there exists a non-trivial
strongly connected node-set D of G_a such that

- (1) D is reachable from s
- (2) for all $1 \leq i \leq k$:
 $D \cap Sat(b_i) = \emptyset$ or $D \cap Sat(c_i) \neq \emptyset$

G_a : digraph that arises from $\mathcal{T}$ by removing all
states s' with $s' \not\models a$

152 / 132

We have that $S \cap Sat(\forall \Box (b \wedge \neg a)) \neq \emptyset$ so the first constrain is satisfied. Analogously we have to check for the second constraint that

$$S \not\subseteq Sat(\exists \Diamond b) \quad \text{or} \quad S \cap Sat(b) \neq \emptyset .$$

The latter is true, so the second (i.e. weak) constraint is satisfied as well. Obviously S is reachable from every state in S , hence *fair* is realisable from all states and we can label all states with a_{fair} .

We can now apply the model checking algorithm assuming that our formula is in existential normal form.

$$\begin{aligned} \Phi &= \forall \Box (a \rightarrow \forall \Diamond (b \wedge \neg a)) \\ &\equiv \neg \exists \Diamond \neg (a \rightarrow \forall \Diamond (b \wedge \neg a)) \\ &\equiv \neg \exists \Diamond \neg (\neg a \vee \forall \Diamond (b \wedge \neg a)) \\ &\equiv \neg \exists \Diamond (a \wedge \neg \forall \Diamond (b \wedge \neg a)) \\ &\equiv \neg \exists \Diamond (a \wedge \exists \Box (\neg b \vee a)) \\ &= \Phi' \end{aligned}$$

- $Sat_{fair}(a) = Sat(a) = \{s_1, s_3, s_4\}$
- $Sat_{fair}(b) = Sat(b) = \{s_2, s_4\}$
- $Sat_{fair}(\neg b) = S \setminus Sat_{fair}(b) = \{s_0, s_1, s_3\}$
- $Sat_{fair}(\neg b \vee a) = Sat(\neg b \vee a) = \{s_0, s_1, s_3, s_4\}$

¹Mind the difference to an SCC which has to be maximal, i.e. an SCC cannot be a subgraph of a bigger SCC.

- $Sat_{fair}(\exists\Box(\neg b \vee a)) \dots$

Calculation of $Sat_{fair}(\exists\Box a)$

CTLFair4.4-27

compute the SCCs of the digraph G_a ;

$T := \emptyset$;

FOR ALL nontrivial SCCs C of G_a DO

IF CheckFair($C, \dots$) THEN $T := T \cup C$ FI
OD

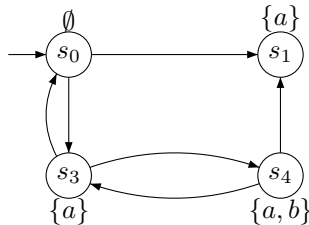
$Sat_{fair}(\exists\Box a) := \{s \in S : Reach_{G_a}(s) \cap T \neq \emptyset\}$

backward search from T

time complexity: $\mathcal{O}(\text{size}(T) \cdot |fair|)$

205 / 132

$G_{\neg b \vee a}$ looks as:



It has only one SCC $C = \{s_0, s_3, s_4\}$ for which *CheckFair* returns *false* as **no** state satisfies $\forall \bigcirc (b \wedge \neg a)$ but it impossible to (infinitely often) avoid states that satisfy $\forall \bigcirc (a \wedge \neg b)$.

Recursive algorithm *CheckFair(...)*

CTLFAIR4.4-28

pseudo code for *CheckFair*($C, k, \bigwedge_{1 \leq i \leq k} (\Box \Diamond b_i \rightarrow \Box \Diamond c_i)$)

```

IF  $\forall i \in \{1, \dots, k\}. C \cap \text{Sat}(c_i) \neq \emptyset$  THEN return "true" FI
choose  $j \in \{1, \dots, k\}$  with  $C \cap \text{Sat}(c_j) = \emptyset$ ;
remove all states in  $\text{Sat}(b_j)$ ;
IF the resulting graph  $G$  is acyclic THEN return "false" FI
FOR ALL nontrivial SCCs  $D$  of  $G$  DO
  IF CheckFair( $D, k-1, \bigwedge_{i \neq j} (\Box \Diamond b_i \rightarrow \Box \Diamond c_i)$ )
  THEN return "true" FI
OD
return "false"

```

215 / 132

Hence $\text{Sat}_{\text{fair}}(\exists \Box (\neg b \vee a)) = \emptyset$

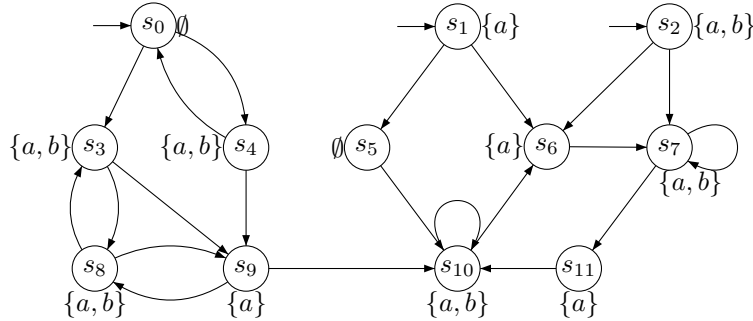
- $\text{Sat}_{\text{fair}}(a \wedge \exists \Box (\neg b \vee a)) = \text{Sat}(a \wedge \exists \Box (\neg b \vee a) \wedge a_{\text{fair}}) = \emptyset$
- $\text{Sat}_{\text{fair}}(\exists \Diamond (a \wedge \exists \Box (\neg b \vee a))) = \text{Sat}(\exists \Diamond (a \wedge \exists \Box (\neg b \vee a) \wedge a_{\text{fair}})) = \emptyset$
- $\text{Sat}_{\text{fair}}(\Phi') = S$

We have that $\text{Sat}_{\text{fair}}(\Phi) = \text{Sat}_{\text{fair}}(\Phi') = S \supseteq I$ and therefore $TS \models_{\text{fair}} \Phi$!

Exercise 3 (Bisimulation I):

(2 points)

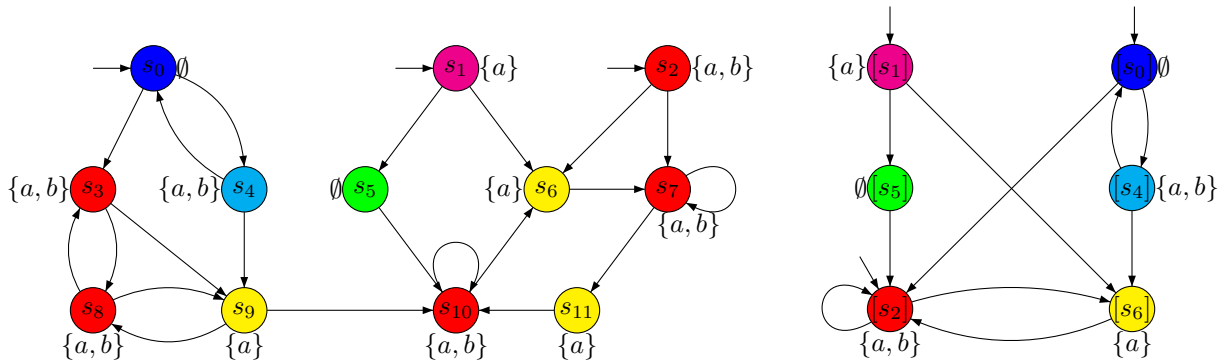
Consider the transition system TS over $AP = \{a, b\}$ outlined below:



1. Determine the bisimulation equivalence $\sim_{TS}$ and depict the bisimulation quotient system $TS/\sim$.
2. For each bisimulation equivalence class C , provide a CTL formula Φ_C that holds only in the states in C .

Solution: _____

1. The bisimulation equivalence classes are as follows:



2. Corresponding master formulas are given below:

$$\Phi_{[s_0]} = \neg a \wedge \neg b \wedge \exists \bigcirc \exists \bigcirc \neg a$$

$$\Phi_{[s_1]} = a \wedge \neg b \wedge \exists \bigcirc (\neg a \wedge \neg b)$$

$$\Phi_{[s_4]} = a \wedge b \wedge \exists \bigcirc (\neg a \wedge \neg b)$$

$$\Phi_{[s_5]} = \neg a \wedge \neg b \wedge (\forall \bigcirc \forall \bigcirc a)$$

$$\Phi_{[s_6]} = a \wedge \neg b \wedge \forall \bigcirc (a \wedge b)$$

$$\Phi_{[s_2]} = a \wedge b \wedge \forall \bigcirc a$$

Exercise 4 (Bisimulation II):

(2 points)

Consider the following three transition systems TS_1 , TS_2 , and TS_3 , where all the states labelled with $\{a\}$ are initial states. Decide whether $TS_i \sim TS_j$ for $i, j \in \{1, 2, 3\}$ and $i \neq j$. If $TS_i \not\sim TS_j$, then provide a distinguishing CTL formula Φ such that $TS_i \models \Phi \iff TS_j \not\models \Phi$.

Solution: _____

- $TS_1 \not\sim TS_2$.
 $TS_1 \models \exists \bigcirc b$ and $TS_2 \not\models \exists \bigcirc b$.

- $TS_1 \not\sim TS_3$.
 $TS_1 \models \exists \bigcirc b$ and $TS_3 \not\models \exists \bigcirc b$.

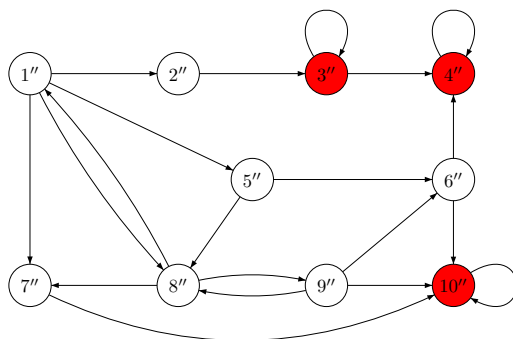
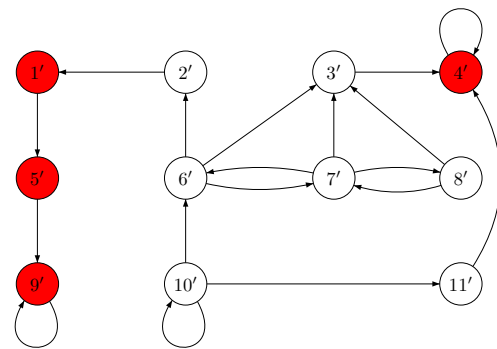
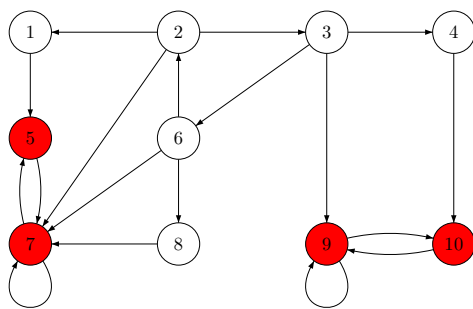
- $TS_2 \not\sim TS_3$.

$$9'' \models \overbrace{\exists \bigcirc b \wedge \exists \bigcirc a}^{\Phi}$$

$$TS_3 \models \overbrace{\underbrace{\forall \bigcirc b}_{\text{for } 2'', 6'', 7''} \vee \underbrace{\exists \bigcirc \Phi}_{\text{for } 8''} \vee \underbrace{\exists \bigcirc \exists \bigcirc \Phi}_{\text{for } 1'', 5''} \vee \underbrace{\Phi}_{\text{for } 9''}}^{\Phi'}$$

$$TS_2 \not\models \Phi'.$$

(Note that $TS \models \Phi$ iff $s \models \Phi$ for all $s \in I$.)

 $\bigcirc \{a\}$