

Exercise 1 (Weak Until):

(4 points)

Prove the following theorem: For any finite DTMC it holds that

$$\mathbb{P}_{=1}(\Diamond a) \equiv \forall((\exists \Diamond a) W a)$$

where W is the weak until operator defined by $\Phi W \Psi = (\Phi U \Psi) \vee \Box \Phi$.

Solution: _____

$$\begin{aligned} & Pr(s \models \Diamond a) < 1 \\ \Leftrightarrow & Pr(s \models \Box \neg a) > 0 \\ \Leftrightarrow & \exists \hat{\pi} \in Paths^*(s) \text{ such that } Cyl(\hat{\pi}) \cap \Diamond a = \emptyset \\ \stackrel{(1)}{\Leftrightarrow} & \exists \hat{\pi} \in Paths^*(s), \exists \neg a\text{-BSCC } B \text{ such that "}\hat{\pi} \text{ is a } \neg a\text{-path" and } s \xrightarrow{\hat{\pi}} B \\ \Leftrightarrow & \exists \pi \in Paths(s) \text{ such that } \underbrace{(\pi \models \Box \neg a \wedge \pi \models \Diamond \Box \neg a)}_{\pi \text{ is a } \neg a\text{-path that ends in a } \neg a\text{-BSCC}} \\ \Leftrightarrow & \exists \pi \in Paths(s) \text{ such that } (\pi \not\models \Diamond a \wedge \pi \not\models \Box \Diamond a) \\ \Leftrightarrow & s \models \exists(\neg(true U a) \wedge \neg \Box \Diamond a) \\ \Leftrightarrow & s \models \exists(\neg((\exists \Diamond a) U a) \wedge \neg \Box \Diamond a) \\ \Leftrightarrow & s \models \neg \forall((\exists \Diamond a) W a) \\ \Leftrightarrow & s \not\models \forall((\exists \Diamond a) W a) \end{aligned}$$

where (1) is due to the Long-run Theorem.

Exercise 2 (PCTL Equivalences):

(2 points)

Prove or disprove the following *PCTL* equivalences.

- a) $\mathbb{P}_{>0.5}(\bigcirc \mathbb{P}_{>0.5}(\Diamond a)) \equiv \mathbb{P}_{>0.5}(\Diamond \mathbb{P}_{>0.5}(\bigcirc a))$
- b) $\mathbb{P}_{=1}(\bigcirc \mathbb{P}_{=1}(\Diamond a)) \equiv \mathbb{P}_{=1}(\Diamond \mathbb{P}_{=1}(\bigcirc a))$

Solution: _____

Exercise 2

a) True. Consider:

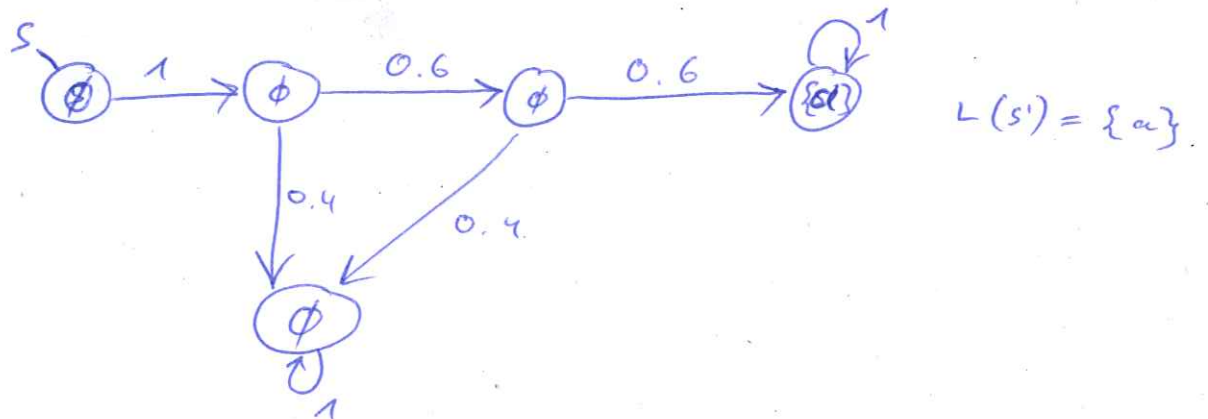
$$S \models IP_{=1}(\bigcirc IP_{=1}(\Box a))$$

$\Leftrightarrow$ "Pick ^{at random} any path $\sigma = s \cdot s_1 \cdot s_2 \dots$. Then almost ~~certainly~~ surely σ is of the form such that: $\sigma' = s_1 \cdot s_2 \dots$ visits a -states ad infinitum almost certainly"

$\Leftrightarrow$ "Pick at random any path $\sigma = s \cdot s_1 \cdot s_2 \dots$. Then almost surely the successor of every state along σ is almost surely an a -state"

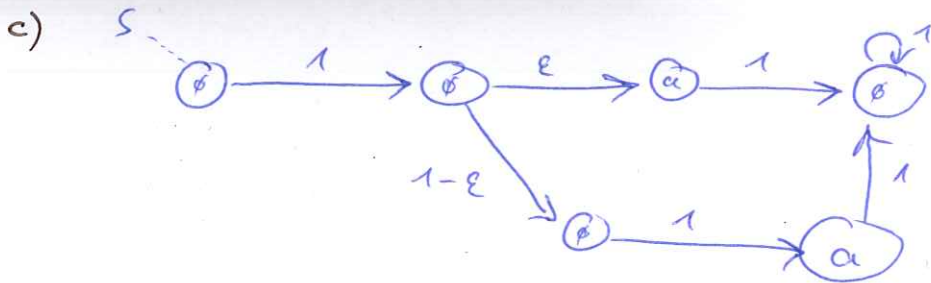
$$\Leftrightarrow S \models IP_{=1}(\Box IP_{=1}(\bigcirc a))$$

b) False. Counter example



$$S \not\models IP_{>0.5}(\bigcirc IP_{>0.5}(\Diamond a)) \quad , \quad \text{but}$$

$$S \models IP_{>0.5}(\Diamond IP_{>0.5}(\bigcirc a))$$



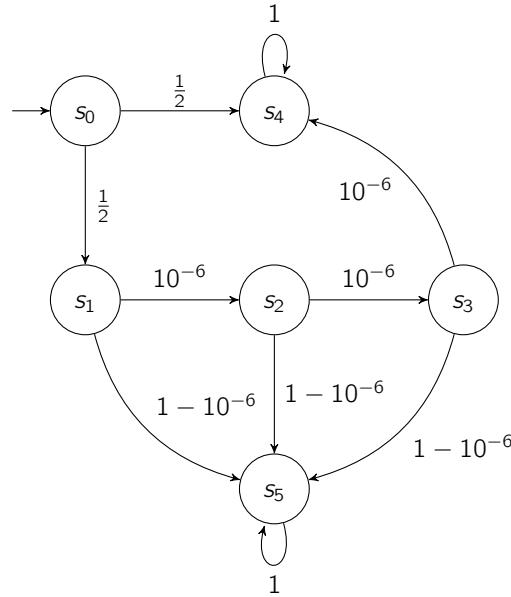
$$S \models \mathbb{P}_{=1}(\bigcirc \mathbb{P}_{=1}(\Diamond a)), \text{ but}$$

$$S \not\models \mathbb{P}_{=1}(\Diamond \mathbb{P}_{=1}(\bigcirc a))$$

Exercise 3 (PCTL Model Checking):

(4 points)

a) Consider the following DTMC D .



(i) Without any computation, decide whether $D \models \underbrace{\mathbb{P}_{\leq 0.5}(\Diamond s_4)}_{\Phi}$.

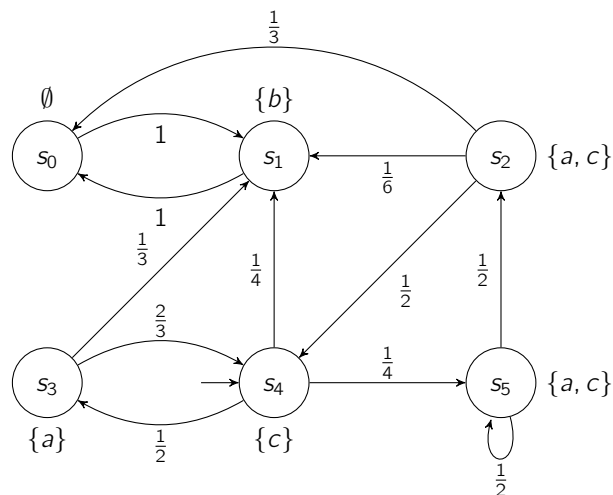
(ii) Write a PRISM model that precisely captures the above DTMC.

(iii) Use PRISM to verify or refute your result for Φ .

To document your work, please include your model and the (relevant) output of PRISM. Additionally, explain any remarkable observations.

Hint: PRISM and further documentation can be obtained on www.prismmodelchecker.org.

b) Consider the DTMC D' .



Determine the set $Sat(\Phi)$ where $\Phi = \mathbb{P}_{< \frac{3}{4}}(\bigcirc \mathbb{P}_{\geq \frac{1}{3}}(a \text{ U}^{\leq 3} (b \vee \neg c)))$ using the algorithm from the lecture. Please give intermediate results and explain your reasoning. A full computation, however, is not needed.

Solution: _____

- a)** (i) Obviously, the probability to reach s_4 is strictly larger than $\frac{1}{2}$.
(ii) A PRISM model that captures this DTMC is the following:

```
dtmc

module test
  s : [0..5] init 0;

  [] s=0 -> 1/2 : (s'=4) + 1/2 : (s'=1);
  [] s>0 & s<4 -> 0.000001 : (s'=s+1) + 0.999999 : (s'=5);
  [] s>=4 -> 1 : true;

endmodule

label "target" = s=4;
```

- (iii) PRISM (falsely) reports that the property is satisfied. This is because of two phenomena:

- “false” detection of convergence, because of a tolerance that is too high in this case,
- rounding errors in the arithmetic (using double precision floating point numbers).

Using the parametric engine of PRISM (switch “-param s”), one can use exact arithmetic and get the exact result.

- b)** Just like the algorithm, we compute the satisfaction sets bottom up over the syntax tree of the formula. We compute

- $Sat(b) = \{s_1\}$
- $Sat(c) = \{s_2, s_4, s_5\}$
- $Sat(b \vee \neg c) = Sat(b) \cup (S \setminus Sat(c)) = \{s_0, s_1, s_3\}$
- $Sat(a) = \{s_2, s_3, s_5\}$
- $Sat(\underbrace{\mathbb{P}_{\geq \frac{1}{3}}(a \text{ U}^{\leq 3} (b \vee \neg c))}_{\Phi'}) = S \setminus \{s_4\}$, because
 - $Pr(s \models \Phi') = 1$ for $s \in \{s_0, s_1, s_3\}$
 - $Pr(s_2 \models \Phi') = \frac{1}{2}$
 - $Pr(s_5 \models \Phi') = \frac{1}{4} + \frac{1}{8} = \frac{3}{8}$
 - $Pr(s_4 \models \Phi') = 0$
- $Sat(\Phi) = \{s_2, s_3\}$, because only s_2 and s_3 have a probability of more than $\frac{1}{4}$ of being in s_4 in the next step, which is the only state not satisfying the nested formula Φ' .