Homework 2

CS6501-006: Safety and Security in CPS

William Young

April 4, 2017

1. **Problem 1.**

   (a) *Compute the probability measure of the union of cylinder sets*

$$\begin{aligned}
&\Pr(Cyl(s_0s_1) \cup Cyl(s_0s_5s_6) \cup Cyl(s_0s_5s_4s_3) \cup Cyl(s_0s_1s_6)) \\
&\quad = P_{s0}(s_0s_1) + P_{s0}(s_0s_5s_6) + P_{s0}(s_0s_5s_4s_3) + P_{s0}(s_0s_1s_6) \\
&\quad = \left(\frac{1}{3}\right) + \left(\frac{2}{3} \cdot \frac{1}{4}\right) + \left(\frac{2}{3} \cdot \frac{1}{4} \cdot 1\right) + \left(\frac{1}{3} \cdot \frac{2}{3}\right) \\
&\quad = \frac{32}{36} = \frac{8}{9}
\end{aligned}$$

   (b) *Compute* $\text{ProbReach}^{\leq 4}(s_i, \{A\}) \forall i \in [0..6]$

$$\begin{aligned}
&\text{ProbReach}^{\leq 4}(s_0, \{A\}) = P_{s0}(s_0s_5s_4s_3) = \left(\frac{2}{3} \cdot \frac{1}{4} \cdot 1\right) = \frac{1}{6} \\
&\text{ProbReach}^{\leq 4}(s_1, \{A\}) = 0 \\
&\text{ProbReach}^{\leq 4}(s_2, \{A\}) = 0 \\
&\text{ProbReach}^{\leq 4}(s_3, \{A\}) = 1 \\
&\text{ProbReach}^{\leq 4}(s_4, \{A\}) = 1 \\
&\text{ProbReach}^{\leq 4}(s_5, \{A\}) = P_{s5}(s_5s_4s_3) + P_{s5}(s_5s_0s_5s_4s_3) = \frac{1}{4} + \left(\frac{1}{2} \cdot \frac{2}{3} \cdot \frac{1}{4} \cdot 1\right) = \frac{1}{3} \\
&\text{ProbReach}^{\leq 4}(s_6, \{A\}) = 0
\end{aligned}$$

(c) *Compute* $\text{ProbReach}(s_i, \{A\}) \forall i \in [0..6]$

$$\begin{aligned}
\text{ProbReach}(s_0, \{A\}) &= P_{s0}(s_0 s_5 s_4 s_3) + P_{s0}(s_0 s_5 s_0 s_5 s_4 s_3) + P_{s0}(s_0 s_5 s_0 s_5 s_0 s_5 s_4 s_3) + ... \\
&= \left[\left(\frac{1}{3}\right)^0 \cdot \frac{2}{3} \cdot \frac{1}{4}\right] + \left[\left(\frac{1}{3}\right)^1 \cdot \frac{2}{3} \cdot \frac{1}{4}\right] + \left[\left(\frac{1}{3}\right)^2 \cdot \frac{2}{3} \cdot \frac{1}{4}\right] + ... = \frac{1}{4} \\
\text{ProbReach}(s_1, \{A\}) &= 0 \\
\text{ProbReach}(s_2, \{A\}) &= 0 \\
\text{ProbReach}(s_3, \{A\}) &= 1 \\
\text{ProbReach}(s_4, \{A\}) &= 1 \\
\text{ProbReach}(s_5, \{A\}) &= P_{s5}(s_5 s_4 s_3) + P_{s5}(s_5 s_0 s_5 s_4 s_3) + P_{s5}(s_5 s_0 s_5 s_0 s_5 s_4 s_3) + ... \\
&= \left[\left(\frac{1}{3}\right)^0 \cdot \frac{1}{4}\right] + \left[\left(\frac{1}{3}\right)^1 \cdot \frac{1}{4}\right] + \left[\left(\frac{1}{3}\right)^2 \cdot \frac{1}{4}\right] + ... = \frac{3}{8} \\
\text{ProbReach}(s_6, \{A\}) &= 0
\end{aligned}$$

(d) *Compute the reachability of* $\{A \cup B\}$

We are asked to compute the probability that, from the initial state s_0, we will reach the set of states $\{A \cup B\}$. We can model this in PCTL and then verify it using the Prism model checker. The probability of reaching the set of states $\{A \cup B\}$ is 1.

$$\text{P}_{=?}\ [\text{F}\ (s = 2 | s = 3)] \rightarrow \text{P} = 1.0$$

Intuitively, this makes sense. From the fundamental property of DTMCs, we know that with probability 1, a bottom strongly connected component will be reached and all of its states visited. In this case, we have two BSCCs each encapsulating one of either state A (s_3) or state B (s_2):

$$\{s_3, s_4\} \in \text{BSCC } T_1 \text{ and } \{s_1, s_2, s_6\} \in \text{BSCC } T_2$$

Therefore, it is guaranteed that we will reach a state in ($\{A \cup B\}$), hence probability 1.

(e) *Compute the repeated reachability of* $\{A \cup B\}$

We are asked to compute the probability that, from the initial state s_0, we will reach the set of states $\{A \cup B\}$ infinitely often. In other words, we are computing the repeated reachability of $\{A \cup B\}$. Similar to the previous problem, we can model this problem in PCTL and verify it using the Prism model checker. The probability of of always eventually reaching the set of states $\{A \cup B\}$ is 1.

$$\text{P}_{=?}\ [\text{G F}\ (s = 2 | s = 3)] \rightarrow \text{P} = 1.0$$

Again, intuitively, this makes sense. In the last problem, we observed the fundamental property of DTMCs which states that with probability 1, a bottom

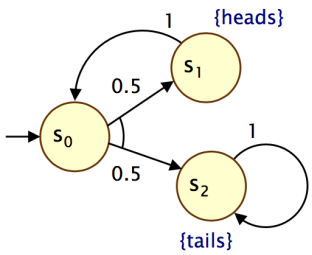

Figure 1: A simple finite DTMC

strongly connected component will be reached and all of its states visited infinitely often. Since A and B are encapsulated by both BSCCs, we observe that one of either state A or state B will be visited infinitely often, hence probability 1.

2. **Problem 2.**

   (a) *The formulae are equivalent.*
   We observe that $\mathrm{P}_{\geq 1}[\mathrm{G}\ a] = \mathrm{P}_{=0}[\mathrm{F}\neg a] = \neg\mathrm{P}_{>0}[\mathrm{F}\neg a]$.
   We already know $\mathrm{P}_{>0}[\mathrm{F}\ a] \equiv \mathrm{EF}\ a$, thus $\mathrm{P}_{>0}[\mathrm{F}\neg a] \equiv \mathrm{EF}\neg\ a$.
   Substituting, given an arbitrary state s:
   $s \models \neg\mathrm{P}_{>0}[\mathrm{F}\neg a] \iff s \not\models \mathrm{P}_{>0}[\mathrm{F}\neg a] \iff s \not\models \mathrm{EF}\neg a \iff s \models \neg\mathrm{EF}\neg\ a$
   Note that $\neg\mathrm{EF}\neg a = \mathrm{AG}\ a$.
   Therefore, $\mathrm{P}_{\geq 1}[\mathrm{G}\ a] \equiv \mathrm{AG}\ a$.

   (b) *The formulae are not equivalent.*
   Consider the simple finite DTMC from lecture (Figure 1).
   Let atomic proposition $a = \neg\texttt{tails}$.
   Consider the CTL formula EG a, meaning *for some path ω, a is always satisfied.*
   The infinite path $\omega = s_0s_1s_0s_1s_0s_1...$ satisfies EG $\neg\texttt{tails}$.
   Now consider the PCTL formula $\mathrm{P}_{>0}[\mathrm{G}\ a]$. We must show that the probability that a path ω exists in which $\neg\texttt{tails}$ is always satisfied is greater than zero. However, the infinite path $\omega = s_0s_1s_0s_1s_0s_1...$ has zero probability, thus no path exists with probability greater than zero in which $\neg\texttt{tails}$ is always satisfied.
   Therefore, EG $a \not\equiv \mathrm{P}_{>0}[\mathrm{G}\ a]$.

3. **Problem 3.**

   (a) The long-run probability of being in state s_2 can be computed as the unique solution of the linear equation system:

$$\vec{\pi} \cdot \mathbf{P} = \vec{\pi} \text{ and } \sum_{s \in S} \vec{\pi}(s) = 1$$

I used the Prism model checker to calculate the long-run probabilities. Note that I selected an arbitrary value for p since it has no effect on state s_2.

```
dtmc
const double p = 1;
module prob3
// local state
s : [0..5] init 0;
[] s=0 -> 0.5 : (s'=1) + 0.5 : (s'=3);
[] s=1 -> 1 : (s'=2);
[] s=2 -> 1 : (s'=1);
[] s=3 -> 1 : (s'=4);
[] s=4 -> p : (s'=4) + 1-p : (s'=5);
[] s=5 -> 1 : (s'=3);
endmodule
```

The resultant values from Prism (in matrix form):
$\begin{bmatrix} 0 & 0.25 & 0.25 & 0.16667 & 0.16667 & 0.16667 \end{bmatrix}$
As we see, the long run probability of being in state s_2 is 0.25. Intuitively, this makes sense—the chance of entering the BSCC $\{s_1, s_2\}$ is 0.5, and there is an equal chance of being in s_1 and s_2, hence $\frac{0.5}{2} = 0.25$.

(b) In order to calculate the value of p such that the long-run probability of being in states $\{s_2, s_4\}$ is above 0.6, I again utilized the Prism model checker's steady state compute feature.
Setting p to 0.78571429 (or $\frac{11}{14}$) resulted in the satisfaction of the steady state property $\mathrm{S}_{=0.6}\,[s = 2|s = 4]$. Therefore, when $p > \frac{11}{14}$, the steady state probability of being in state s_4 is greater than 0.35, meaning the long-run probability of being in states $\{s_2, s_4\}$ is greater than 0.6 (probability of s_2 remains unchanged at 0.25, independent of s_4).

4. **Problem 4.**

My answers for this problem are predicated on the assumption that, given a state s satisfying both a and b, the property a `W` b is also satisfied.

(a) *Deterministic Finite Automata*
See Figure 2. Note that state c refers to

(b) *DTMC and DFA Product*
See Figure 3.

(c) *Probability that the safety property holds*
To compute the probability that the safety property holds from the initial state of the DTMC in part (b), we model the DTMC using the Prism model checker.

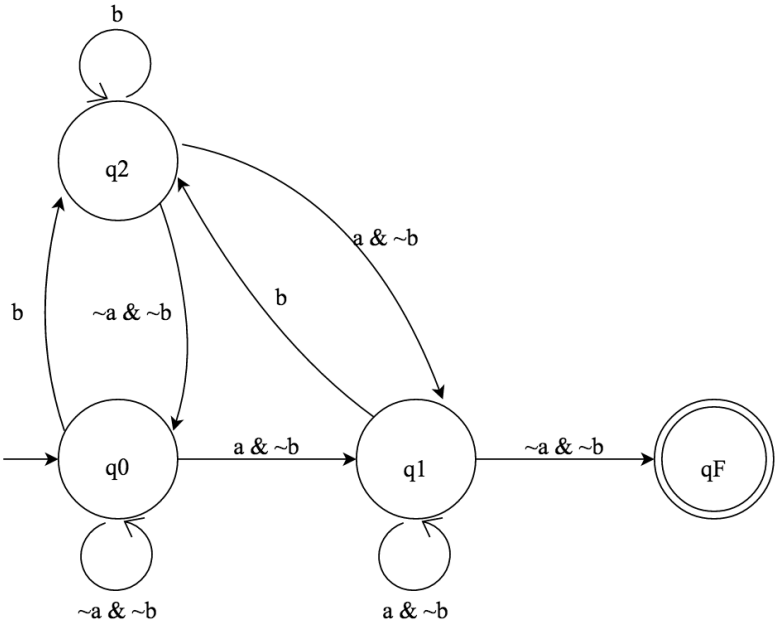

Figure 2: DFA for a `W` b

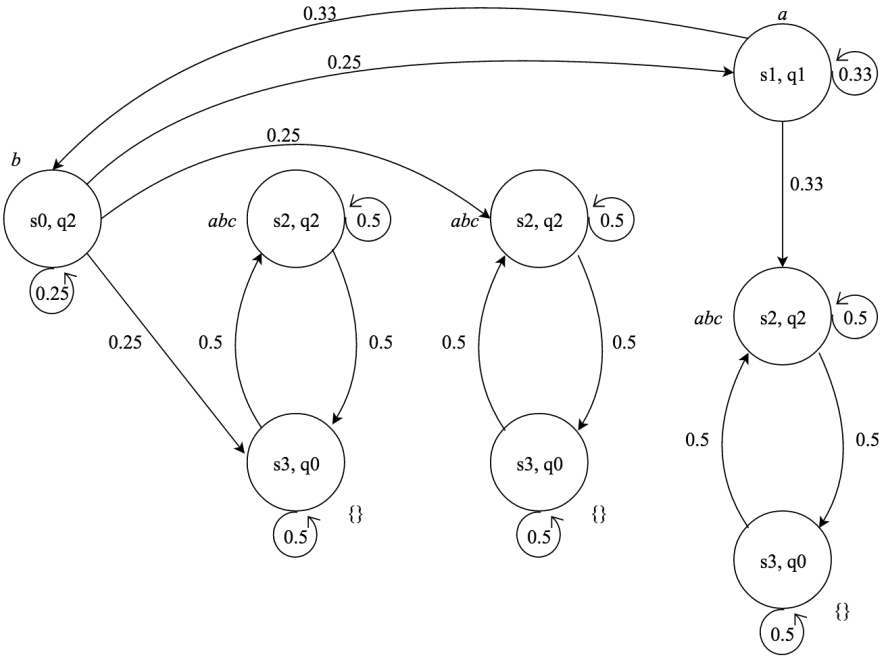

Figure 3: Product of DFA and DMTC

Note that we declare two labels, a and b. State s_3 (abc) dually satisfies a and b, since it does not violate the safety property `G` a `W` b (according to the definition of *until*, given `G` a `W` b, a must hold at least until b, which holds at the current or a future position—thus abc satisfies *until* because a and b hold at the current position, abc).

```
dtmc
module prob4
// local state
s : [0..3] init 0;
[] s=0 -> 1/4 : (s'=1) + 1/4 : (s'=3) + 1/4 : (s'=2) + 1/4 : (s'=0);
[] s=1 -> 1/3 : (s'=1) + 1/3 : (s'=0) + 1/3 : (s'=2);
[] s=2 -> 1/2 : (s'=2) + 1/2 : (s'=3);
[] s=3 -> 1/2 : (s'=3) + 1/2 : (s'=2);
endmodule
label "a" = s=1 & s=2;
label "b" = s=0 & s=2;
```

To compute whether the safety property holds, we use Prism to verify the property and obtain the following probability:

$$\mathrm{P}_{=?}\ [\texttt{G}\ a\ \texttt{W}\ b] \rightarrow \mathrm{P} = 1$$

In other words, the safety property almost surely holds with a probability of 1. Intuitively, this makes sense—for the safety property to fail, we must pass from the first encountered instance of a to $\emptyset$ without satisfying b. It is evident from the DTMC that no such transition exists, since all paths traversing from a to $\emptyset$ travel through one of either s_0 (b) or s_2 (abc). Thus, `G` a `W` b almost surely holds.