

Model Checking SS22
Assignment 10
Due: June 9nd, 2022, 16:00

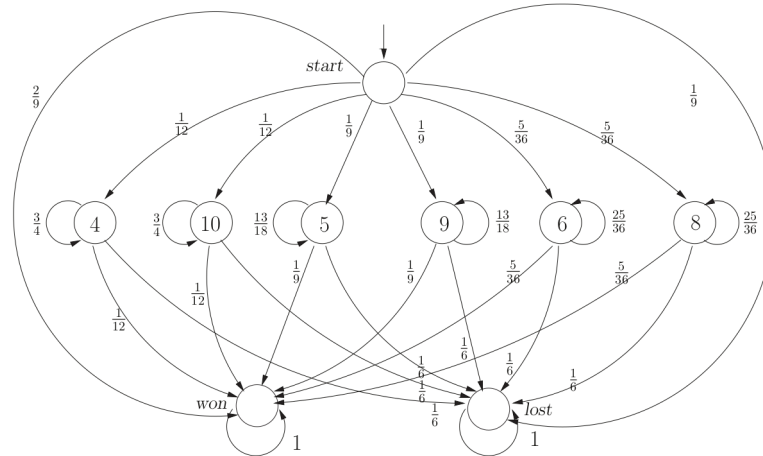


Figure 1: Fig 1: The model of the "Craps"-game

For this assignment we are going to compute the satisfaction sets for properties in PCTL and the given MC for the game "Craps".

The game is played as follows:

- The player throws two dice at once. If the dice show 7 or 11 the player wins, if the dice show 2, 3 or 12 the player loses.
- In any other case the outcome of the first throw is remembered and the player has to throw the two dice again until she throws either a 7 or the point.
 - If the dice now show 7 the player loses.
 - If the next roll shows the point the player wins.
 - In any other case she has to throw the dice again.

In the following we will use the points 4, 10, 5, 9, 6 and 8 as well as *won* and *lost* as labels. Further let s_0 be the state labelled with *start*. Compute whether the following properties hold:

1. (a) $\mathcal{M}, s_0 \models \mathbb{P}_{\geq \frac{2}{3}}(\mathbf{X} (won \vee lost))$

We would like to know whether a game ends after the first roll with a probability of $\geq \frac{2}{3}$. State the whole transition matrix and the vector $\mathbf{b}_{(won \vee lost)}$, as well as the result.

(b) $\mathcal{M}, s_0 \models \mathbb{P}_{\geq 0.32}(\neg(10 \vee 9 \vee 8) \mathbf{U}^{\leq 5} won)$

(c) $\mathcal{M}, s_0 \models \mathbb{P}_{\geq 0.27}(\neg(4 \vee 10 \vee 5 \vee 9 \vee 6) \mathbf{U}^{\leq 5} won)$

For part b and c you should setup the smaller versions of the transition matrices $\mathbf{A}_{\varphi, \psi}$, i.e. the matrix containing rows and columns for states that in the satisfaction set of φ or ψ as well as the vectors \mathbf{b}_{ψ} . Use these to compute the probabilities.

2. Translate the "Craps"-model into the **PRISM**-language and validate your solutions.

If you have not done so already, please download a docker image from

<https://hub.docker.com/r/movesrwith/storm/tags/>

The debug image of the current version will suffice:

```
docker pull movesrwth/storm:1.6.4-debug
```

You can then start the container in the directory that holds your **PRISM**-model:

```
sudo docker run -it -v "$(pwd):/home" movesrwth/storm:1.6.4-debug
```

Depending on your implementation you can ask the modelchecker to give you the probabilities for the initial state:

```
storm --prism /home/<your-prism-model-file> --prop "P=? [ ! ('8'|'9'|'10') U<=5 'won']"  
--general:exact
```

Note that you have to replace the labels in this example command or use the explicit state informationi (e.g. `s=0|s=2`, `s=1&d=2`, etc.).

Please submit your model as well as the command line output of your model checking queries.