



# Counterexamples in Probabilistic LTL Model Checking for Markov Chains

Matthias Schmalz<sup>1</sup>    Daniele Varacca<sup>2</sup>    Hagen Völzer<sup>3</sup>

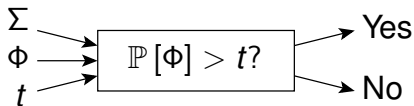
<sup>1</sup>ETH Zurich, Switzerland

<sup>2</sup>PPS - CNRS & Univ. Paris 7, France

<sup>3</sup>IBM Research – Zurich, Switzerland

September 1st, 2009

# Probabilistic Model Checking



$\Sigma$ : discrete-time finite-state Markov chain

$\Phi$ : **linear-time** temporal logic (LTL) formula

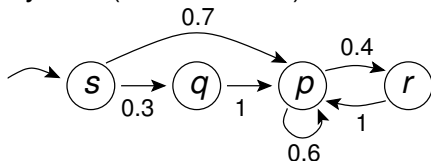
- “One of the **most important** advantages of model checking ... is its **counterexample** facility.” (Clarke et al.)

# Contributions

- a **way of representing** counterexamples in probabilistic LTL model checking
- a method supporting the user in **finding the error**
- **algorithms** for computing our counterexample representations

# Terminology

System (Markov chain)  $\Sigma$ :



**Notion:**

**Example:**

Path  $x$

$s q p r p r \dots$

Property  $Y$

$spr \uparrow$

(set of paths **with prefix  $spr$** )

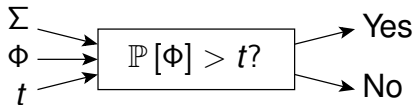
**Sat**( $\square \diamond r$ )

(set of paths infinitely often visiting  $r$ )

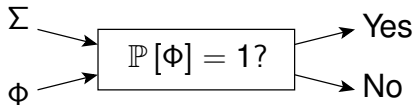
- Transition probabilities are positive.
- **Paths** are **infinite**.
- Properties are sets of paths.
- Probability of a property:  $\mathbb{P}[spr \uparrow] = 0.7 \cdot 0.4$ .

# Quantitative and Qualitative

**Quantitative** Probabilistic Model Checking:



**Qualitative** Probabilistic Model Checking:



# Outline

Qualitative Counterexamples

Other Results

# Validity: Counterexample

Specification:  $A\Phi$

The model checker claims:  $\Sigma \not\models A\Phi$

Counterexample: a path violating  $\Phi$

The user finds the bug by **inspecting the counterexample**.

# Satisfiability: Simulation

Specification:  $E \diamond \textit{jackpot}$

The model checker claims:  $\Sigma \not\models E \diamond \textit{jackpot}$

Counterexample: set of all paths of  $\Sigma$  (**useless**)

How to find the bug?

- The user defined  $\Sigma$  and  $\Phi$ .
- He has an idea how to reach the jackpot.
- The user tries to reach the jackpot.
- The user finds the bug by **simulating the system**.

# Probabilistic Correctness: Interaction

Validity

$\Sigma \models A\Phi$

$\Rightarrow$

Probabilistic  
Correctness

$\mathbb{P}[\Phi] = 1$

$\Rightarrow$

Satisfiability

$\Sigma \models E\Phi$

Counterexample:  
mc creates  
a path.

$\rightsquigarrow$

**Interaction:**  
both create  
a path.

$\rightsquigarrow$

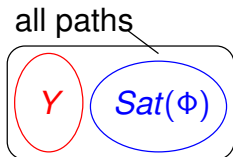
Simulation:  
user creates  
a path.

# Our Approach

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:** a property  $Y$  with

1.  $Y \cap \text{Sat}(\Phi) = \emptyset$ ,
2.  $\mathbb{P}[Y] > 0$ .

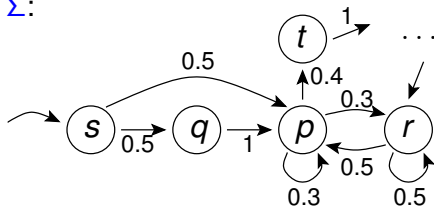


**Interaction:**

- The user learns why 1 and 2 hold.
- Helps the user find a bug.

# An Example System

$\Sigma$ :

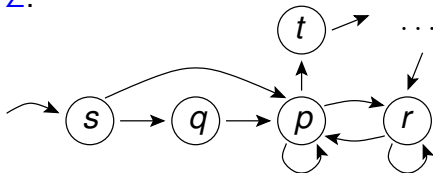


“ $\mathbb{P}[\Phi] = 1$ ”

- is **independent of precise transition probabilities!**
- only depends on which states are connected by a transition.

# An Example System

$\Sigma$ :

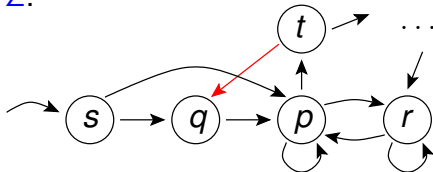


“ $\mathbb{P}[\Phi] = 1$ ”

- is **independent of precise transition probabilities!**
- only depends on which states are connected by a transition.

# An Example System

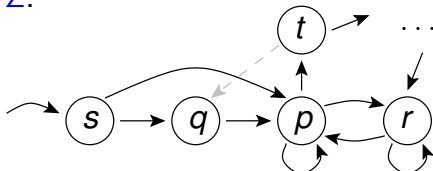
$\Sigma$ :



Bug: **transition**  $t \rightarrow q$  is **missing**

# An Example System

$\Sigma$ :



I will ...

- give a specification  $\Phi$ ,
- give a counterexample  $Y$  in our representation,
- explain the interaction helping the user find the bug.

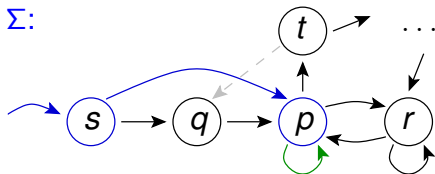
# Finitary Counterexamples

Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?



Try a **finitary** counterexample, e.g.,  $Y := sp\uparrow$ .

- $Y \cap \text{Sat}(\Phi) \neq \emptyset$ , as  
 $sp\overset{r}{p} \in Y \cap \text{Sat}(\Phi)$ .

$\Rightarrow Y$  is no counterexample.

Moreover: there is no finitary counterexample!

# Beyond Finitary Counterexamples

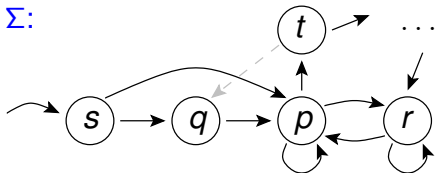
Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



Counterexample:  $Y := sp^\uparrow \cap \text{Sat}(\square \diamond rr)$

- $Y \cap \text{Sat}(\Phi) \subseteq sp^\uparrow \cap \text{Sat}(\diamond q) = \emptyset$ .
- $rr$  belongs to a bsc reachable after  $sp$ .
- Hence,  $\mathbb{P}[Y] = \mathbb{P}[sp^\uparrow] > 0$ .

$\Rightarrow Y$  is a counterexample.

# Finding the Bug

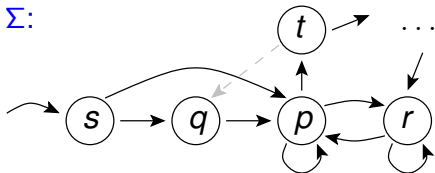
## Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

## Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



The model checker outputs  $Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$  and explains:

1.  $rr$  is in a bsc reachable after  $sp$ .
  2.  $Y \cap \text{Sat}(\Phi) = \emptyset$ .
- $\Rightarrow \mathbb{P}[\Phi] < 1$ .

# Finding the Bug

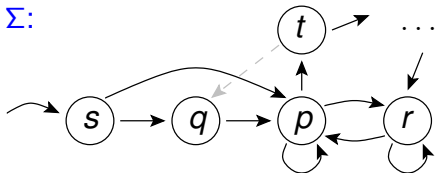
## Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

## Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



The model checker outputs  $Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$  and explains:

1.  $rr$  is in a bsc reachable after  $sp$ . ✓
  2.  $Y \cap \text{Sat}(\Phi) = \emptyset$ .
- $\Rightarrow \mathbb{P}[\Phi] < 1$ .

# Finding the Bug

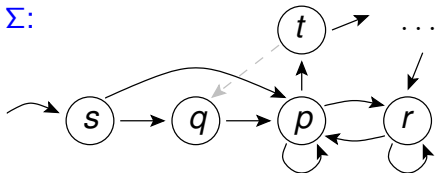
Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



The model checker outputs  $Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$  and explains:

1.  $rr$  is in a bsc reachable after  $sp$ . ✓
  2.  $Y \cap \text{Sat}(\Phi) = \emptyset$ .
- $\Rightarrow \mathbb{P}[\Phi] < 1$ . ✓

# Finding the Bug

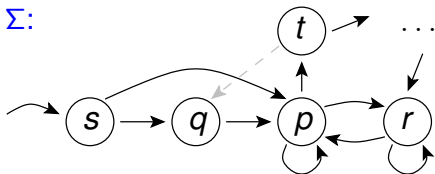
Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



The model checker outputs  $Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$  and explains:

1.  $rr$  is in a bsc reachable after  $sp$ . ✓

2.  $Y \cap \text{Sat}(\Phi) = \emptyset$ . ???

$\Rightarrow \mathbb{P}[\Phi] < 1$ . ✓

# Finding the Bug

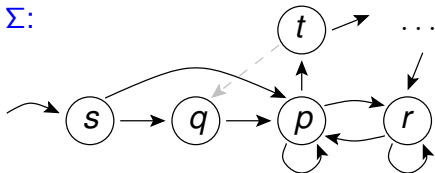
## Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

## Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



$$Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$$

Why is  $Y \cap \text{Sat}(\Phi) = \emptyset$ ?

- **User** and **MC** create a path  $x$ .
- **MC** ensures  $x \in Y$ .
- **User** aims for  $x \models \Phi$ .
- By failing the **user** finds the bug!

# Finding the Bug

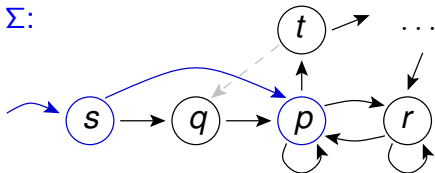
## Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

## Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



$$Y := sp \uparrow \cap \text{Sat}(\square \diamond rr)$$

$sp$

Why is  $Y \cap \text{Sat}(\Phi) = \emptyset$ ?

- **User** and **MC** create a path  $x$ .
- **MC** ensures  $x \in Y$ .
- **User** aims for  $x \models \Phi$ .
- By failing the **user** finds the bug!

# Finding the Bug

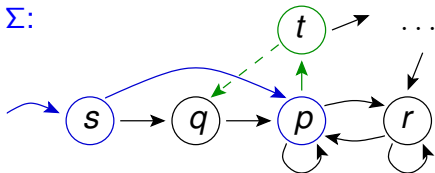
## Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

## Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



$$Y := sp\uparrow \cap \text{Sat}(\square \diamond rr)$$

*s p t q*

Why is  $Y \cap \text{Sat}(\Phi) = \emptyset$ ?

- **User** and **MC** create a path  $x$ .
- **MC** ensures  $x \in Y$ .
- **User** aims for  $x \models \Phi$ .
- By failing the **user** finds the bug!

# Finding the Bug

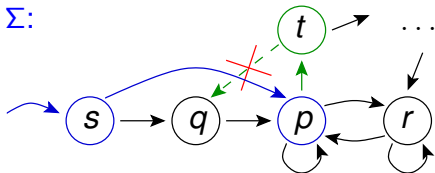
Specification:

$$\Phi := \square \diamond rr \Rightarrow \diamond q$$

Question:

Why is  $\mathbb{P}[\Phi] < 1$ ?

$\Sigma$ :



$$Y := sp \uparrow \cap \text{Sat}(\square \diamond rr)$$

~~s p t q~~

Why is  $Y \cap \text{Sat}(\Phi) = \emptyset$ ?

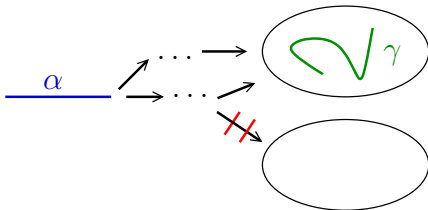
- User and MC create a path  $x$ .
- MC ensures  $x \in Y$ .
- User aims for  $x \models \Phi$ .
- By failing the user finds the bug!



# Finite Path Leading to a Recurrent Word

## Definition

- **Recurrent word** := finite path fragment belonging to a bscc
- A **finite path**  $\alpha$  (almost surely) leads to a **recurrent word**  $\gamma \neq \lambda$ .  $\iff$  The bscc of  $\gamma$  is the only bscc reachable after  $\alpha$ .



# Qualitative Counterexamples

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:**  $Y := \alpha \uparrow \cap \text{Sat}(\Box \Diamond \gamma)$ , where

1.  $\gamma$  recurrent
2.  $\alpha$  (almost surely) leads to  $\gamma$
3.  $Y \cap \text{Sat}(\Phi) = \emptyset$

## Theorem (Soundness)

(a)  $1, 2 \implies \mathbb{P}[\Box \Diamond \gamma \mid \alpha \uparrow] = 1$  and hence  $\mathbb{P}[Y] > 0$

(b)  $1, 2, 3 \implies \mathbb{P}[\Phi \mid \alpha \uparrow] = 0$  and hence  $\mathbb{P}[\Phi] \leq 1 - \mathbb{P}[\alpha \uparrow] < 1$

- $\alpha$  explains how much probability is lost.
- $\alpha$  explains where the probability is lost.
- $\gamma$  explains why the probability is lost.

## Qualitative Counterexamples

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:**  $Y := \alpha\uparrow \cap \text{Sat}(\Box\Diamond\gamma)$ , where

1.  $\gamma$  recurrent
2.  $\alpha$  (almost surely) leads to  $\gamma$
3.  $Y \cap \text{Sat}(\Phi) = \emptyset$

### Theorem (Soundness)

(a)  $1, 2 \implies \mathbb{P}[\Box\Diamond\gamma \mid \alpha\uparrow] = 1$  and hence  $\mathbb{P}[Y] > 0$

(b)  $1, 2, 3 \implies \mathbb{P}[\Phi \mid \alpha\uparrow] = 0$  and hence  $\mathbb{P}[\Phi] \leq 1 - \mathbb{P}[\alpha\uparrow] < 1$

- $\alpha$  explains **how much** probability is lost.
- $\alpha$  explains where the probability is lost.
- $\gamma$  explains why the probability is lost.

# Qualitative Counterexamples

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:**  $Y := \alpha\uparrow \cap \text{Sat}(\Box\Diamond\gamma)$ , where

1.  $\gamma$  recurrent
2.  $\alpha$  (almost surely) leads to  $\gamma$
3.  $Y \cap \text{Sat}(\Phi) = \emptyset$

## Theorem (Soundness)

(a)  $1, 2 \implies \mathbb{P}[\Box\Diamond\gamma \mid \alpha\uparrow] = 1$  and hence  $\mathbb{P}[Y] > 0$

(b)  $1, 2, 3 \implies \mathbb{P}[\Phi \mid \alpha\uparrow] = 0$  and hence  $\mathbb{P}[\Phi] \leq 1 - \mathbb{P}[\alpha\uparrow] < 1$

- $\alpha$  explains how much probability is lost.
- $\alpha$  explains **where** the probability is lost.
- $\gamma$  explains why the probability is lost.

# Qualitative Counterexamples

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:**  $Y := \alpha\uparrow \cap \text{Sat}(\Box\Diamond\gamma)$ , where

1.  $\gamma$  recurrent
2.  $\alpha$  (almost surely) leads to  $\gamma$
3.  $Y \cap \text{Sat}(\Phi) = \emptyset$

## Theorem (Soundness)

(a)  $1, 2 \implies \mathbb{P}[\Box\Diamond\gamma \mid \alpha\uparrow] = 1$  and hence  $\mathbb{P}[Y] > 0$

(b)  $1, 2, 3 \implies \mathbb{P}[\Phi \mid \alpha\uparrow] = 0$  and hence  $\mathbb{P}[\Phi] \leq 1 - \mathbb{P}[\alpha\uparrow] < 1$

- $\alpha$  explains how much probability is lost.
- $\alpha$  explains where the probability is lost.
- $\gamma$  explains **why** the probability is lost.

# Qualitative Counterexamples

**Question:** Why is  $\mathbb{P}[\Phi] < 1$ ?

**Counterexample:**  $Y := \alpha \uparrow \cap \text{Sat}(\Box \Diamond \gamma)$ , where

1.  $\gamma$  recurrent
2.  $\alpha$  (almost surely) leads to  $\gamma$
3.  $Y \cap \text{Sat}(\Phi) = \emptyset$

**Theorem (Completeness)**

$\mathbb{P}[\Phi] < 1 \implies$  *there are  $\alpha, \gamma$  such that 1, 2, 3 hold.*

# Interaction

Conditions 1, 2, 3 can be expressed in terms of **path games** between the **user** and the **model checker**.

Condition  $i$  holds  $\iff$  the model checker has a winning strategy in the respective path game.

- To understand why a condition holds, the **user** plays the respective path game against the **model checker**.
- By losing the **user** finds the error in the system.

# Interaction

$$\text{Disjointness} - Y \cap \text{Sat}(\Phi) = \emptyset$$

The path game:

- The **model checker** ensures  $x \in Y$ .
- The **user** wins iff  $x \models \Phi$ .
- The **model checker** has a winning strategy  $\iff$

The **user** is unable to establish  $x \models \Phi \iff$

$$Y \cap \text{Sat}(\Phi) = \emptyset$$

- The game corresponds to the **Banach-Mazur** game.

# Interaction

$$\text{Disjointness} - Y \cap \text{Sat}(\Phi) = \emptyset$$

The path game:

$$x = \alpha$$

- The **model checker** ensures  $x \in Y$ .
- The **user** wins iff  $x \models \Phi$ .
- The **model checker** has a winning strategy  $\iff$

The **user** is unable to establish  $x \models \Phi \iff$

$$Y \cap \text{Sat}(\Phi) = \emptyset$$

- The game corresponds to the **Banach-Mazur** game.











# Outline

Qualitative Counterexamples

Other Results

## Quantitative Counterexamples

Quantitative Counterexample:  $Y := W\uparrow \cap \text{Fair}_\Sigma(R)$

- $W$ : set of finite paths
- $R$ : set of recurrent words
- $Y \cap \text{Sat}(\Phi) = \emptyset$ ,  $\mathbb{P}[Y]$  “sufficiently” large

Theorem (Soundness)

- $\mathbb{P}[\Phi] \leq 1 - \mathbb{P}[W\uparrow]$
- $\mathbb{P}[\Phi \mid W\uparrow] = 0$

Theorem (Completeness)

$\mathbb{P}[\Phi] \leq 1 - t \implies$  *There is a counterexample  $W\uparrow \cap \text{Fair}_\Sigma(R)$ , where  $R$  contains one rec. word per bsc, and  $W$  is **regular**.*

**Interaction:** as  $W$  is regular, various techniques from the literature can be applied for presenting  $W$  to the user.

# Computing Counterexamples

We have developed non-trivial extensions of an algorithm of Courcoubetis and Yannakakis (1995).

	Complexity in $ \Sigma $	Complexity in $ \Phi $
$\alpha, \gamma$	$ \Sigma $	exponential
$\alpha$ of max. probability	$ \Sigma  \cdot \log  \Sigma $	doubly exp.
$W$	$ \Sigma $	doubly exp.
$R$	$ \Sigma  \cdot \#\text{bsccs}$	exponential

# Summary

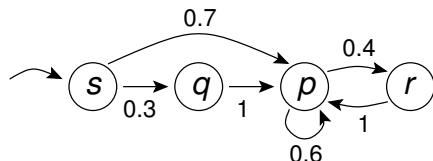
- A **qualitative** counterexample can be represented as  $\alpha \uparrow \cap \text{Sat}(\Box \Diamond \gamma)$ .
- A **quantitative** counterexample can be represented as  $W \uparrow \cap \text{Fair}_{\Sigma}(R)$ , where  $W$  is **regular**.
- We describe an **interactive game** that supports the user in **finding the error**.
- We have developed algorithms **computing** our counterexample representations.

Future directions:

- Generalize results for Markov Decision Processes.
- Case studies



## Periodic Counterexamples



- Each periodic path has probability zero, e.g.,  $\mathbb{P}[\{s(pr)^\omega\}] = 0$ .
  - The set of all periodic paths is countable.
- ⇒ The set of all periodic paths has probability zero.

Sets of **periodic paths can** in general **not be used** as counterexamples!