Optimal test case generation using game theory

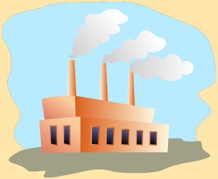
Making tests robustly exploit and explore

Léo Henry
supervisors:
Nicolas Markey Thierry Jéron

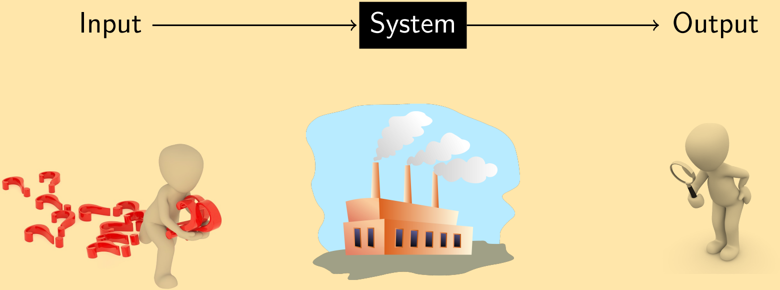
Univ. Rennes, INRIA & CNRS
Rennes (France)

Black-box conformance timed testing

Input ⟶ System ⟶ Output



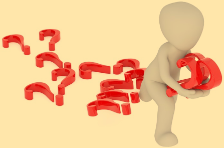
Black-box conformance timed testing

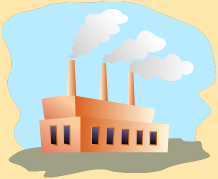


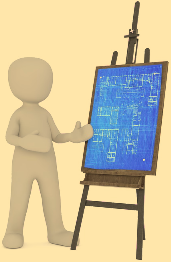
Input → System → Output

Black-box conformance timed testing

Input → **System** → Output



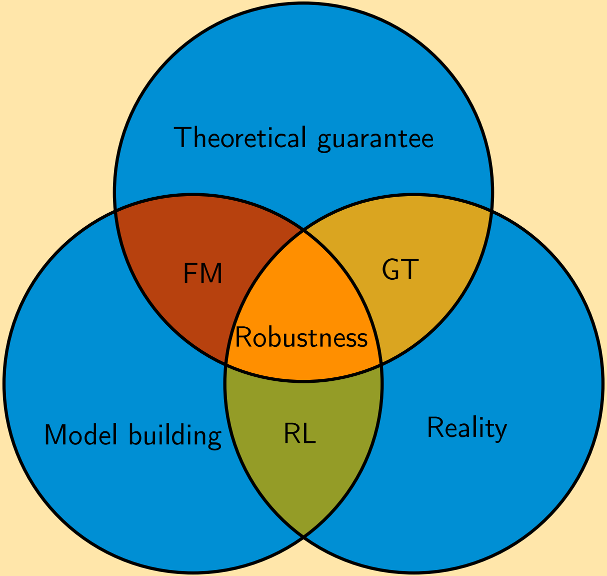




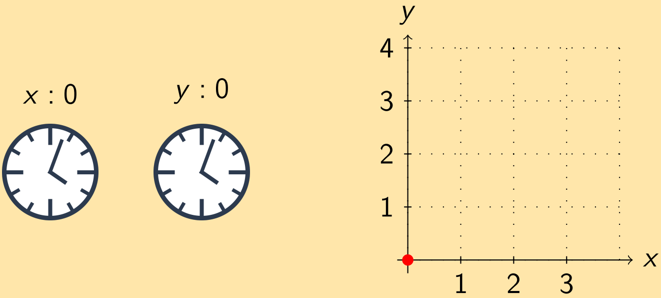
Conformance testing

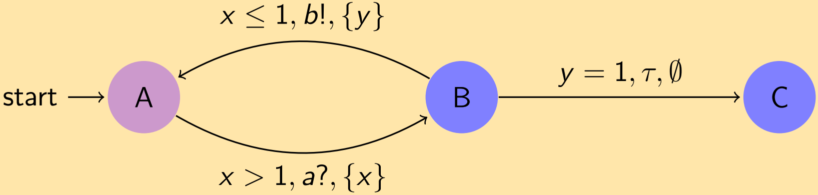
Timed testing

At the edge of many domains

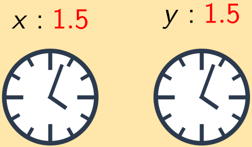


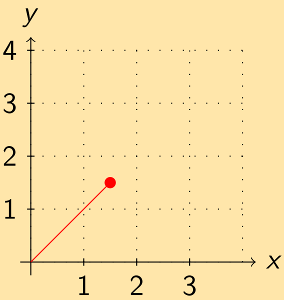
A model for timed systems: TAIO

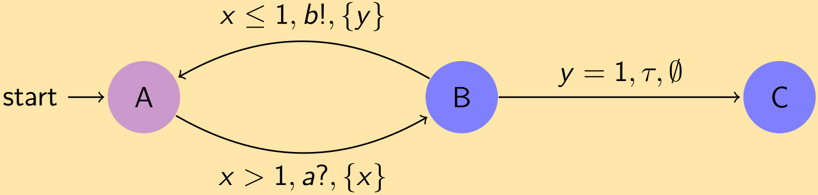




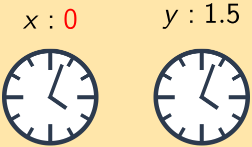
A model for timed systems: TAIO

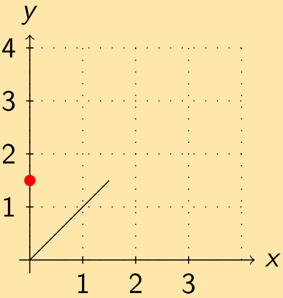


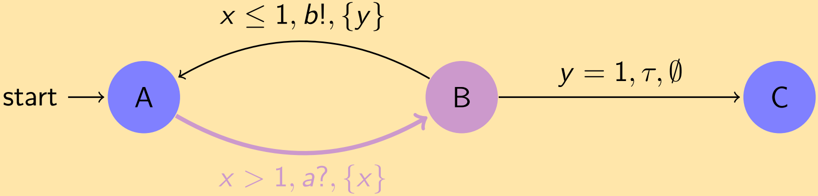




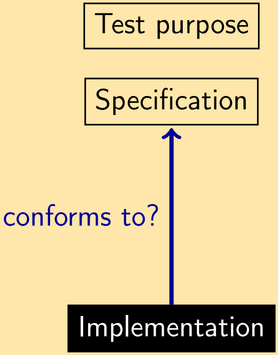
A model for timed systems: TAIO



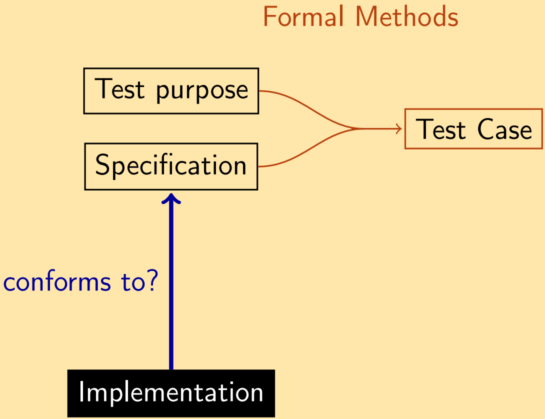




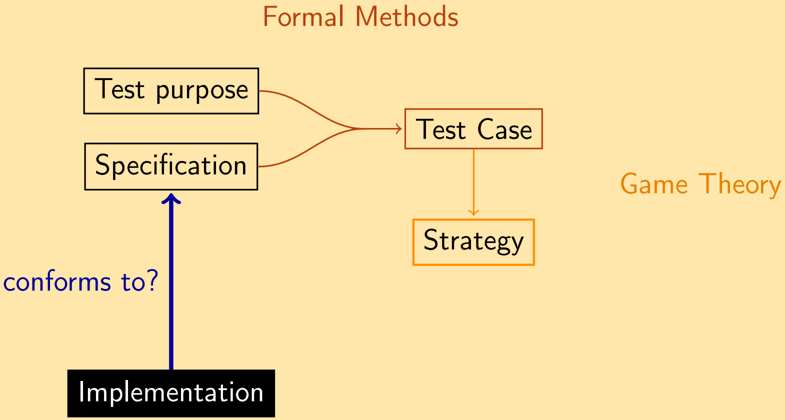
Setup



Setup


Formal Methods
Test purpose
Specification
Test Case
conforms to?
Implementation

Setup



Formal Methods

Test purpose

Specification

Test Case

Strategy

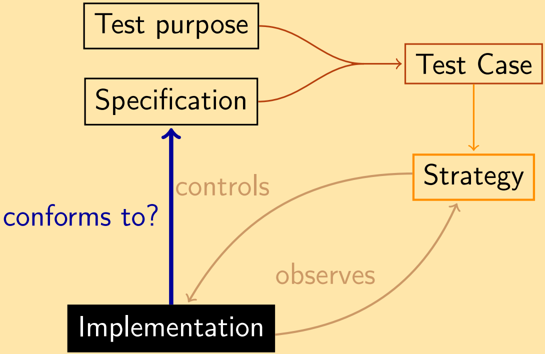
Game Theory

conforms to?

Implementation

Setup

Formal Methods



Test purpose

Specification

Test Case

Strategy

controls

conforms to?

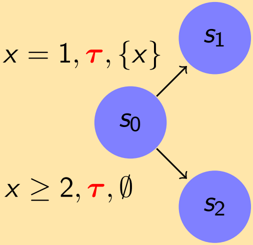
observes

Implementation

Game Theory

First challenges

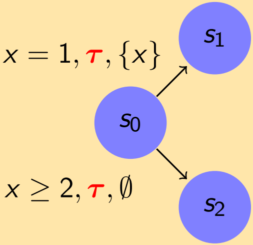
Partial observability[1] Consequence of the black-box assumption.

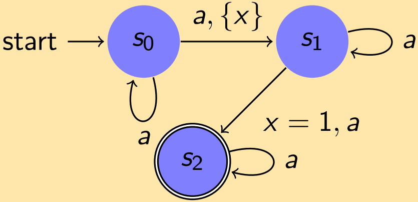


[1] Bertrand, Jéron, Stainer, and Krichen, "Off-line test selection with test purposes for non-deterministic timed automata", 2012, *Logical Methods in Computer Science*.

First challenges

Partial observability[1] Consequence of the black-box assumption.



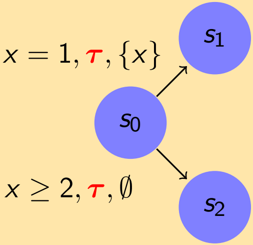


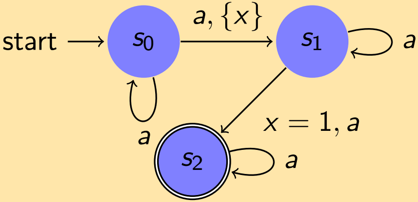
No determinizability[1] A deterministic equivalent does not always exist.

[1] Bertrand, Jéron, Stainer, and Krichen, "Off-line test selection with test purposes for non-deterministic timed automata", 2012, *Logical Methods in Computer Science*.

First challenges

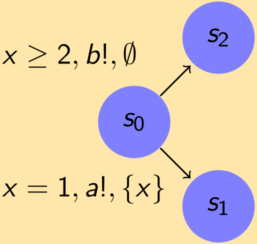
Partial observability[1] Consequence of the black-box assumption.





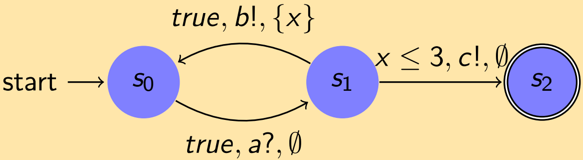
No determinizability[1] A deterministic equivalent does not always exist.

Partial controllability Price of design liberty.



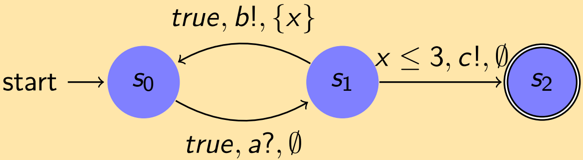
[1] Bertrand, Jéron, Stainer, and Krichen, "Off-line test selection with test purposes for non-deterministic timed automata", 2012, *Logical Methods in Computer Science*.

Difficult timed games



[2] Henry, Jéron, and Markey, "Control Strategies for Off-Line Testing of Timed Systems", 2018, *SPIN*.

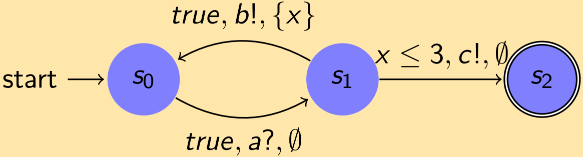
Difficult timed games



We rely on the *fairness* of the implementation.

[2] Henry, Jéron, and Markey, "Control Strategies for Off-Line Testing of Timed Systems", 2018, *SPIN*.

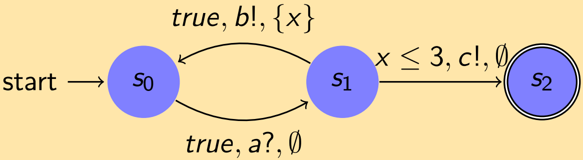
Difficult timed games



We rely on the *fairness* of the implementation. We can construct a strategy that is *winning*

[2] Henry, Jéron, and Markey, "Control Strategies for Off-Line Testing of Timed Systems", 2018, *SPIN*.

Difficult timed games



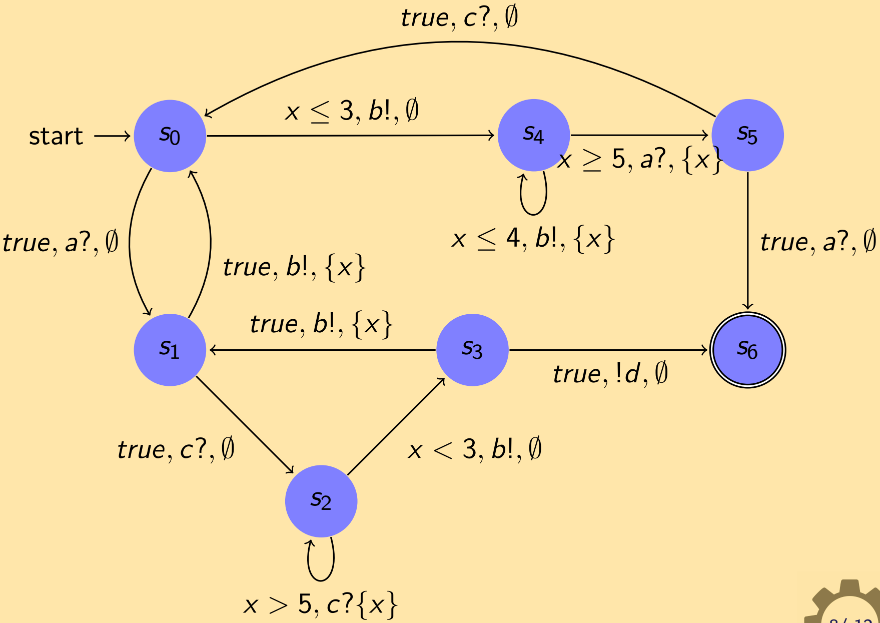
We rely on the *fairness* of the implementation. We can construct a strategy that is *winning*,

and create test cases that are, $\forall \mathcal{I} \in \mathcal{I}(\mathcal{S}),\ \forall(\mathcal{G}, f) \in \mathcal{TC}(\mathcal{S})$:

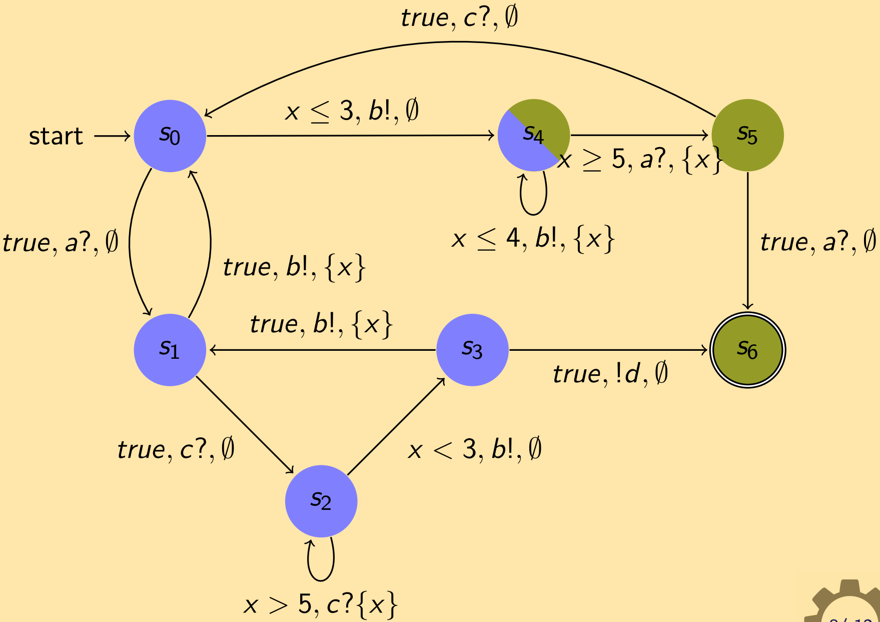
- sound: $\mathcal{I}$ fails $(\mathcal{G}, f) \Rightarrow \neg(\mathcal{I}$ tioco $\mathcal{S})$;
- Up to the determinization approximations:
  - strict: $\neg$(Behaviour$(\mathcal{G}, f, \mathcal{I})$ tioco $\mathcal{S}) \Rightarrow \mathcal{I}$ fails $(\mathcal{G}, f)$;
  - exhaustive: $\neg(\mathcal{I}$ tioco $\mathcal{S}) \Rightarrow \exists(\mathcal{G}, f) \in \mathcal{TC}(\mathcal{S}), \mathcal{I}$ fails $(\mathcal{G}, f)$;
  - precise: there is no approximation in the acceptance condition.

[2] Henry, Jéron, and Markey, "Control Strategies for Off-Line Testing of Timed Systems", 2018, *SPIN*.

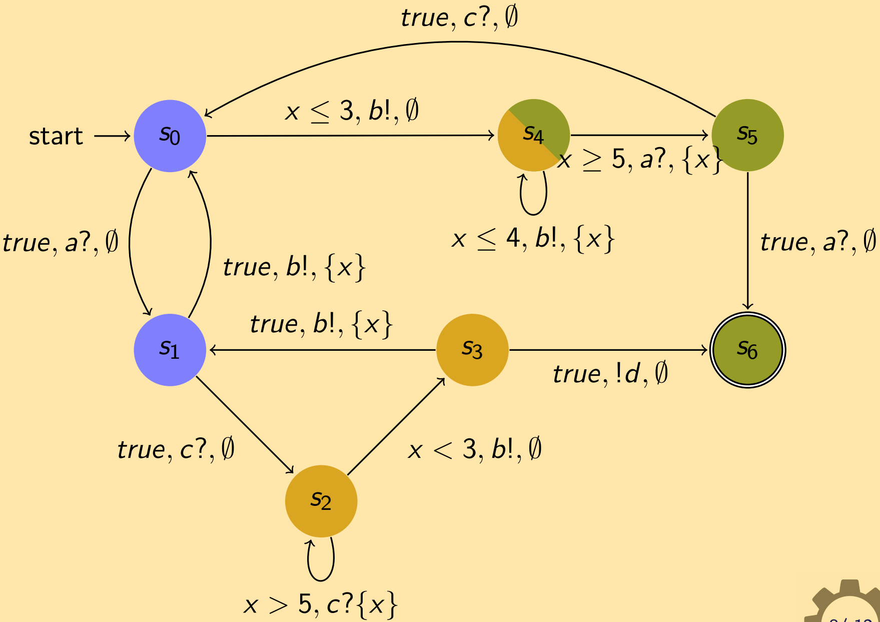
A motivating example



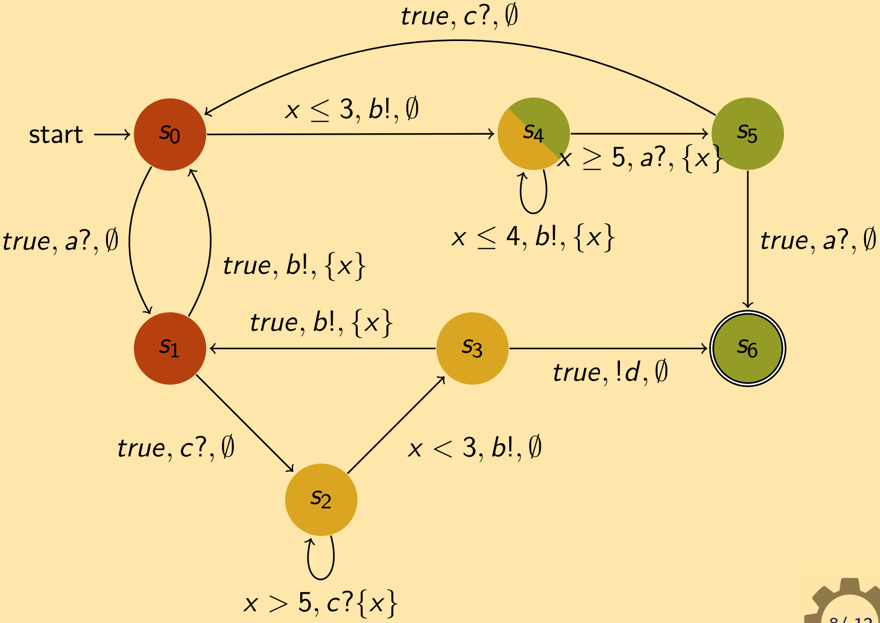
A motivating example



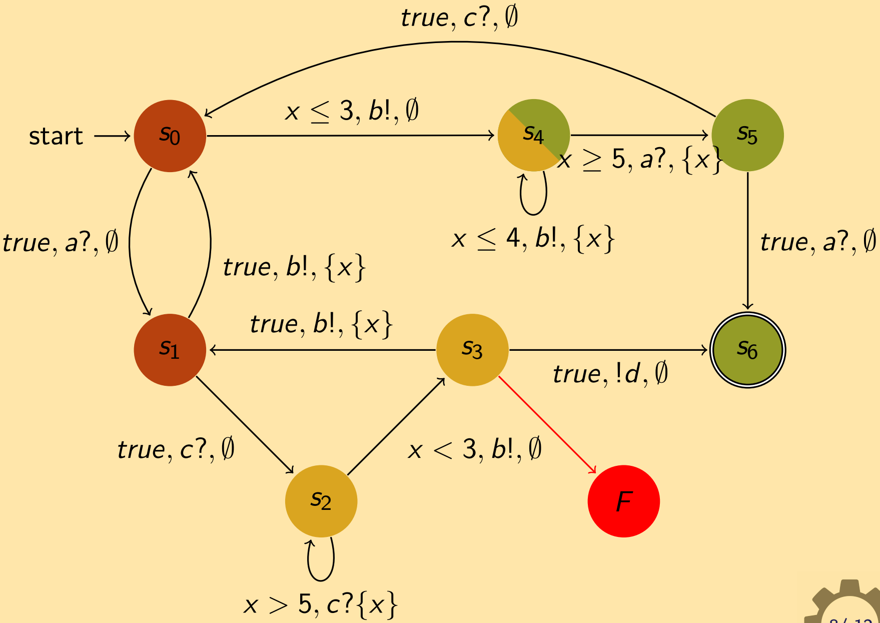
A motivating example



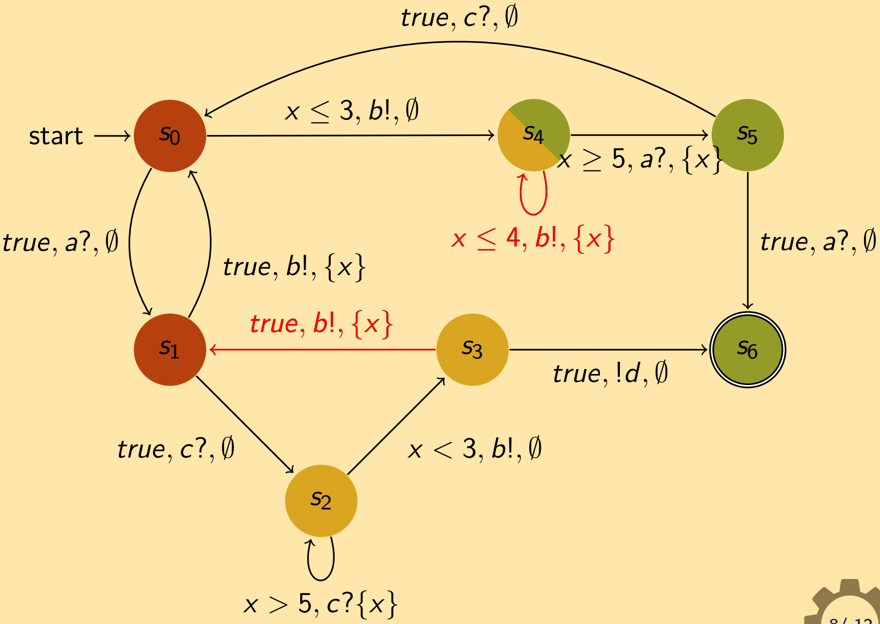
A motivating example



A motivating example



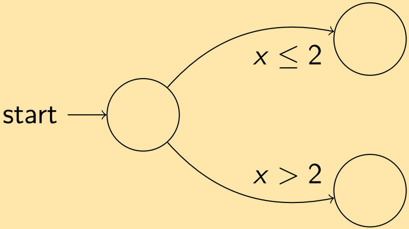
A motivating example



Model weaknesses

Reality strikes back

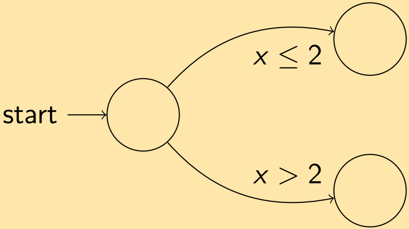
Robustness What precision do we require in the measure of time?



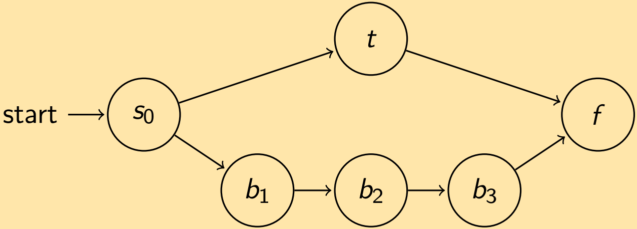
Model weaknesses

Reality strikes back

Robustness What precision do we require in the measure of time?



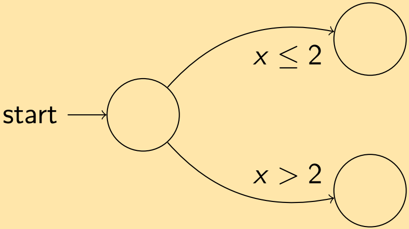
Implementation freedom We only want to enforce "$\mathcal{I}$ conforms $\mathcal{S}$"



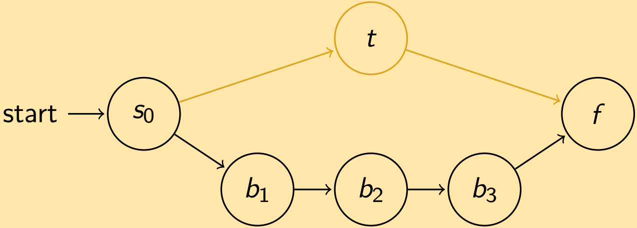
Model weaknesses

Reality strikes back

Robustness What precision do we require in the measure of time?



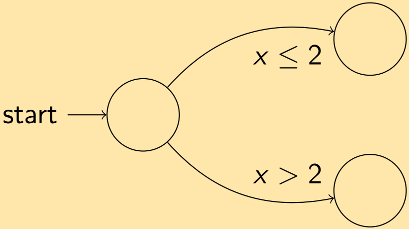
Implementation freedom We only want to enforce "$\mathcal{I}$ conforms $\mathcal{S}$"



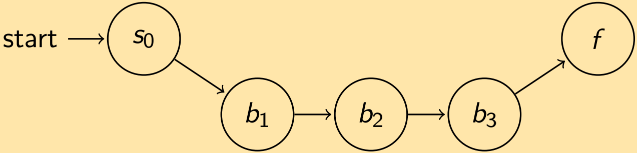
Model weaknesses

Reality strikes back

Robustness What precision do we require in the measure of time?

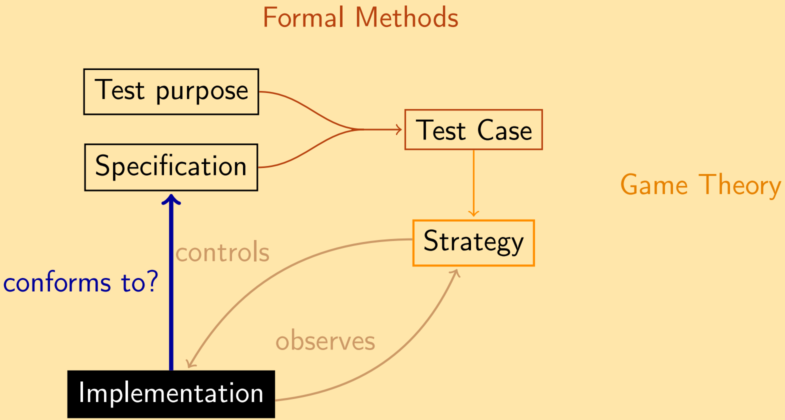


Implementation freedom We only want to enforce "$\mathcal{I}$ conforms $\mathcal{S}$"



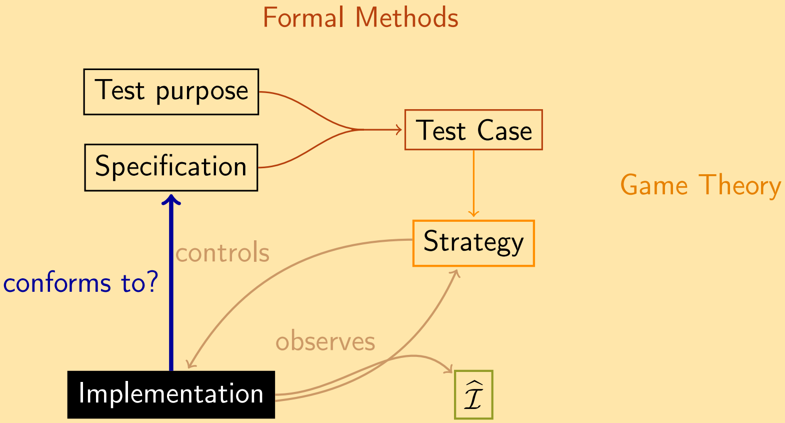
Reinforcement learning

Learning from the implementation



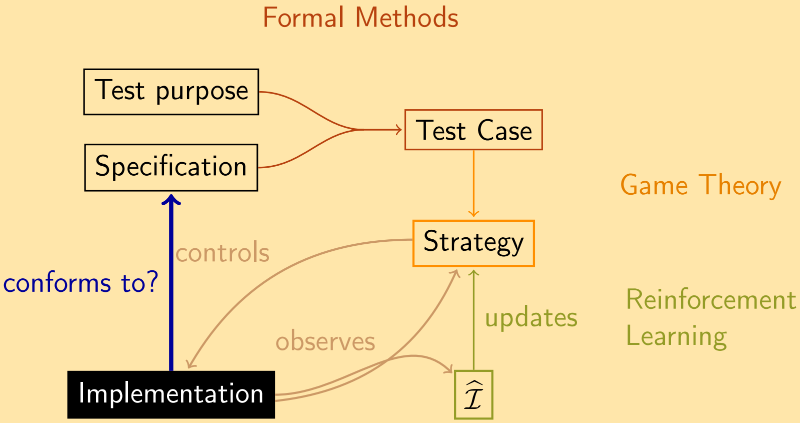
Reinforcement learning

Learning from the implementation



Reinforcement learning

Learning from the implementation



Current and ongoing work

What did I do all this time?

What is done:

- A paper on difficult games for tests... and a journal version;
- A bibliographic study on learning and reinforcement learning;

[3] Bouyer, Jaziri, and Markey, "Efficient Timed Diagnosis Using Automata with Timed Domains", 2018, *RV 2018*.

Current and ongoing work

What did I do all this time?

What is done:

- A paper on difficult games for tests... and a journal version;
- A bibliographic study on learning and reinforcement learning;

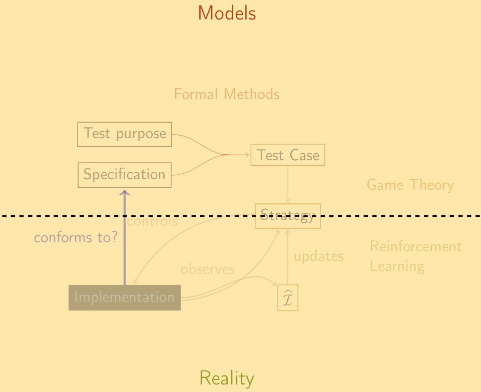
What is ongoing:

- A journal version of [BJM18] on determinization of TAs for diagnosis;
- The generalization to games with inconclusive states.

[3] Bouyer, Jaziri, and Markey, "Efficient Timed Diagnosis Using Automata with Timed Domains", 2018, *RV 2018*.

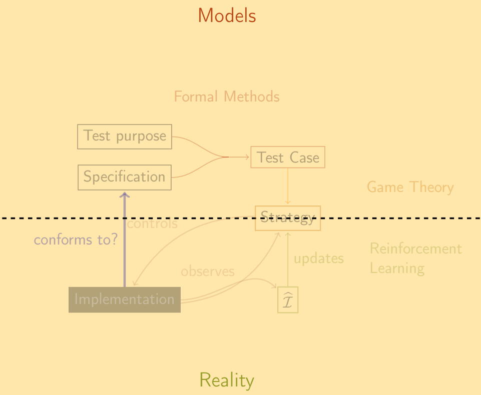
To conclude

- Formal test generation is a the edge between reality and models.



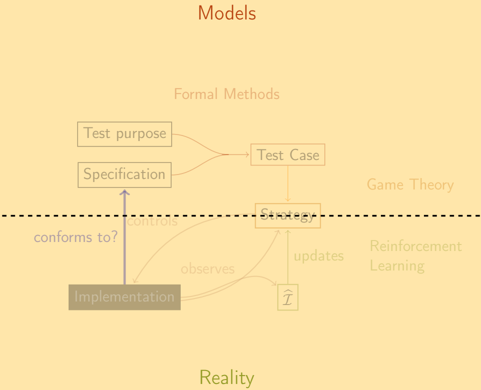
To conclude

- Formal test generation is a the edge between reality and models.
- Information from both the model and the real world *should* be exploited.



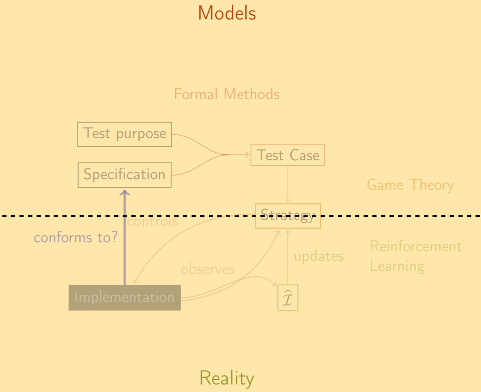
To conclude

- Formal test generation is a the edge between reality and models.
- Information from both the model and the real world *should* be exploited.
- Models robustness is greatly required.



To conclude

- Formal test generation is a the edge between reality and models.
- Information from both the model and the real world *should* be exploited.
- Models robustness is greatly required.



Game Theory

Formal Methods

Reinforcement Learning