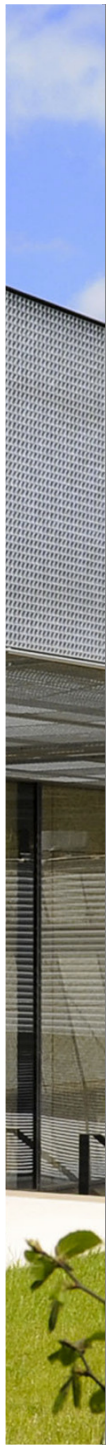
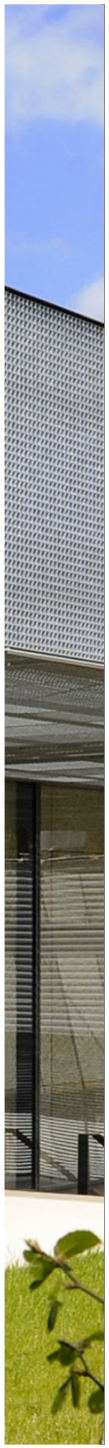


Model Checking for Probabilistic Hybrid Systems

Marta Kwiatkowska, Ernst Moritz Hahn
Oxford University Computing Laboratory

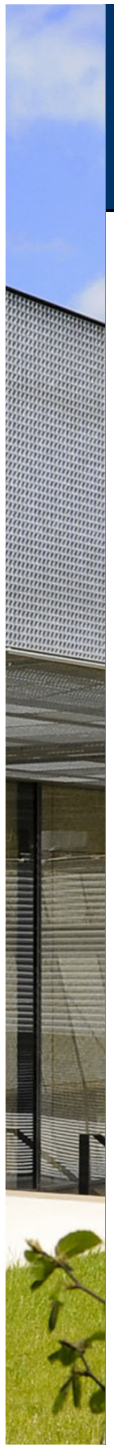
Holger Hermanns, Arnd Hartmanns
Saarland University, Dependable Systems and Software

CPSWeek'13, Philadelphia, April 2013



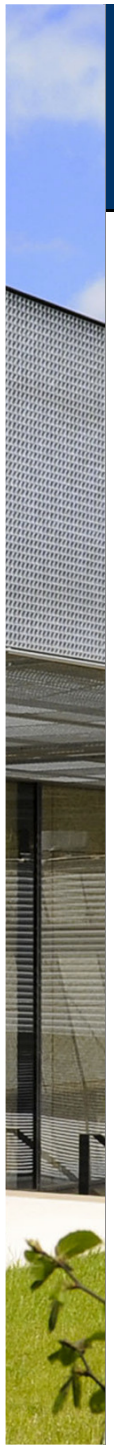
Part 3

Tools and case studies



Overview (Part 3)

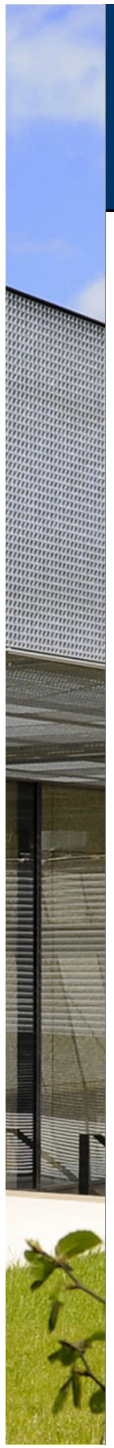
- Tools and modelling languages
  - PRISM & guarded commands
  - Modest & the Modest Toolset
- Probabilistic timed BRP (PTA)
  - Case Study & Demo
- Temperature control (PHA)
  - Case Study & Demo
- ETCS level 3 train control (SHA)
  - Case Study & Demo



Tools for Quantitative Verification

- PRISM
  www.prismmodelchecker.org
  - developed at Birmingham/Oxford University, since 1999
  - modelling language: guarded commands
  - model checking for **PTA**, **MDPs**, DTMCs and CTMCs

- The Modest Toolset
  www.modestchecker.net
  - developed at Saarland University, since 2008
  - modelling language: Modest
  - other languages also supported: e.g. guarded commands
  - model checking and simulation for different subsets of **SHA**



Modelling Languages

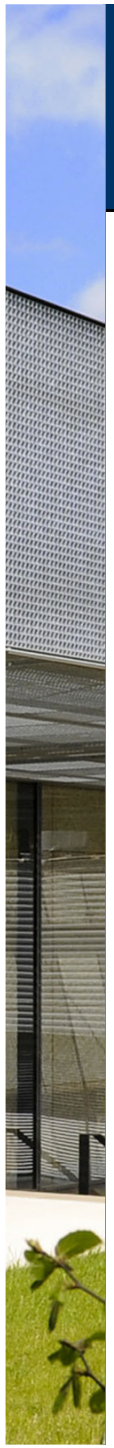
- Guarded Commands
  - low-level language
  - few, but powerful concepts

```
module sender

  s : [0..6] init 0;
  srep : [0..3];
  nrtr : [0..MAX];
  ...
  ls : bool;

  [NewFile] (s = 0) -> (s' = 1) & (i' = 1) & (srep' = 0);
  [aF] (s = 1) -> (s' = 2) & (fs' = (i = 1)) & (ls' = (i = N)) & ...
  [aB] (s = 2) -> (s' = 4) & (s_ab' = !s_ab);
  ...
  [] (s = 4) & (i < N) -> (s' = 1) & (i' = i + 1);
  [] (s = 4) & (i = N) -> (s' = 0) & (srep' = 3);
  [SyncWait] (s = 5) -> (s' = 6);
  [SyncWait] (s = 6) -> (s' = 0) & (s_ab' = false);

endmodule
```

Modelling Languages

- Modest
  - high-level language
  - focus on readability, expressivity and conciseness

```
process Sender() {
    bool bit;
    int(0..MAX) rc;
    new_file {= i = 0, rc = 0 =};
    try {
        do {
        :: when(i < N) {= i = i + 1 =};
           do {
           :: put_k {= ff = (i == 1), lf = (i == N), ab = bit =}
              alt {
              :: get_l {= bit = !bit, rc = 0 =};
                 break
              :: when(rc == MAX && i < N)
                 s_nok {= rc = 0 =};
                 throw(error)
              ...
```

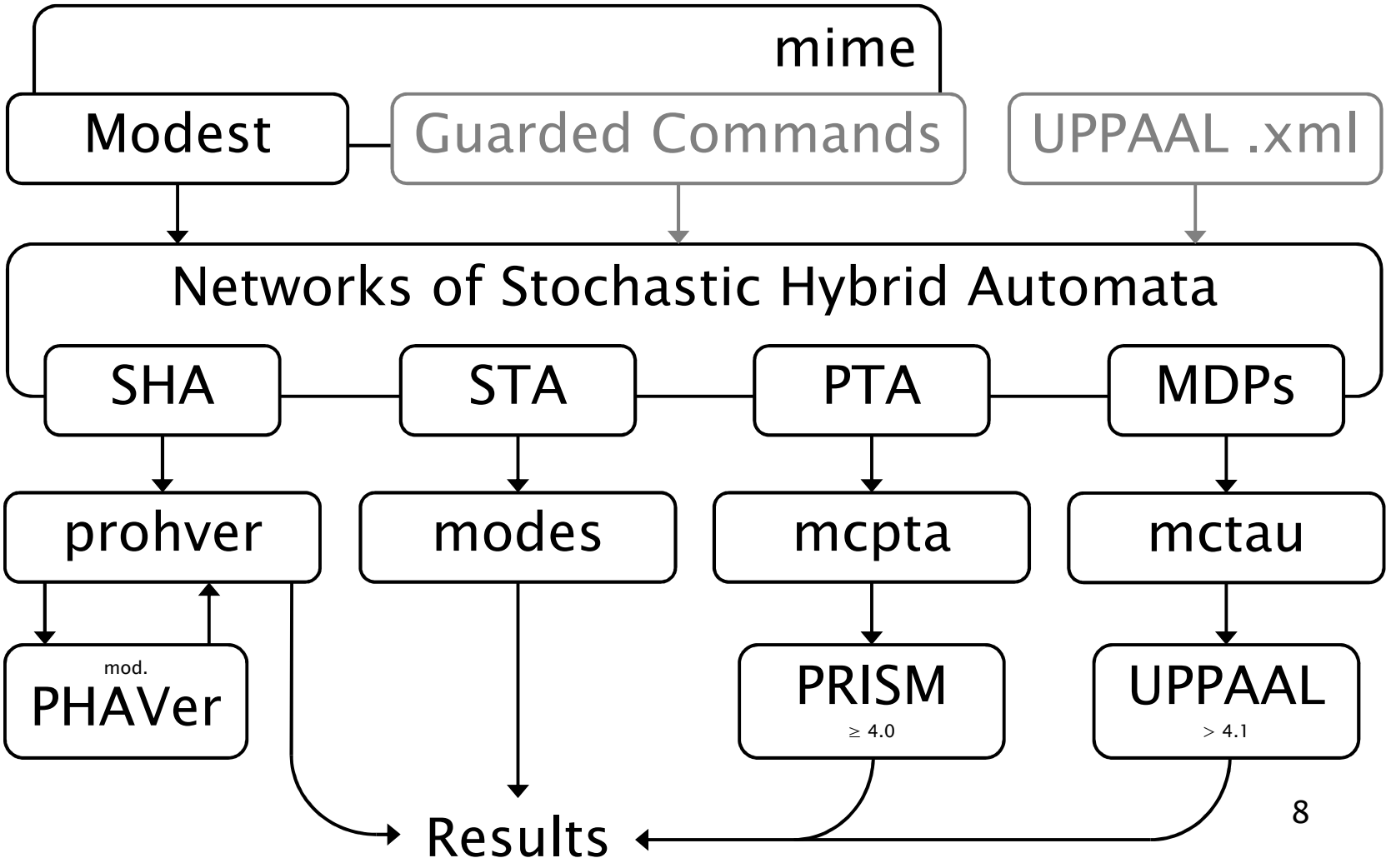


The Modest Toolset

- mcpta
  - model checking for PTA
  - using PRISM
- mctau
  - model checking for TA
  - using the UPPAAL model checker
  - more efficient than mcpta for TA models
- modes
  - statistical model checking (= simulation) for STA
  - sound treatment of nondeterminism (POR, confluence)
- prohver
  - safety verification for SHA
  - uses a modified version of PHAVer

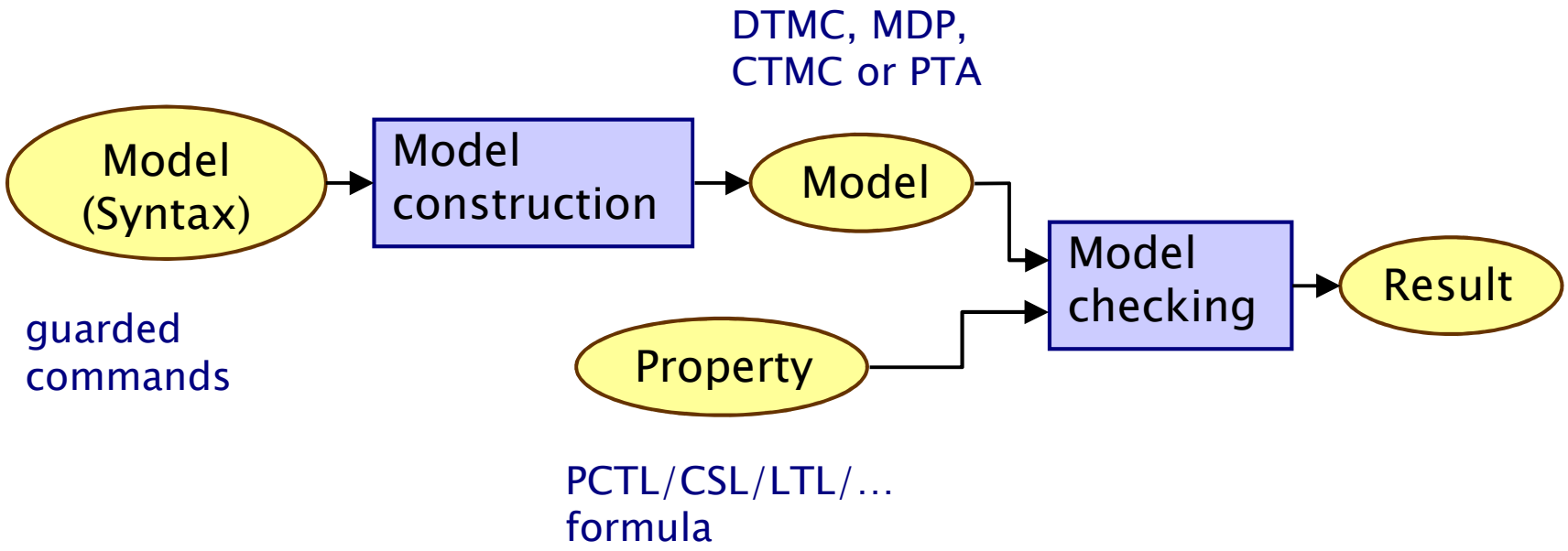
The Modest Toolset

- Toolset overview



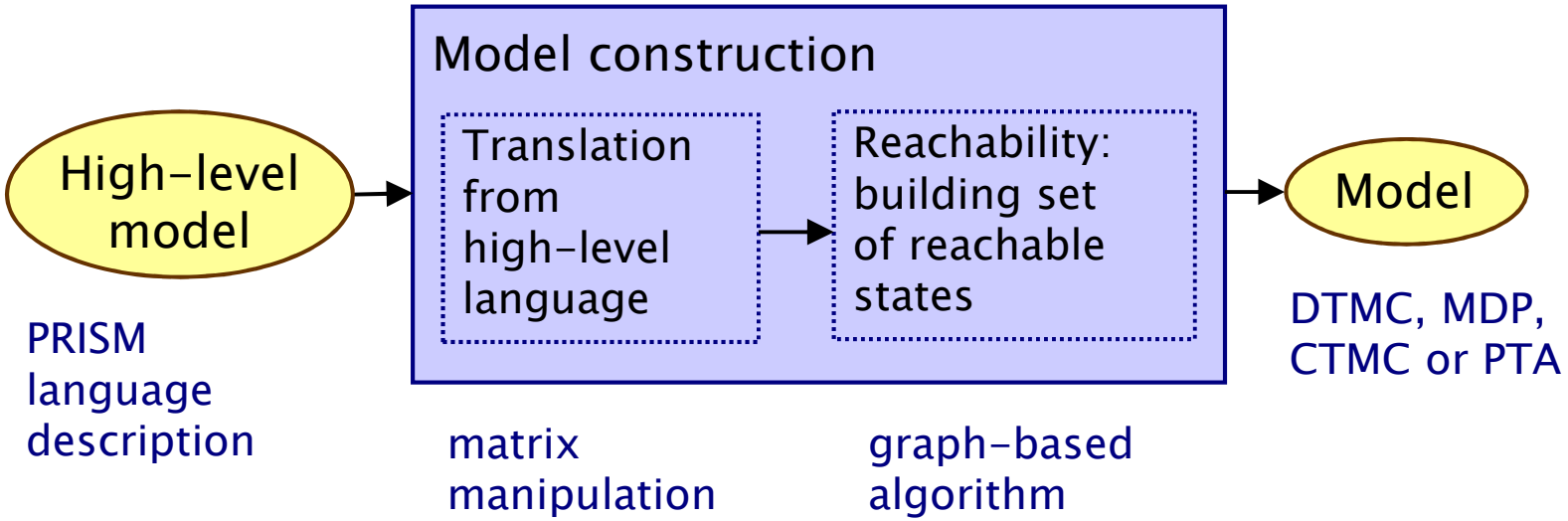
PRISM

- The probabilistic model checking process in PRISM
  - two distinct phases: model construction, model checking



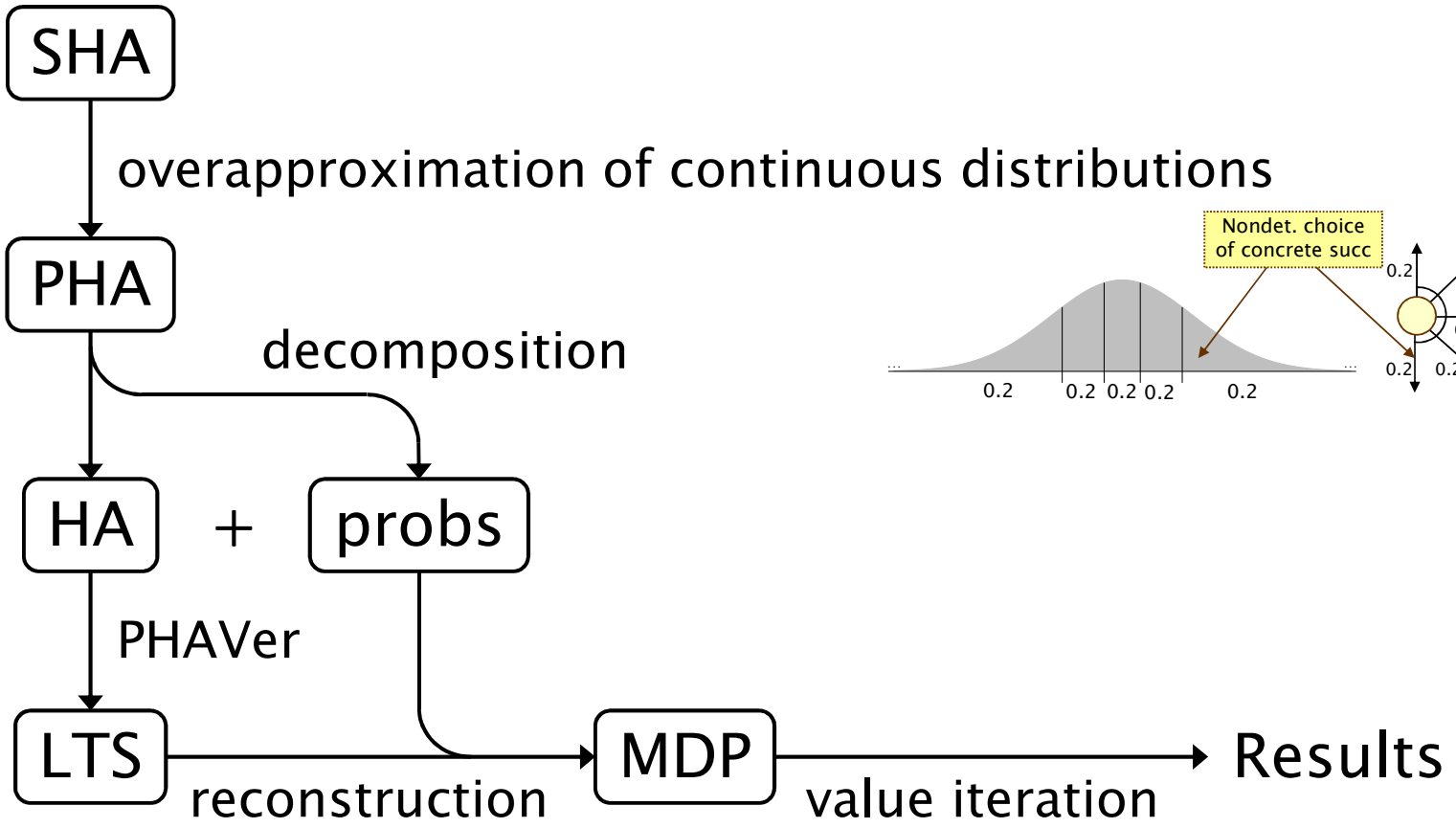
PRISM

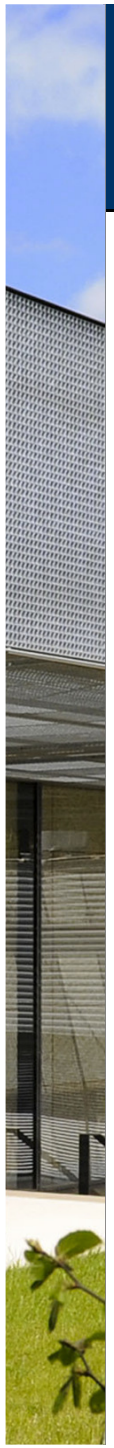
- The probabilistic model checking process in PRISM
  - two distinct phases: model construction, model checking



prohver

- The safety verification process for SHA in prohver



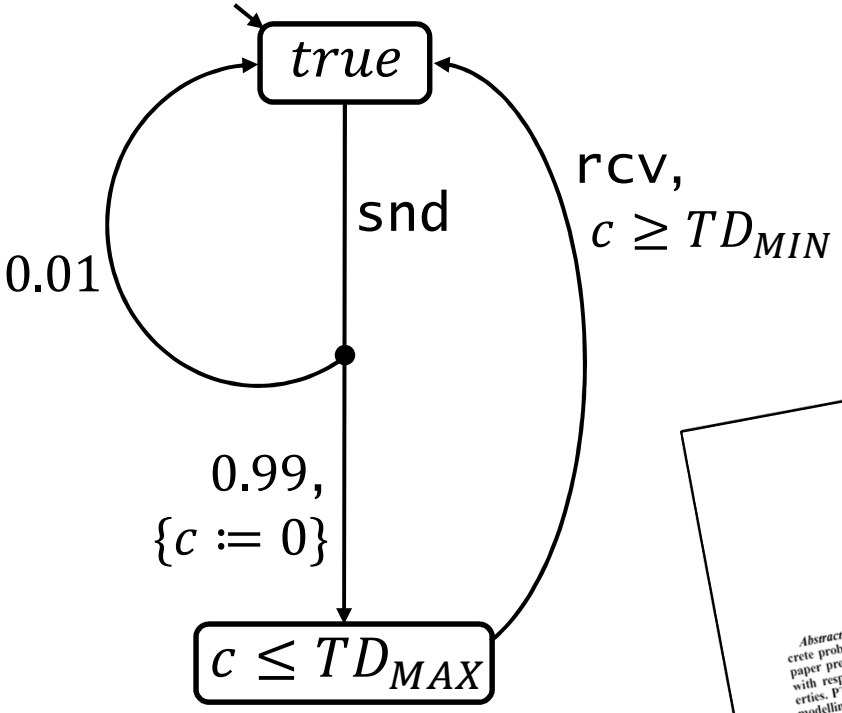


Overview (Part 3)

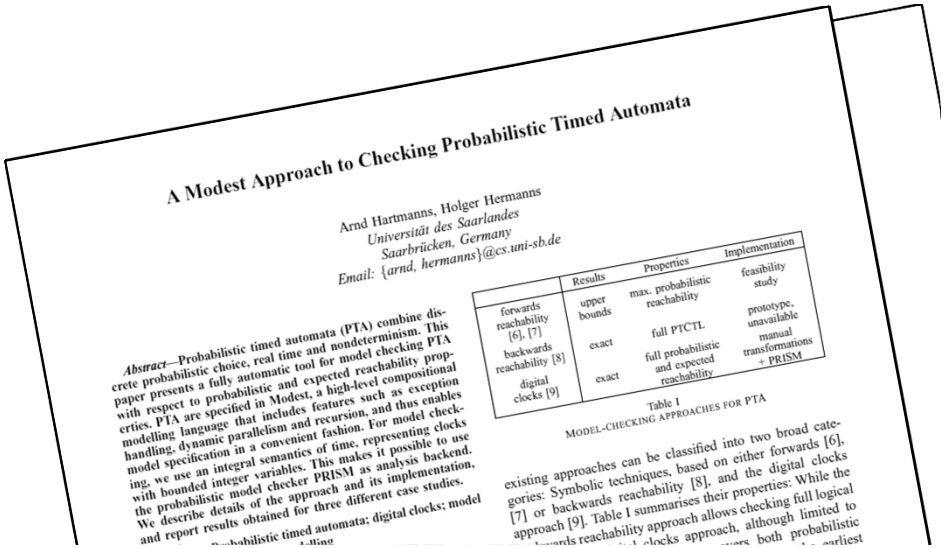
- Tools and modelling languages
  - PRISM & guarded commands
  - Modest & the Modest Toolset
- Probabilistic timed BRP (PTA)
  - Case Study & Demo
- Temperature control (PHA)
  - Case Study & Demo
- ETCS level 3 train control (SHA)
  - Case Study & Demo

Case Study: BRP

- Bounded Retransmission Protocol
  - timed model: timed automata, UPPAAL
  - probabilistic timed model: mcpta
  - allows new kinds of properties to be checked



⇨ DEMO



A Modest Approach to Checking Probabilistic Timed Automata

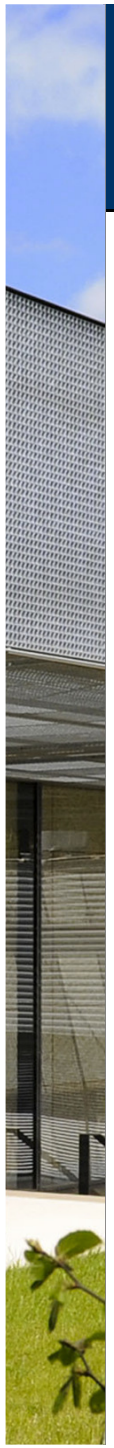
Arnd Hartmanns, Holger Hermanns
Universität des Saarlandes
Saarbrücken, Germany
Email: {arnd, hermanns}@cs.uni-sb.de

	Results	Properties	Implementation
forwards reachability [6], [7]	upper bounds	max. probabilistic reachability	feasibility study
backwards reachability [8]	exact	full PTCTL	prototype, unavailable
digital clocks [9]	exact	full probabilistic and expected reachability	manual transformations + PRISM

Table I
MODEL-CHECKING APPROACHES FOR PTA

Abstract—Probabilistic timed automata (PTA) combine discrete probabilistic choice, real time and nondeterminism. This paper presents a fully automatic tool for model checking PTA with respect to probabilistic and expected reachability properties. PTA are specified in Modest, a high-level compositional modelling language that includes features such as exception handling, dynamic parallelism and recursion, and thus enables model specification in a convenient fashion. For model checking, we use an integral semantics of time, representing clocks with bounded integer variables. This makes it possible to use the probabilistic model checker PRISM as analysis backend. We describe details of the approach and its implementation, and report results obtained for three different case studies.

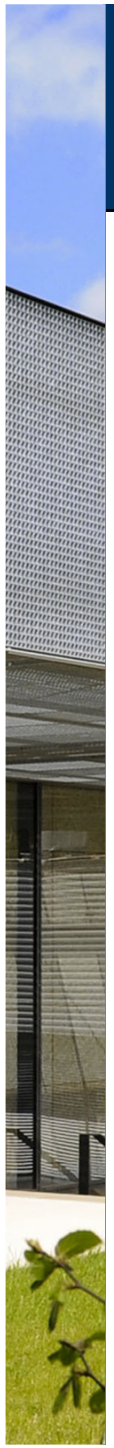
existing approaches can be classified into two broad categories: Symbolic techniques, based on either forwards [6], [7] or backwards reachability [8], and the digital clocks approach [9]. Table I summarises their properties: While the



Case Study: BRP

- Results

property		mctau	mcpta	modes
T_{A1}	true	true	true	true
T_{A2}	true	true	true	true
P_A	0	0	0	0
P_B	0	0	0	0
P_1	?	[0,1]	$4.233 \cdot 10^{-4}$	$\sim 3.0 \cdot 10^{-4}$
P_2	?	[0,1]	$2.645 \cdot 10^{-5}$	0
D_{max}	?	[0,1]	$9.996 \cdot 10^{-1}$	$\sim 9.9 \cdot 10^{-1}$
E_{max}	?	n/a	33.473	~33.473

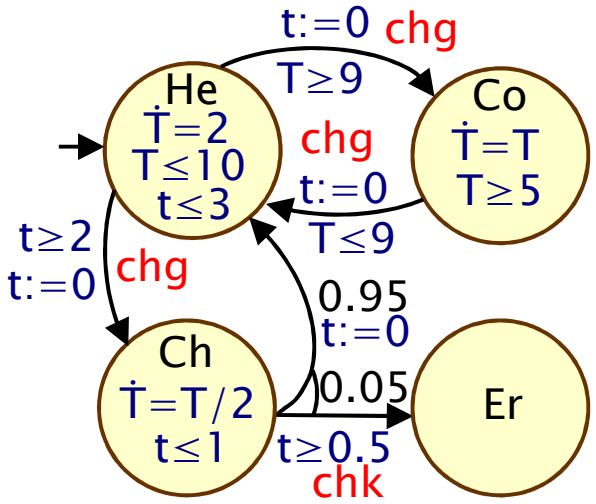


Overview (Part 3)

- Tools and modelling languages
  - PRISM & guarded commands
  - Modest & the Modest Toolset
- Probabilistic timed BRP (PTA)
  - Case Study & Demo
- Temperature control (PHA)
  - Case Study & Demo
- ETCS level 3 train control (SHA)
  - Case Study & Demo

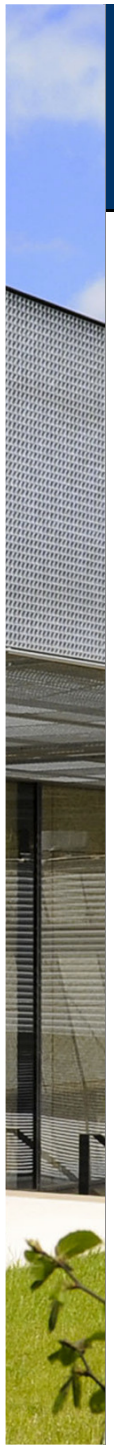
Case Study: Thermostat

- Simple temperature control model
  - PHA: finite-support probabilistic choice
  - nonlinear continuous dynamics
  - probability to reach error?




```
invariant
   (mode = m_cool =>
      T >= 0 & x <= TIME_BOUND
    & der(T) = -T & der(x) = 1 & der(t) = 1)
& (mode = m_heat =>
      T <= 10 & t <= 3 & x <= TIME_BOUND
    & der(T) = 2 & der(x) = 1 & der(t) = 1)
& (mode = m_check =>
      t <= 1 & x <= TIME_BOUND
    & der(T) = -0.5 * T & der(x) = 1 & der(t) = 1)
& (mode = m_error =>
      x <= TIME_BOUND
    & der(T) = 0 & der(x) = 0 & der(t) = 0)
endinvariant
```

⇨ DEMO

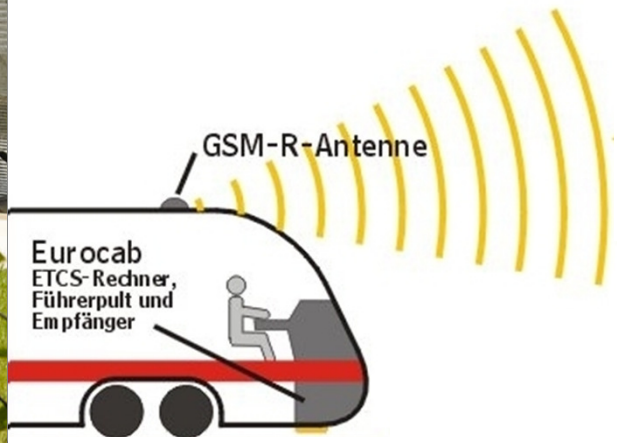


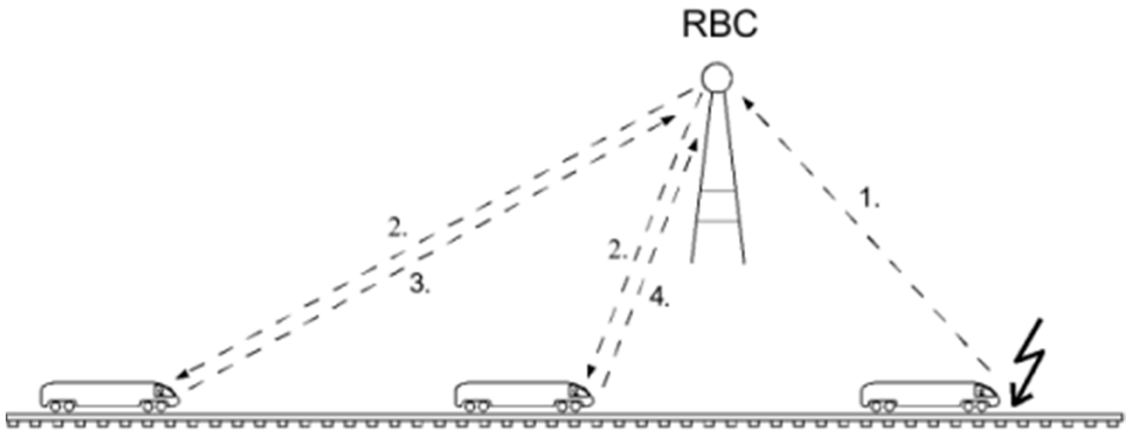
Overview (Part 3)

- Tools and modelling languages
  - PRISM & guarded commands
  - Modest & the Modest Toolset
- Probabilistic timed BRP (PTA)
  - Case Study & Demo
- Temperature control (PHA)
  - Case Study & Demo
- ETCS level 3 train control (SHA)
  - Case Study & Demo

Case Study: ETCS

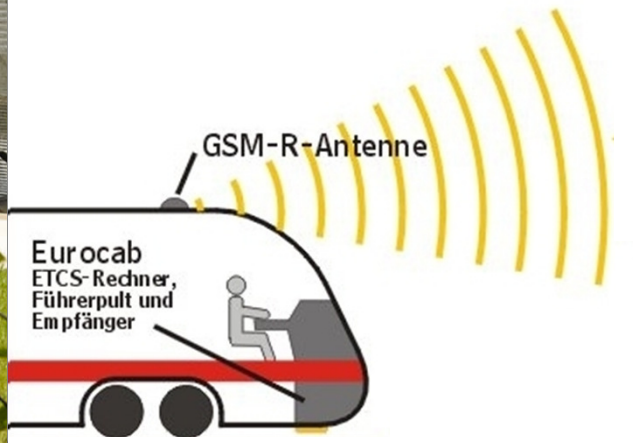
- ETCS Level 3
  - next-generation European train control system
  - moving block train control to increase capacity
  - trains measure and report position to RBC
  - radio block controller (RBC) assigns movement authority
  - communication is wireless

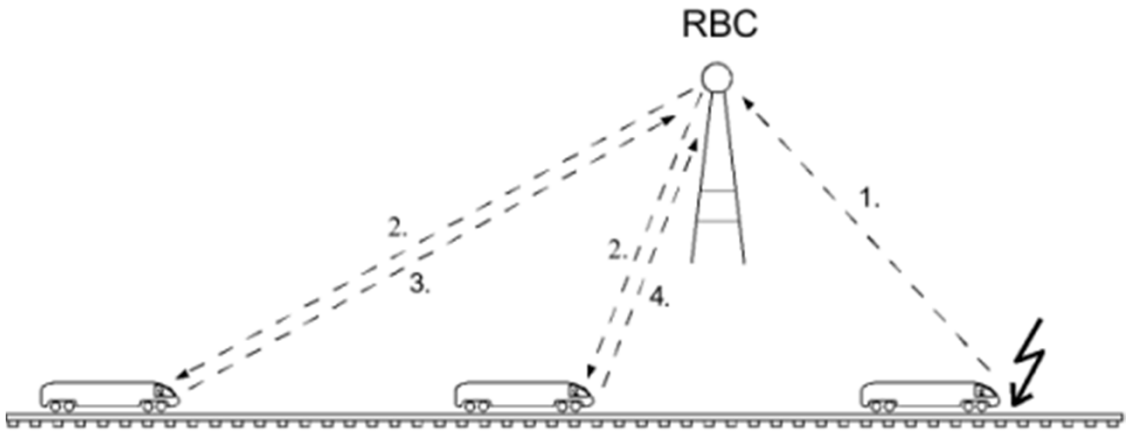




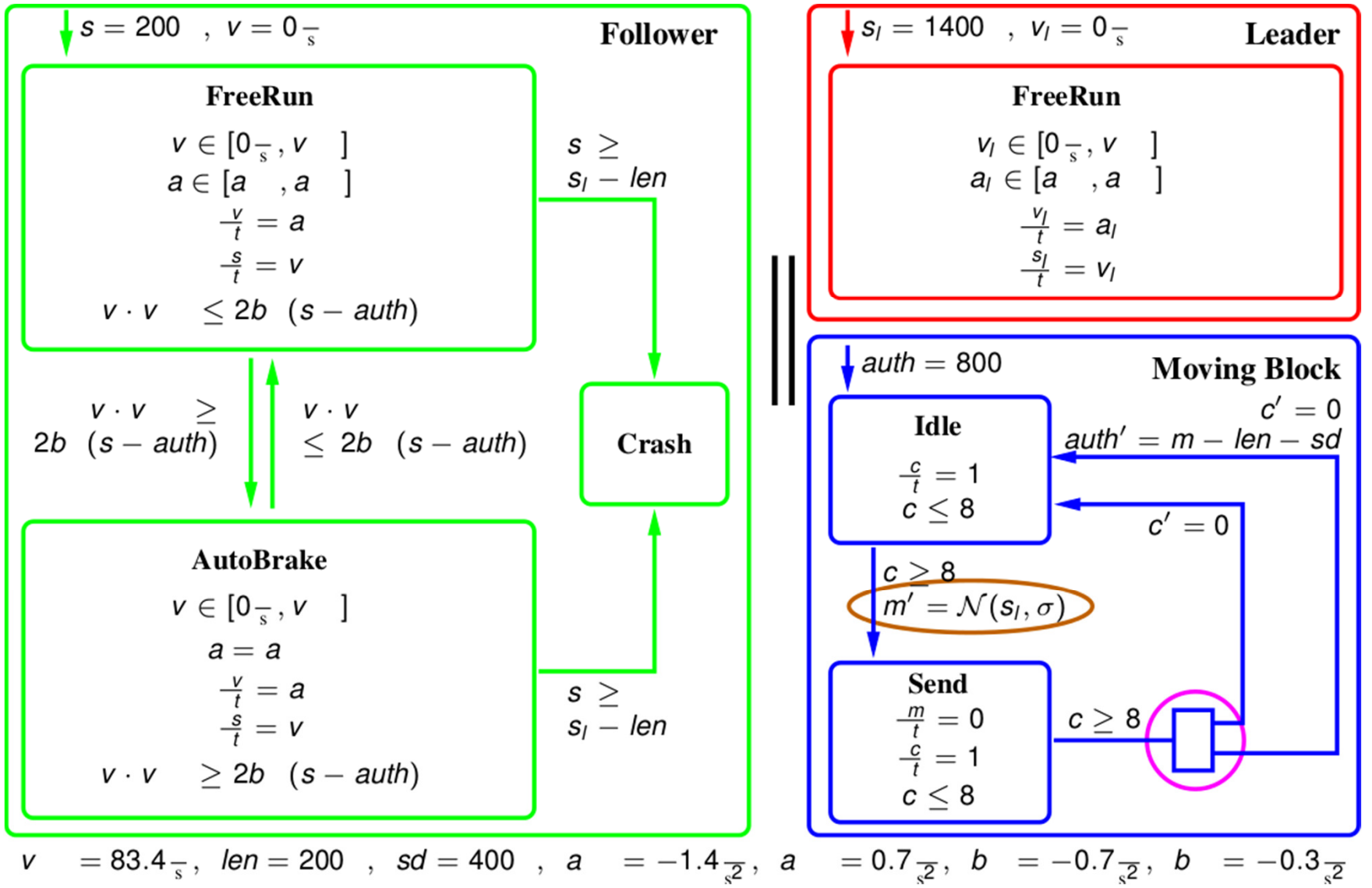
Case Study: ETCS

- SHA model
  - two trains – leader and follower – and Comm+RBC
- Continuous aspects
  - acceleration, deceleration, speed
  - acceleration of leader nondeterministic (within train limits)
- Stochastic aspects
  - position measurements scattered with normal distribution
  - message loss probability during communication





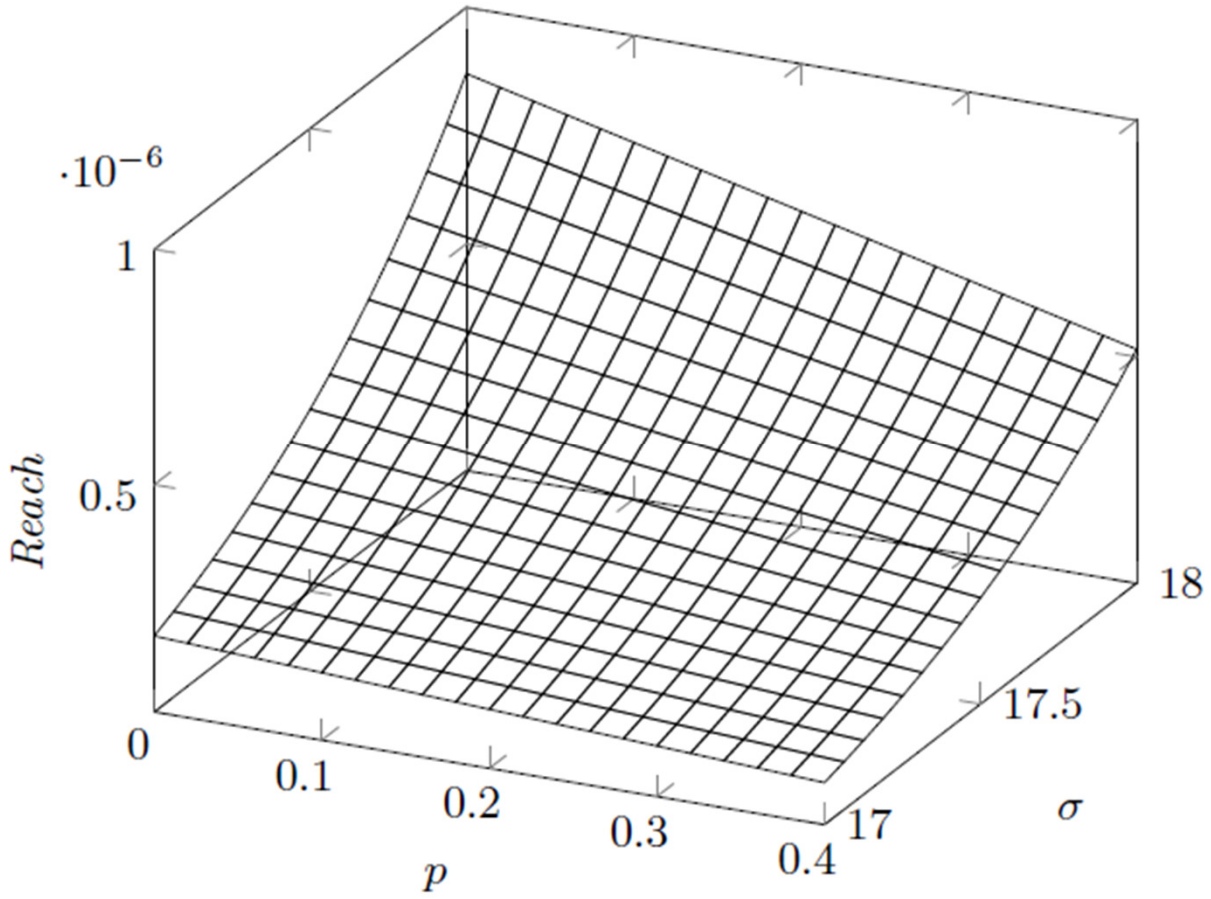
Case Study: ETCS

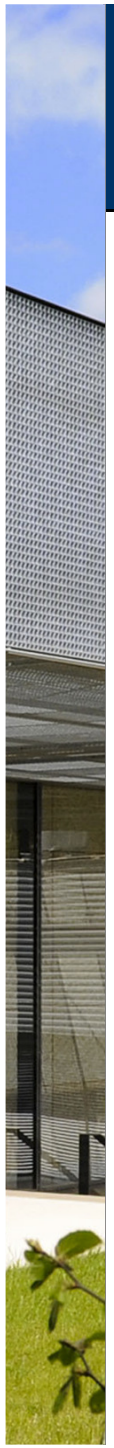


⇨ DEMO

Case Study: ETCS

- Results
  - probability depending on message loss probability (p) and magnitude of measurement error (σ)





Tools – Summary

- PRISM
  www.prismmodelchecker.org
  - modelling language: guarded commands
  - model checking for **PTA** & **MDP**

- The Modest Toolset
  www.modestchecker.net
  - modelling language: Modest + guarded commands
  - prohver for STA (using PHAVer)
  - mcpta for PTA/MDP (using PRISM)
  - mctau for TA (using UPPAAL)
  - modes for statistical model checking

  Modest Toolset demo at poster & demo session tomorrow!