The Quest for Quantifiable Quality

Holger Hermanns

Dependable Systems and Software
Saarland University

and

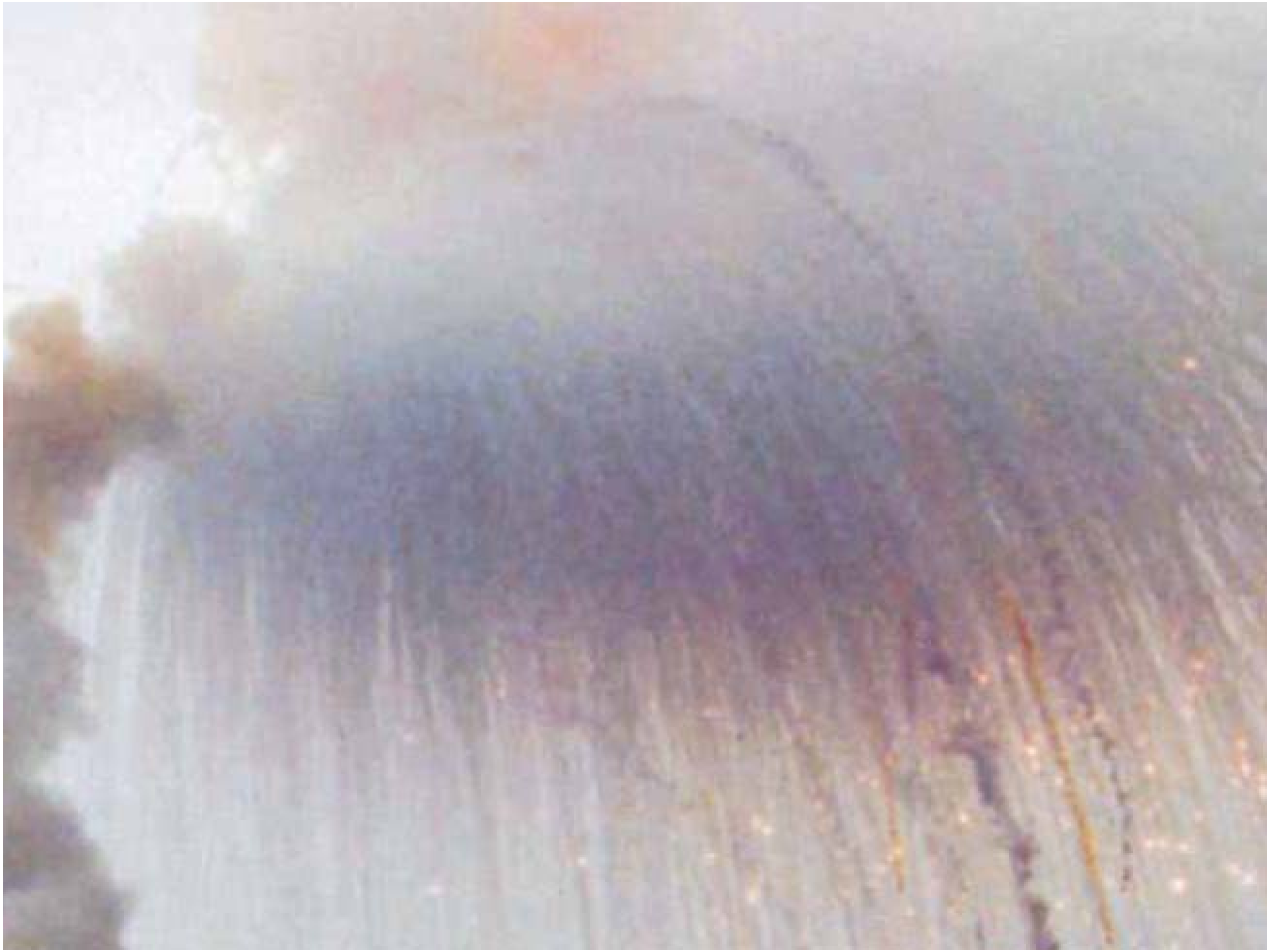
VASY
INRIA Rhône-Alpes

What is this talk about?

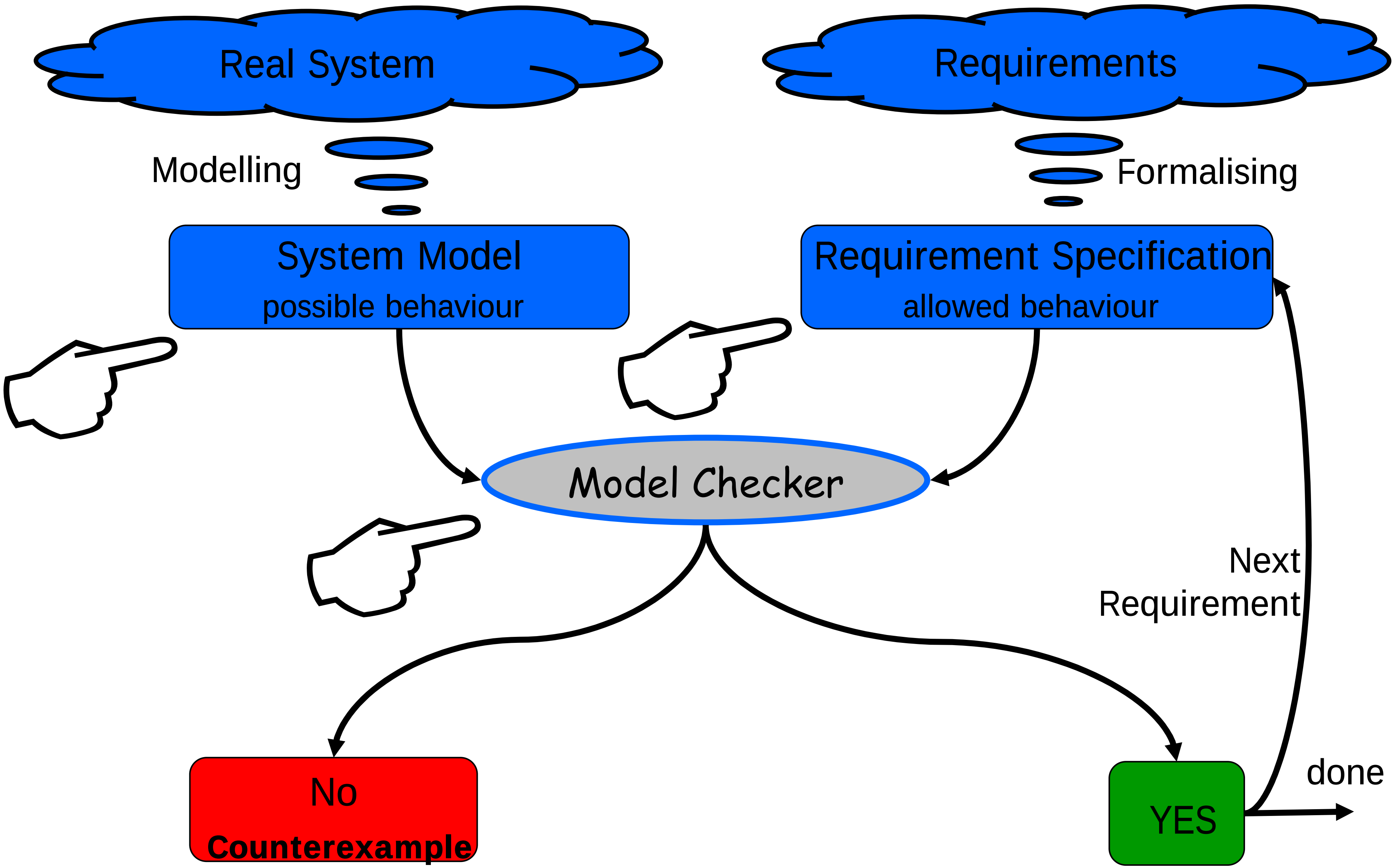
- Guaranteeing properties of computer systems that are not at hand
  - because they are embedded in a physical environment
  - because they are not existing, or not so easy to play with

- Usually systems that are safety-critical, expensive, and/or numerous.

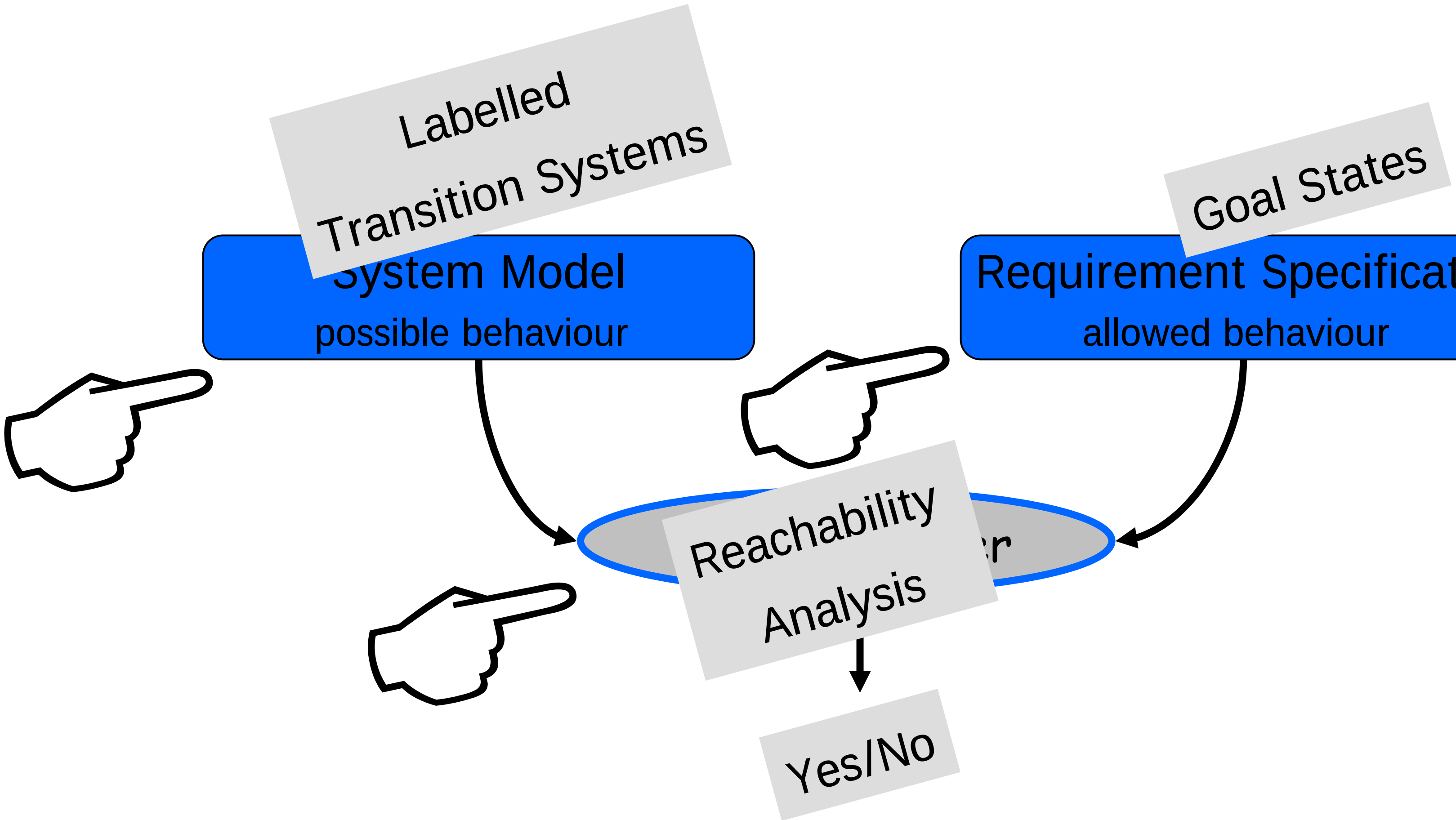
Embedded Systems



Model Checking



Model Checking (the simplest case)



Labelled transition systems

What do we like about this model?

- Is easy to model with
- Fits well to concurrency: Interleaving semantics
  - Message passing,
  - handshake communication, and
  - shared variable communication

  all behave naturally
- Is compositional

Labelled transition systems

What we do not like about this model?

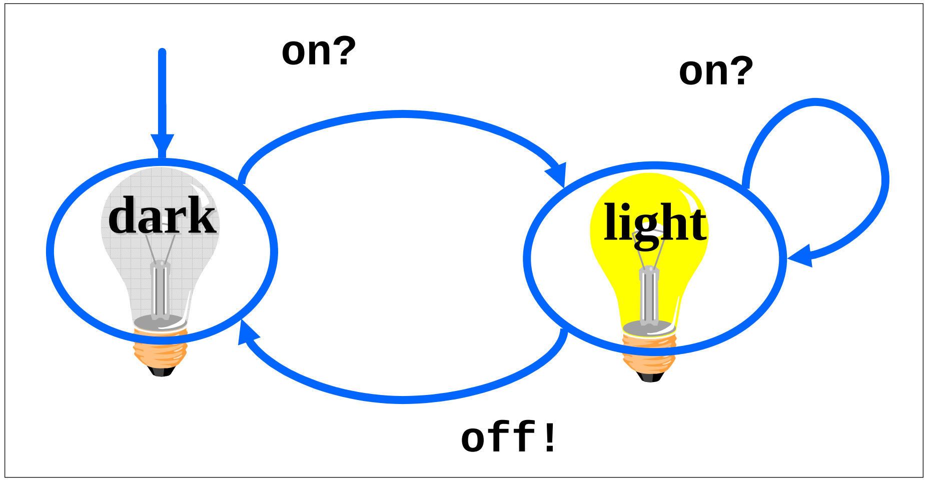
- No support for quantities
  - Time
  - Cost
  - Probabilities

- No support for continuity

- ...

Quantitative Models?

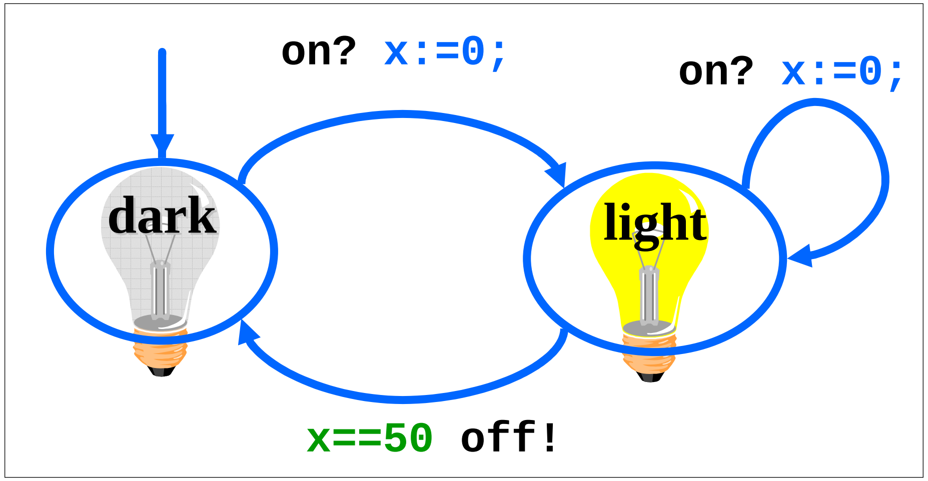
- finite automata



Quantitative Models?

- finite automata
- decorated with clocks

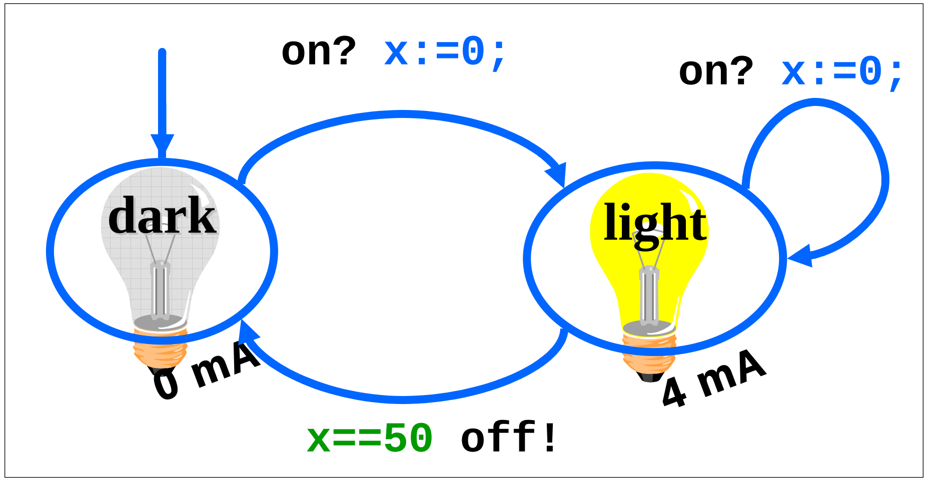
all run at the same speed



Quantitative Models?

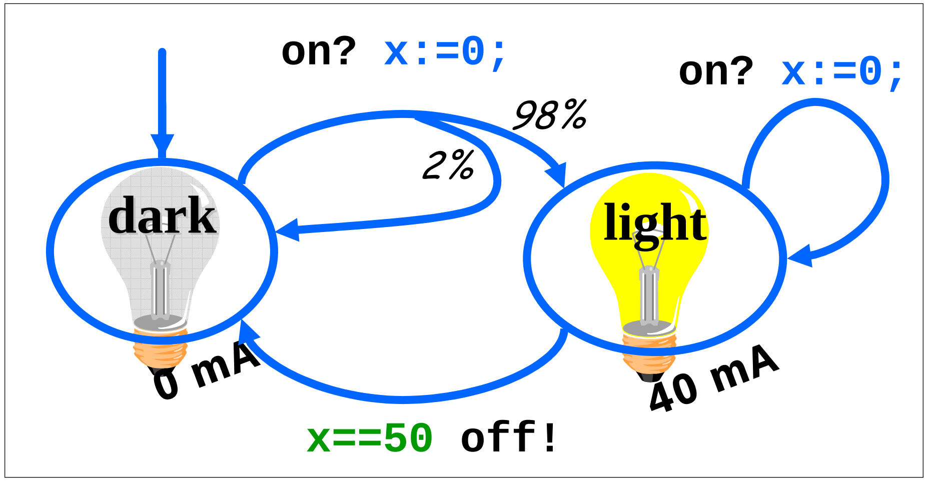
- finite automata
- decorated with clocks
- and with costs

accumulated linear with time



Quantitative Models?

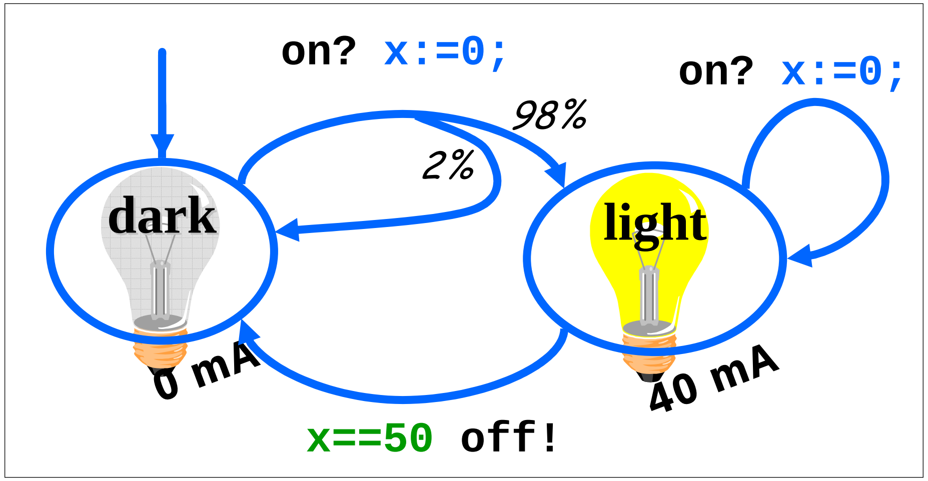
- finite automata
- decorated with clocks
- and with costs
- and with *discrete* probability distributions

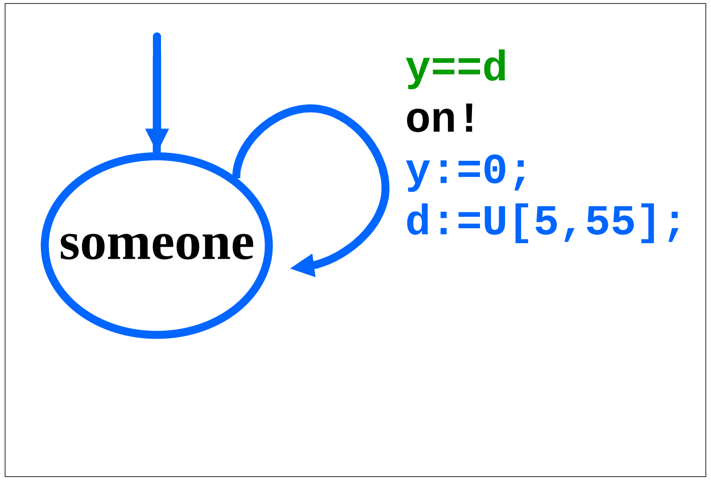


priced Probabilistic Timed Automata (pPTA)

Quantitative Models?

- finite automata
- decorated with clocks
- and with costs
- and with probability distributions





- modular: composition of automata

priced Probabilistic Timed Automata (pPTA)

priced Probabilistic Timed Automata

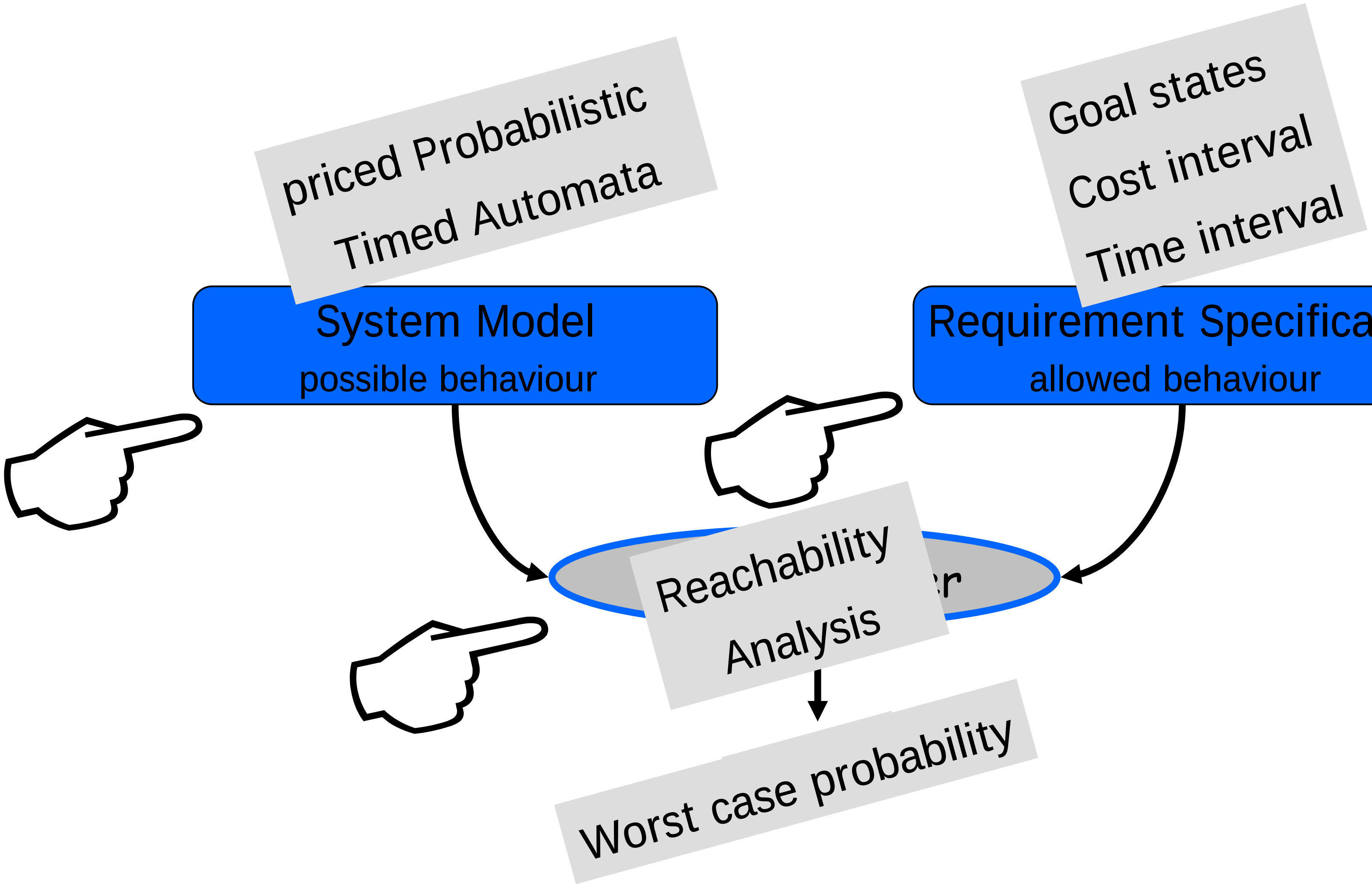
What do we like about this model?

- Is (rather) easy to model with
- Fits somewhat well to concurrency: Interleaving Semantics
  - Message passing,
  - handshake communication, and
  - shared variable communication

all behave naturally

- Synchronicity of clocks is a little awkward.

Quantitative Model Checking



This talk is not about priced Probabilistic Timed Automata

- Anyway, what you may want to know:

UPPAAL

- Reachability for TA is decidable
- Time bounded reachability for TA is decidable [Alur/Dill]

PRISM

- Reachability for PA is decidable
- Hop bounded reachability for PA is decidable [Puterman]

UPPAAL CORA

- Cost bounded reachability for pTA is decidable [Fehnker]
- Time and cost bounded reachability for pTA is decidable [Behrmann et al]

PRISM?

- Reachability for PTA is decidable
- Time bounded reachability for PTA is decidable [Kwiatkowska et al]

(nothing)

- Cost bounded reachability for pPTA is semi-decidable
- Time and cost bounded reachability for pPTA is semi-decidable [Berendsen et al]

infinite zone range

This talk is about an alternative to Probabilistic Timed Automata

Reachability for PA is decidable

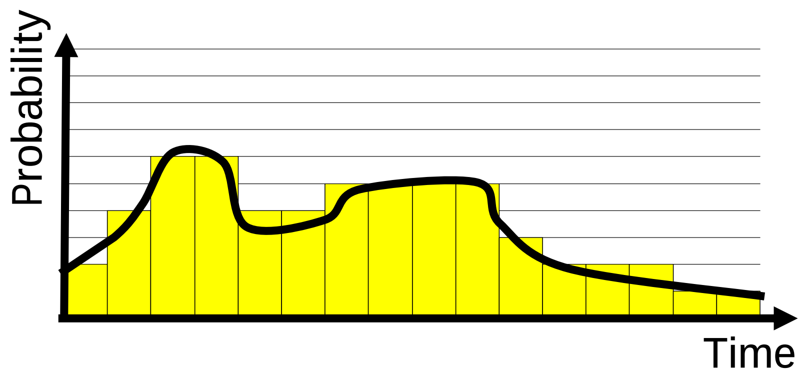
- Hop bounded reachability for PA is decidable [Puterman]

- State-of-the-art in PTA model checking: Discretize time and check reachability [Kwiatkowska et al] [Kattenbelt][Daws]

Reachability for PTA is decidable

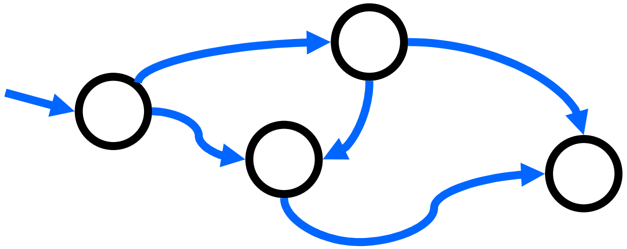
- Time bounded reachability for PTA is decidable [Kwiatkowska et al]

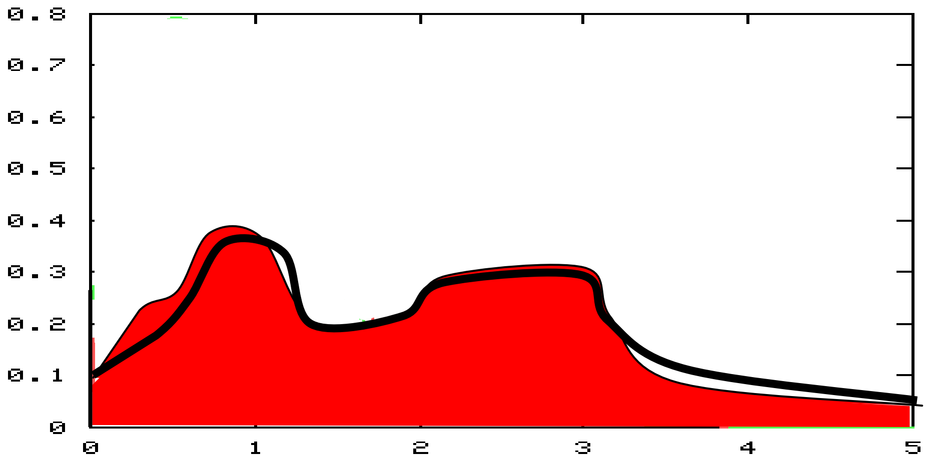
- Double discretisation:



Approximation in continuous time

Memoryless sojourn times, but memorization of phase of the distribution

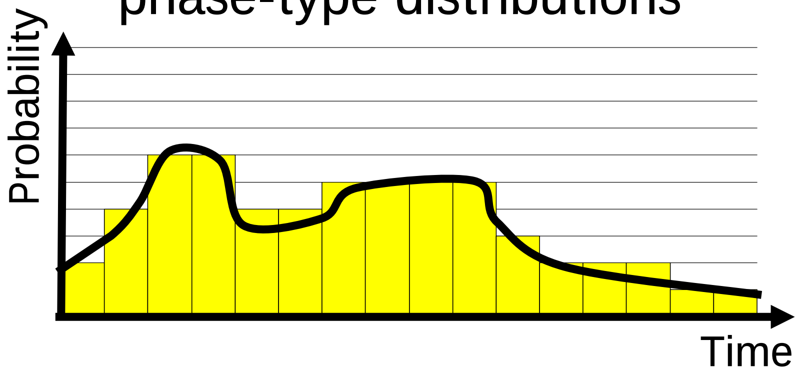




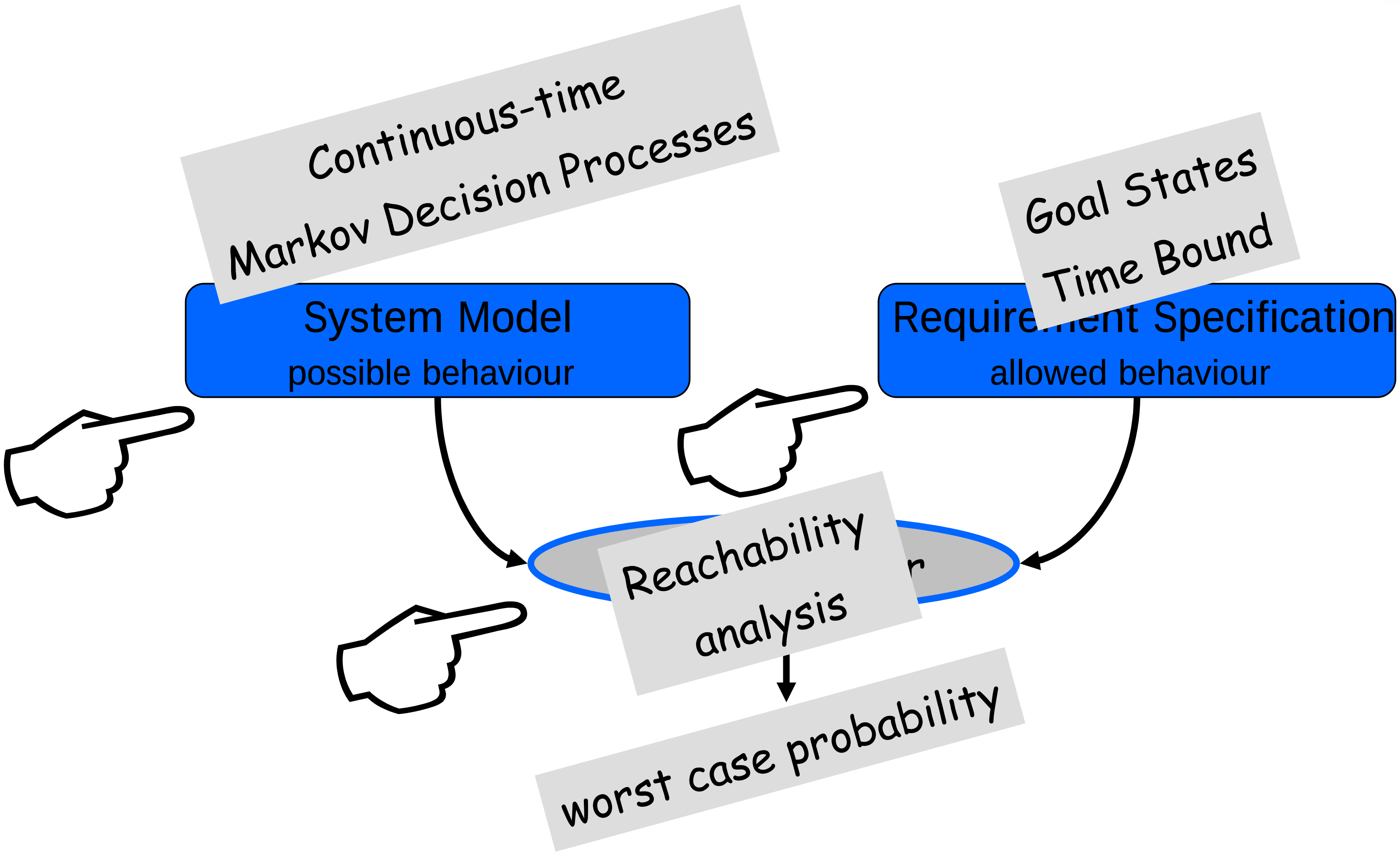
- Recall: double discretisation

- Phase-type distributions form a dense set of continuous distributions

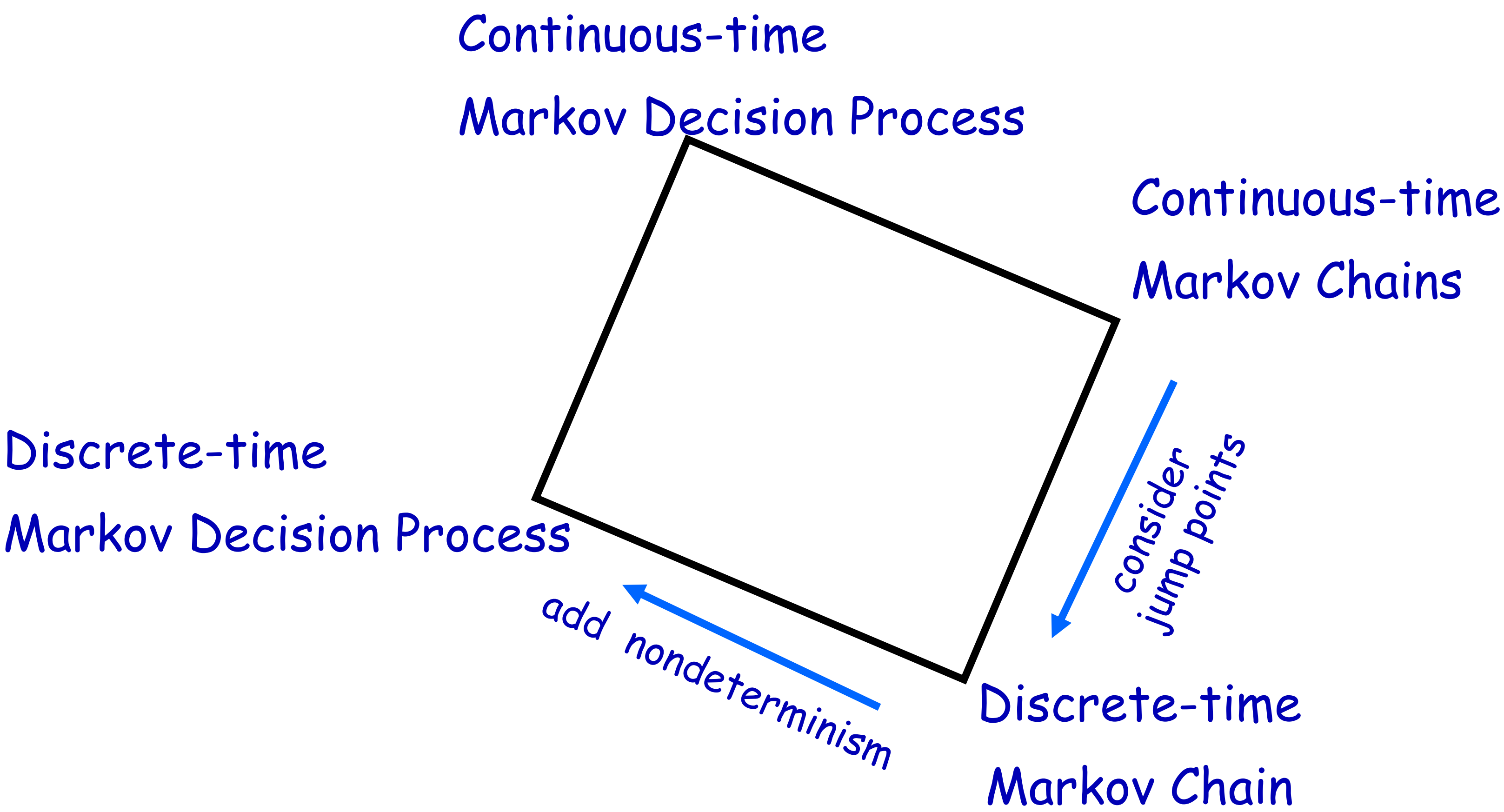
- superpositions can approximate arbitrary continuous distributions: phase-type distributions



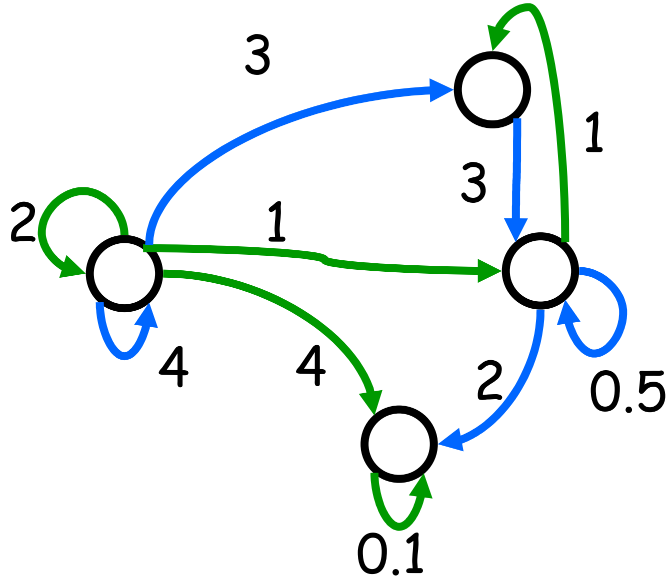
Model Checking



Markov Chains and Decision Processes



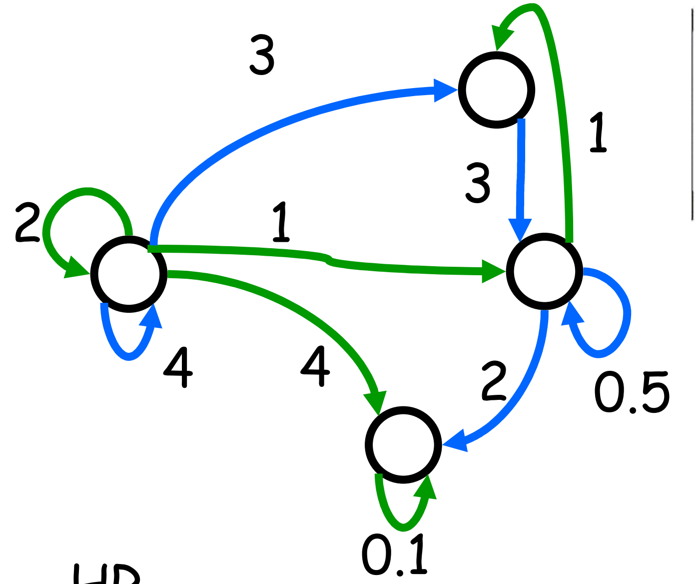
Continuous-Time Markov Decision Processes: CTMDPs



A tuple $\mathcal{M} = (S, Act, \mathbf{R})$, where

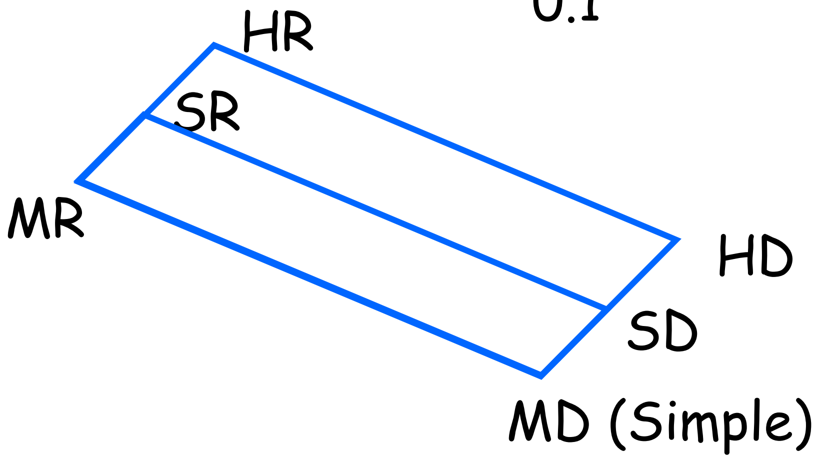
- S is a finite set of states,
- Act a finite set of actions,
- $\mathbf{R} : (S \times Act \times S) \rightarrow \mathbb{R}_{\geq 0}$ is the (three-dimensional) rate matrix.

Scheduler types

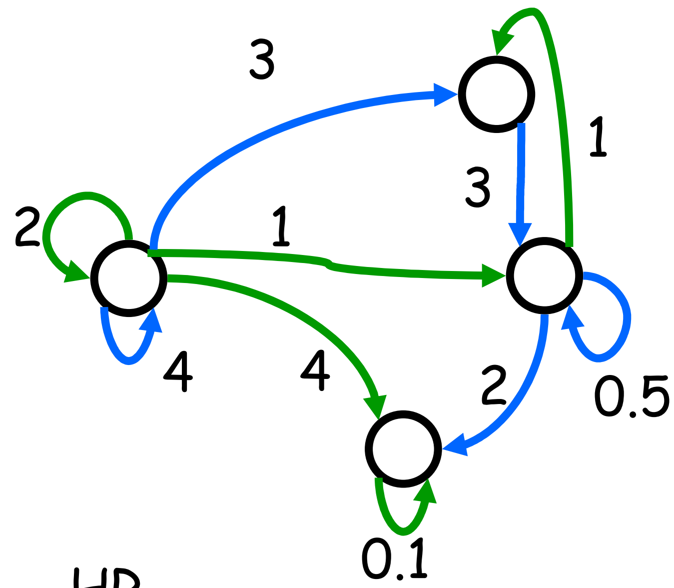


- H: History dependent
- S: Step dependent
- M: Markov (History independent)

- R: Randomized
- D: Deterministic

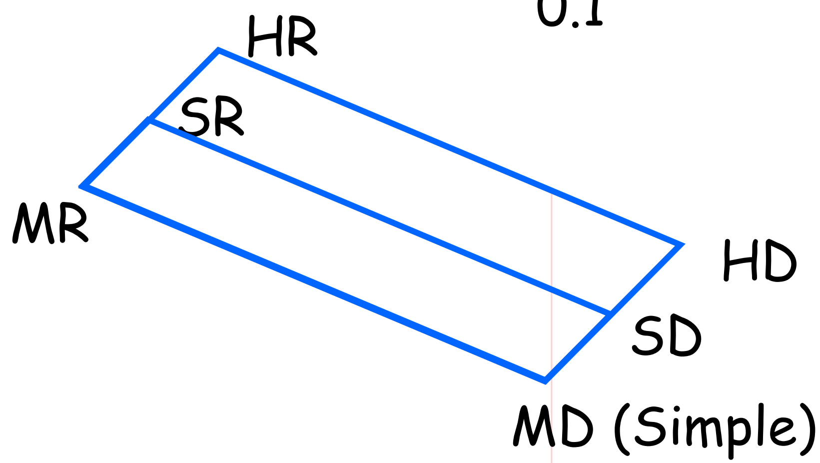


Maximum timed reachability probability



We look for the maximum probability to reach a set of goal states within a given time interval:

$$\sup_{D \in Sched} \Pr_D(s, \overset{\leq t}{\rightsquigarrow} B)$$

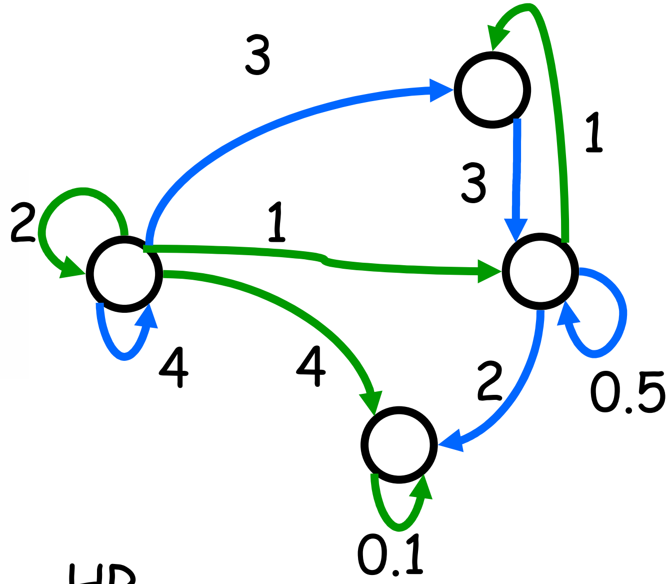


where $t > 0$ is a real time-bound, $B \subseteq S$ and $s \in S$.

And $Sched$ is the class of schedulers considered.

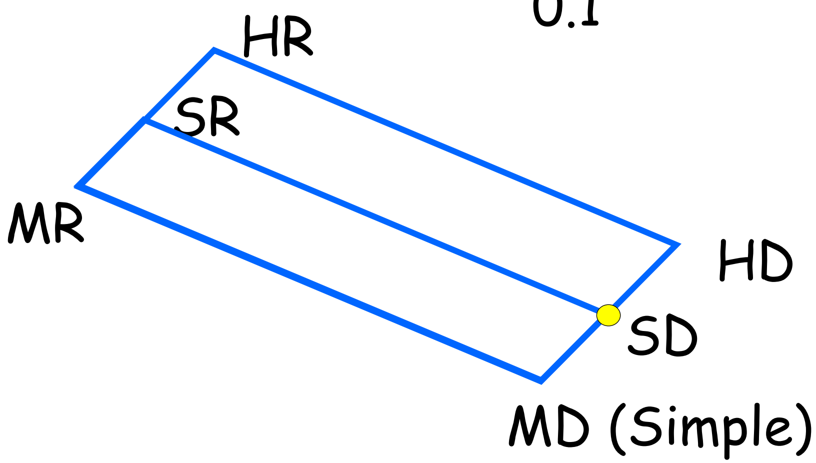
How to compute

$$\sup_{D \in Sched} \Pr_D(s, \overset{\leq t}{\rightsquigarrow} B)$$



We

- look at SD schedulers,
- and restrict to uniform CTMDPs.



Then we'll show that this cannot be outperformed by adding

- more History,
- or Randomization.

A CTMDP $\mathcal{M}$ is called uniform if there is some E such that for all states s, $\alpha \in Act(s)$ implies $E(s, \alpha) = E$.

A greedy backward algorithm (for truncated SD-schedulers)

Idea:

- Consider truncated SD-schedulers

$$D : S \times \{1 \ldots k\} \rightarrow Act$$

- Try to construct the optimal one.
- To do so, use a greedy backwards strategy:
  - For each state determine the action that maximises the probability to reach set B in one step.
  - Use these actions to calculate the last (the k -th) summand of

$$\left(\sum_{n=0}^{k} \pi(n) \cdot \mathbf{P}^{n}_{D,B} \cdot \mathbf{i}_B \right)(s)$$

  - Continue this way.

  - Make sure that the resulting schedule is of interest.

Optimality for HD and HR schedulers

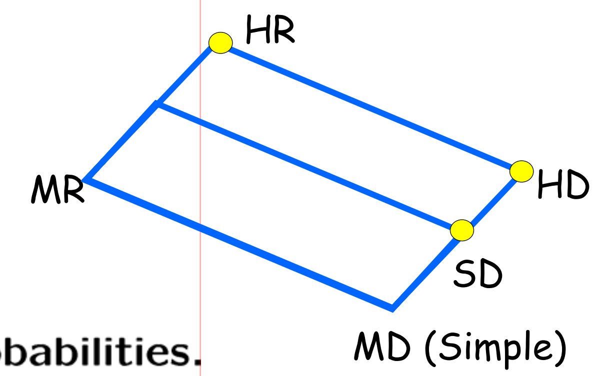
Optimality of the returned vector q up to ε.

For all states $s \in S$:

$$\sup_{D \in HD} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B) \;-\; \varepsilon \;\leq\; q(s) \;\leq\; \sup_{D \in HD} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B)$$

Suprema under HD and SD agree for accumulated probabilities.

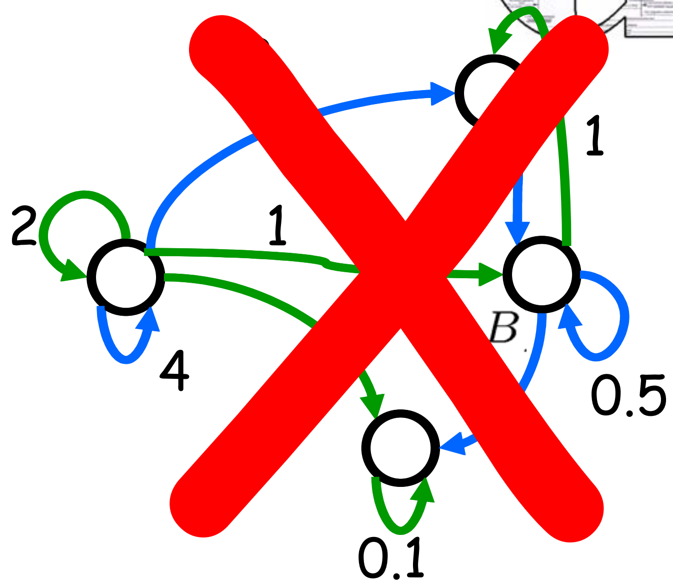
$$\sup_{D \in SD} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B) \;=\; \sup_{D \in HD} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B)$$



Suprema under HD and HR agree for accumulated probabilities.

$$\sup_{D \in HD} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B) \;=\; \sup_{D \in HR} \Pr_D(s \overset{\leq t}{\rightsquigarrow} B)$$

Complexity for uniform CTMDPs



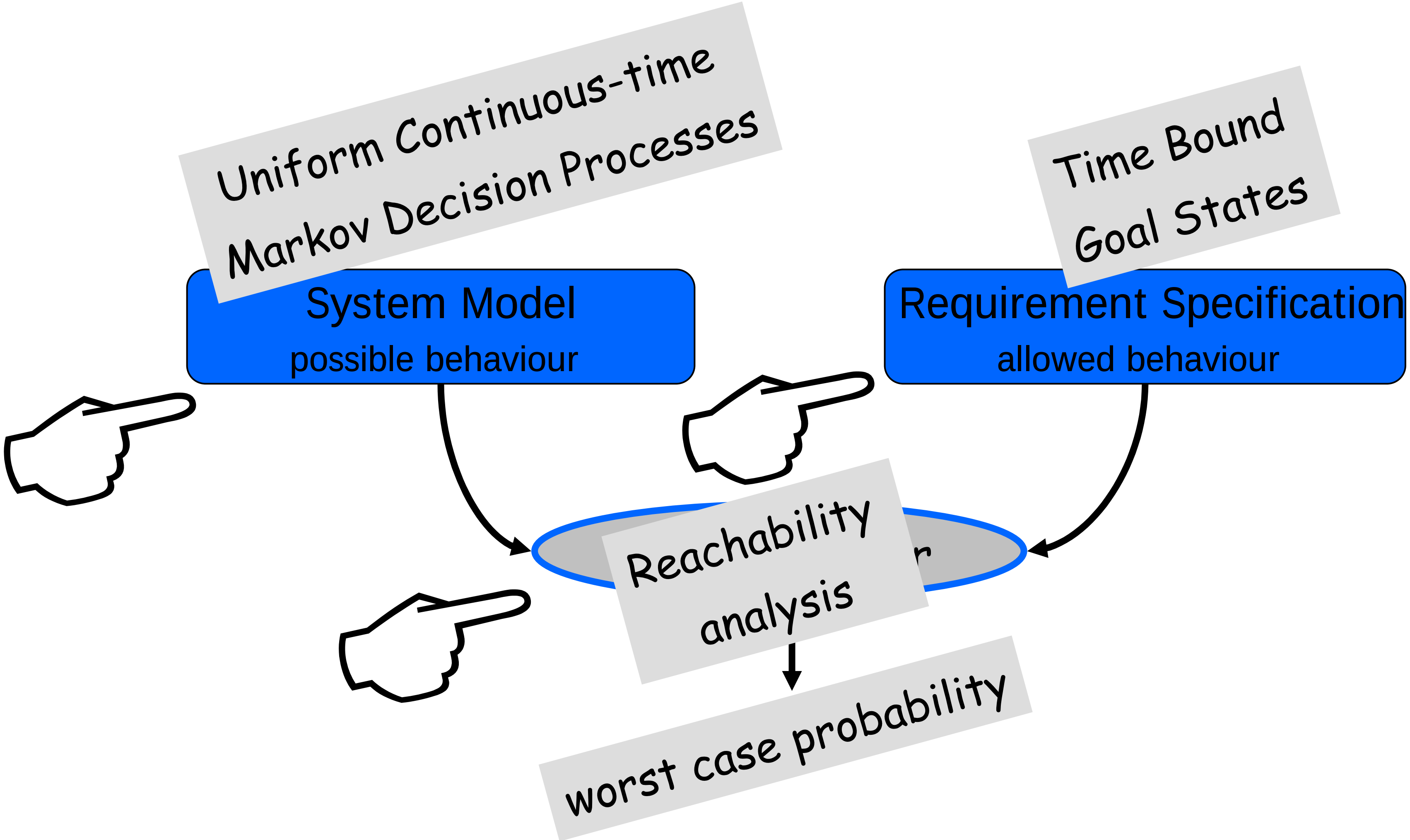
Space complexity: $\mathcal{O}\left(|S|^2 \cdot |Act|\right)$

Time complexity: $\mathcal{O}\left(E \cdot t \cdot |S|^2 \cdot |Act|\right)$

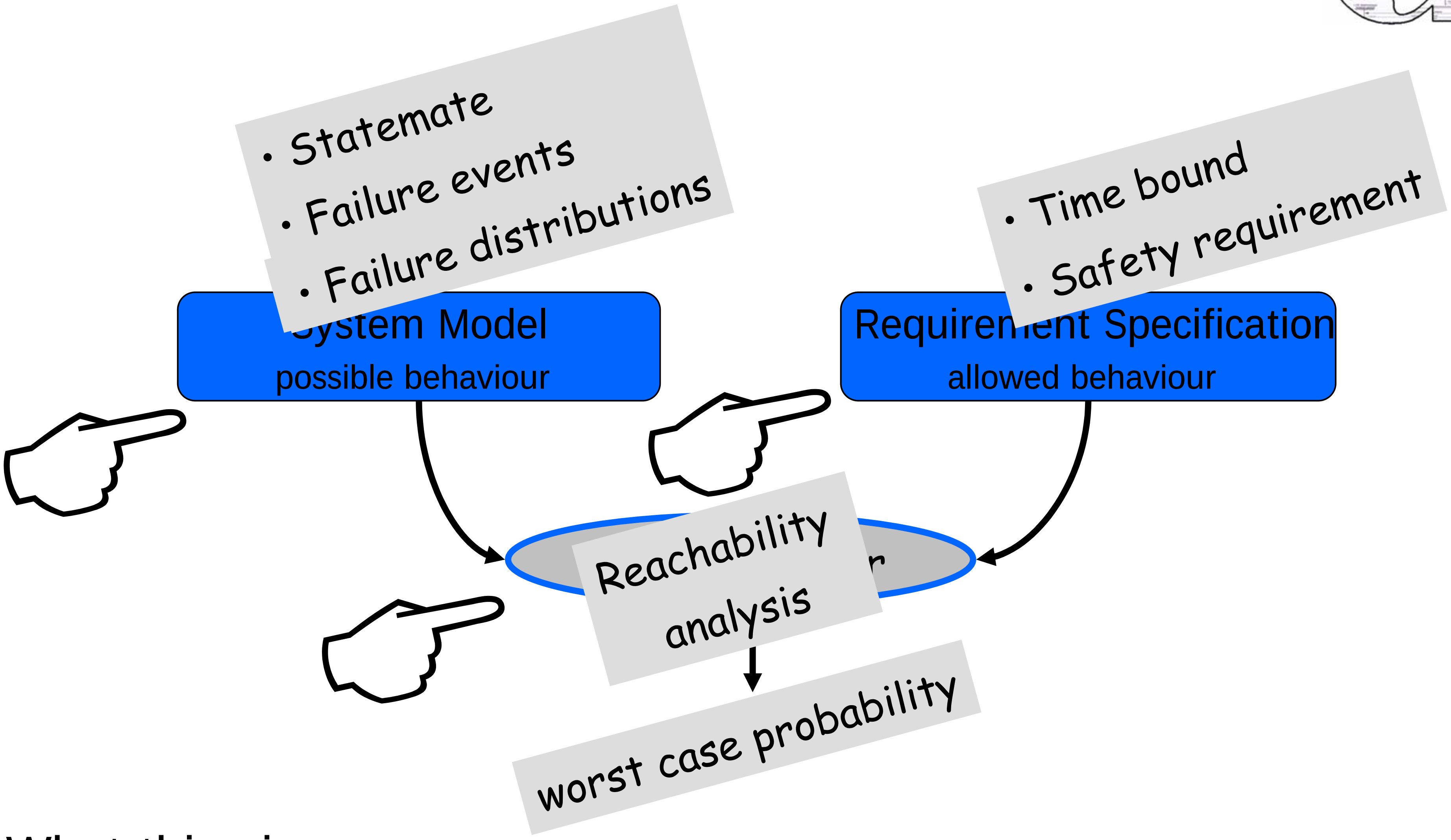
A CTMDP $\mathcal{M}$ is called uniform
if there is some E such that for all states s,
$\alpha \in Act(s)$ implies $E(s, \alpha) = E$.

[Baier et al]

Model Checking, what we have

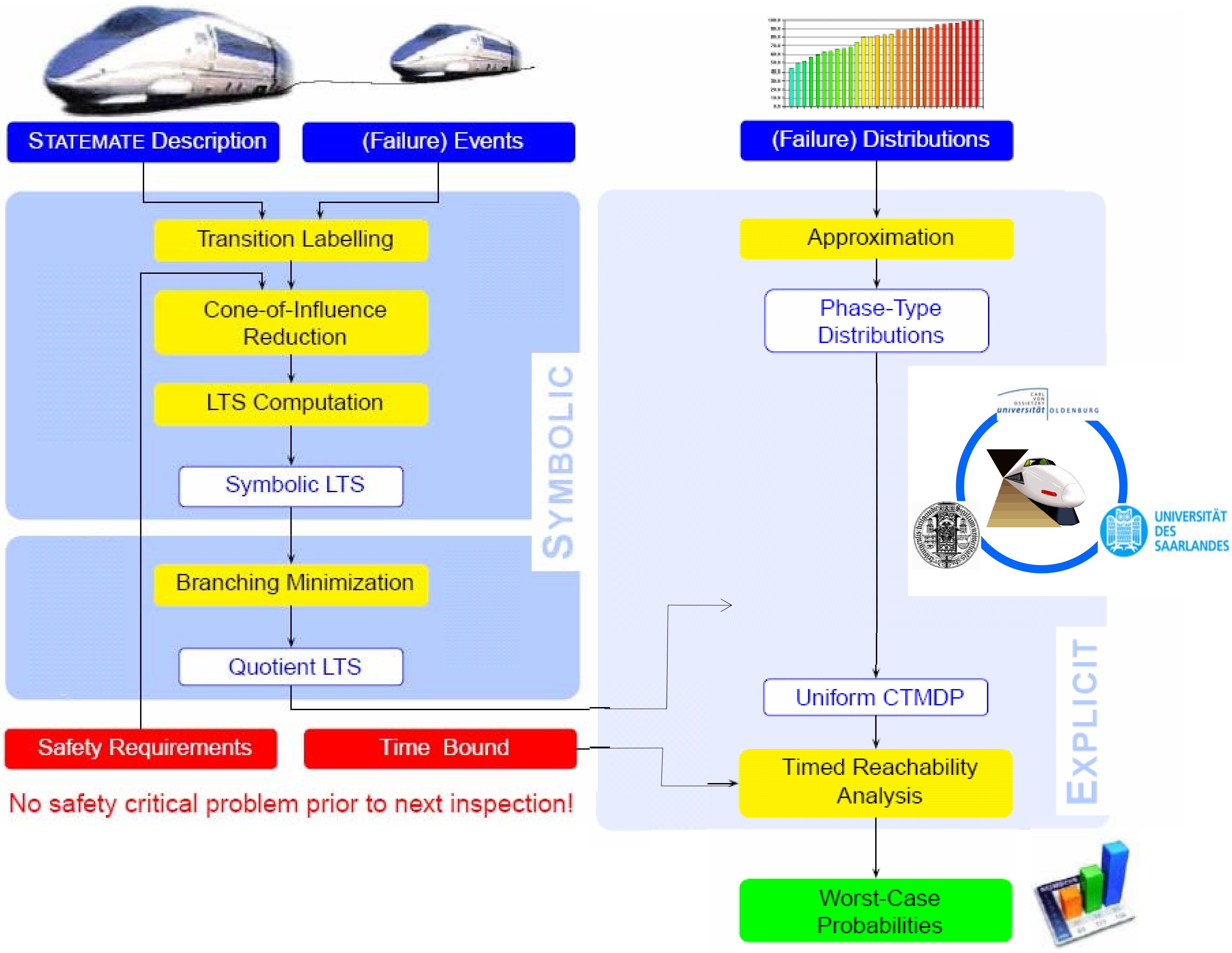


Model Checking, what we also have

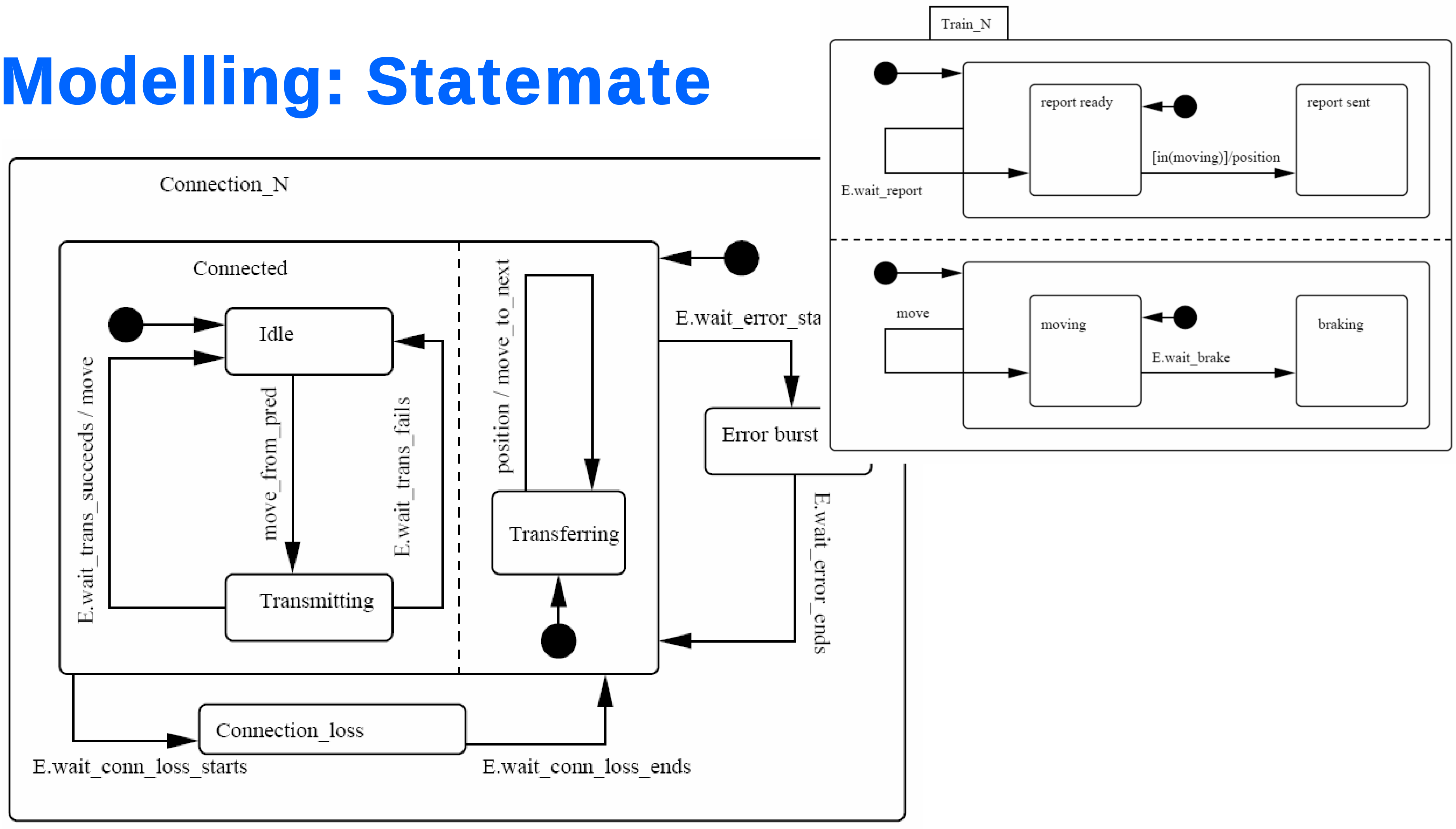


What this gives us:

the worst case probability to hit an unsafe situation within a given time bound


STATEMATE Description
(Failure) Events
(Failure) Distributions
Transition Labelling
Approximation
Cone-of-Influence Reduction
Phase-Type Distributions
LTS Computation
Symbolic LTS
SYMBOLIC
Branching Minimization
Quotient LTS
Uniform CTMDP
Safety Requirements
Time Bound
Timed Reachability Analysis
EXPLICIT
No safety critical problem prior to next inspection!
Worst-Case Probabilities
UNIVERSITÄT DES SAARLANDES

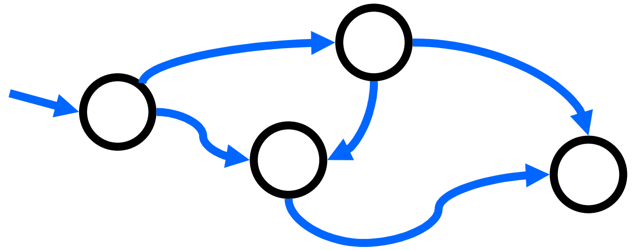
Modelling: Statemate



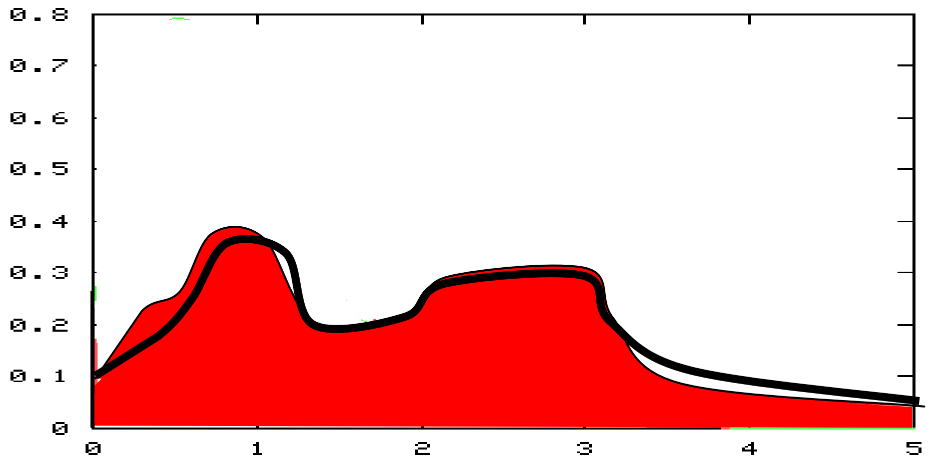
- Hierarchical, state-transition oriented specifications of reactive systems.
- Underlying: a finite-state transition system.

Failure modes and general distributions

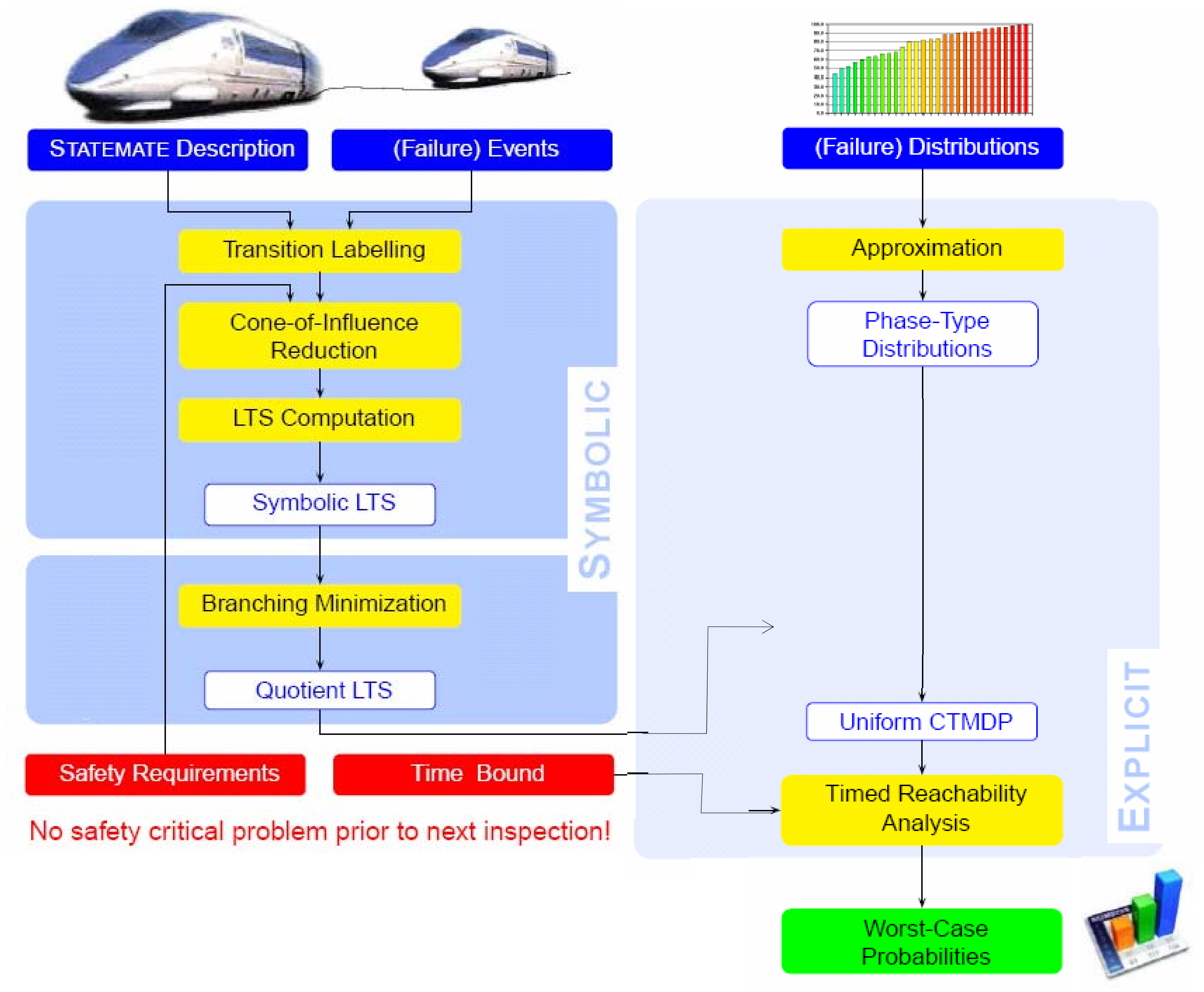
- Failure modes alllow a 'fault injection'-style behavioural extension of the TS underlying a Statechart.
- Available in Statemate.

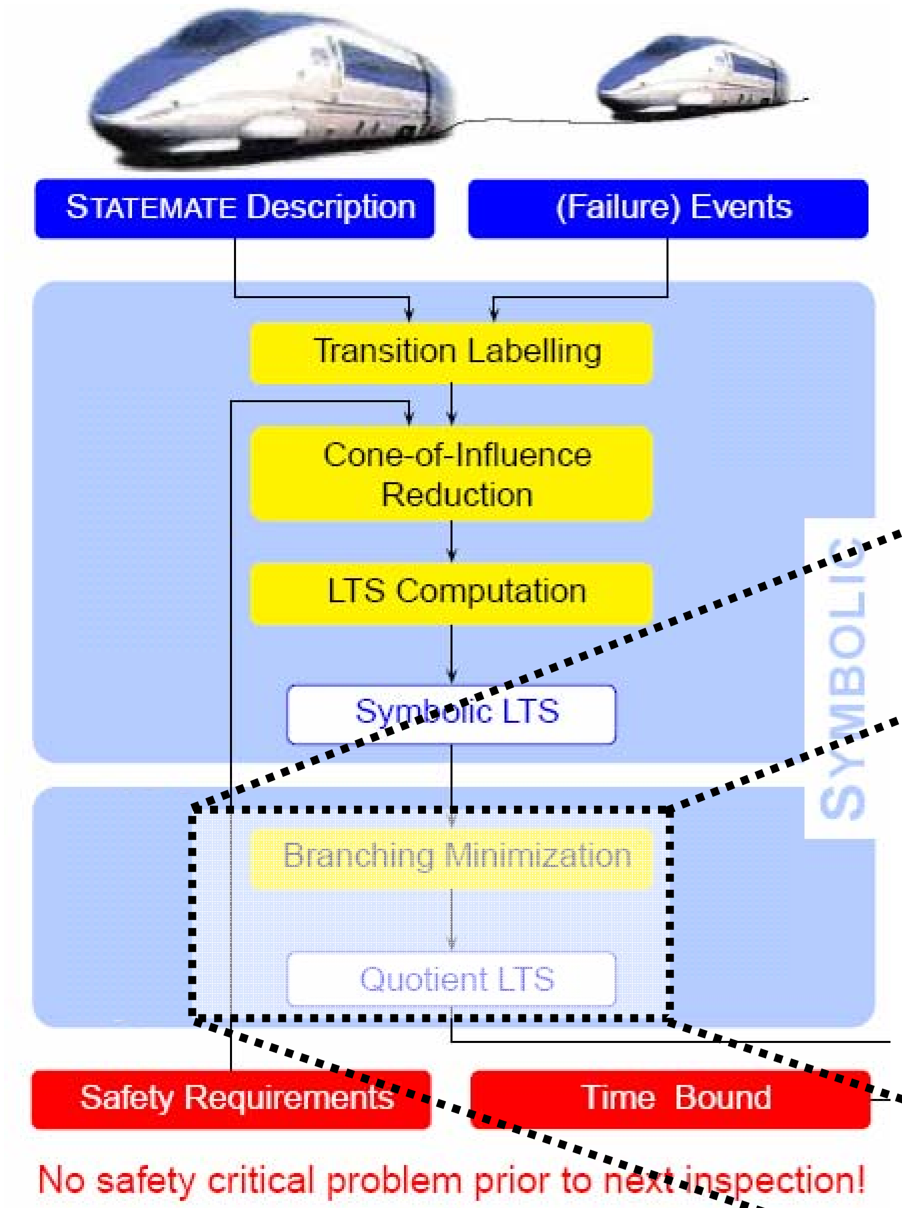


- We enable to delay the occurrence of these failures by *general* probability distributions.
- Those are approximated by Phase-Type distributions.



- And integrated by composition.


STATEMATE Description
(Failure) Events
(Failure) Distributions
Transition Labelling
Cone-of-Influence Reduction
LTS Computation
Symbolic LTS
SYMBOLIC
Branching Minimization
Quotient LTS
Approximation
Phase-Type Distributions
Uniform CTMDP
EXPLICIT
Safety Requirements
Time Bound
Timed Reachability Analysis
No safety critical problem prior to next inspection!
Worst-Case Probabilities

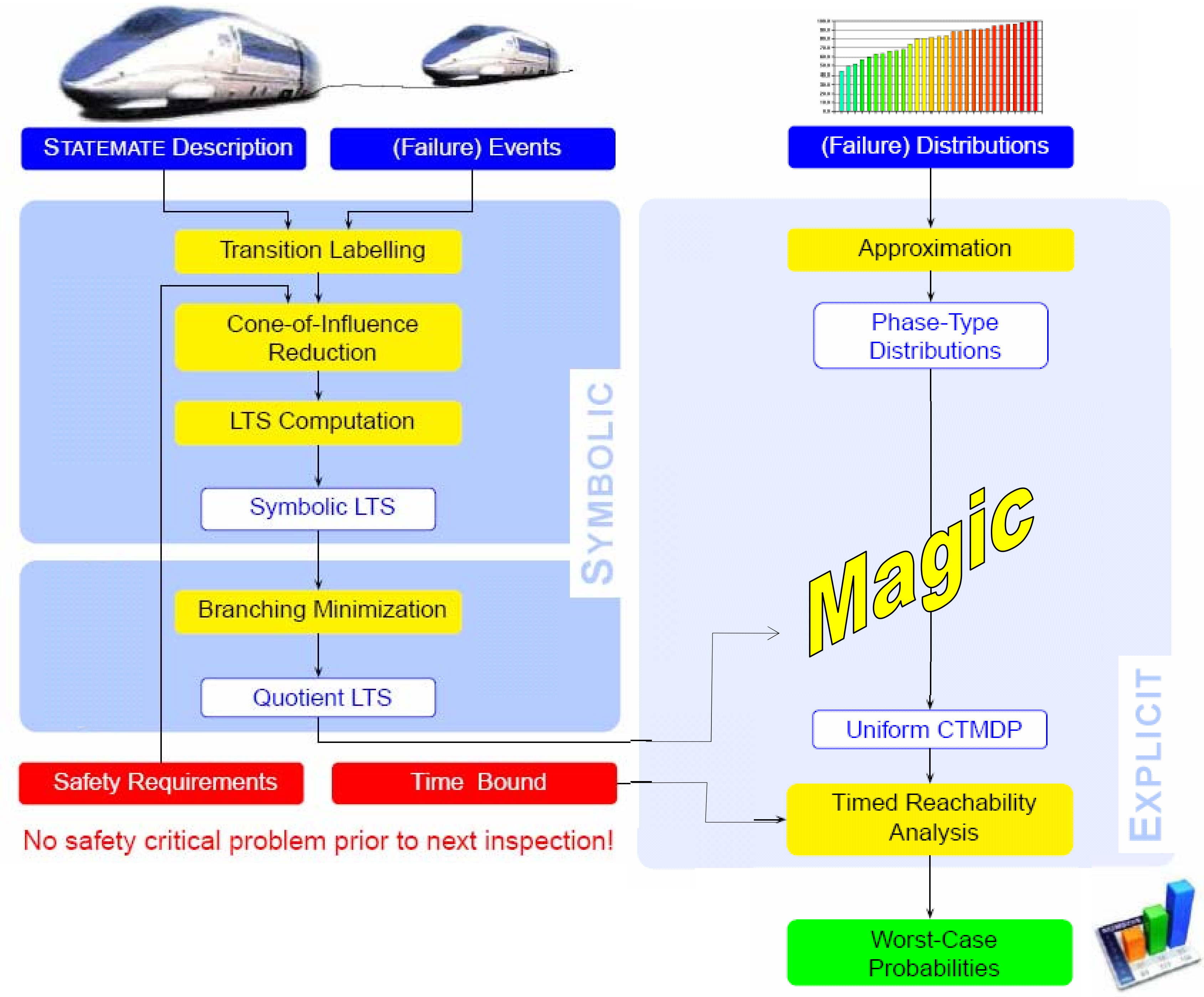


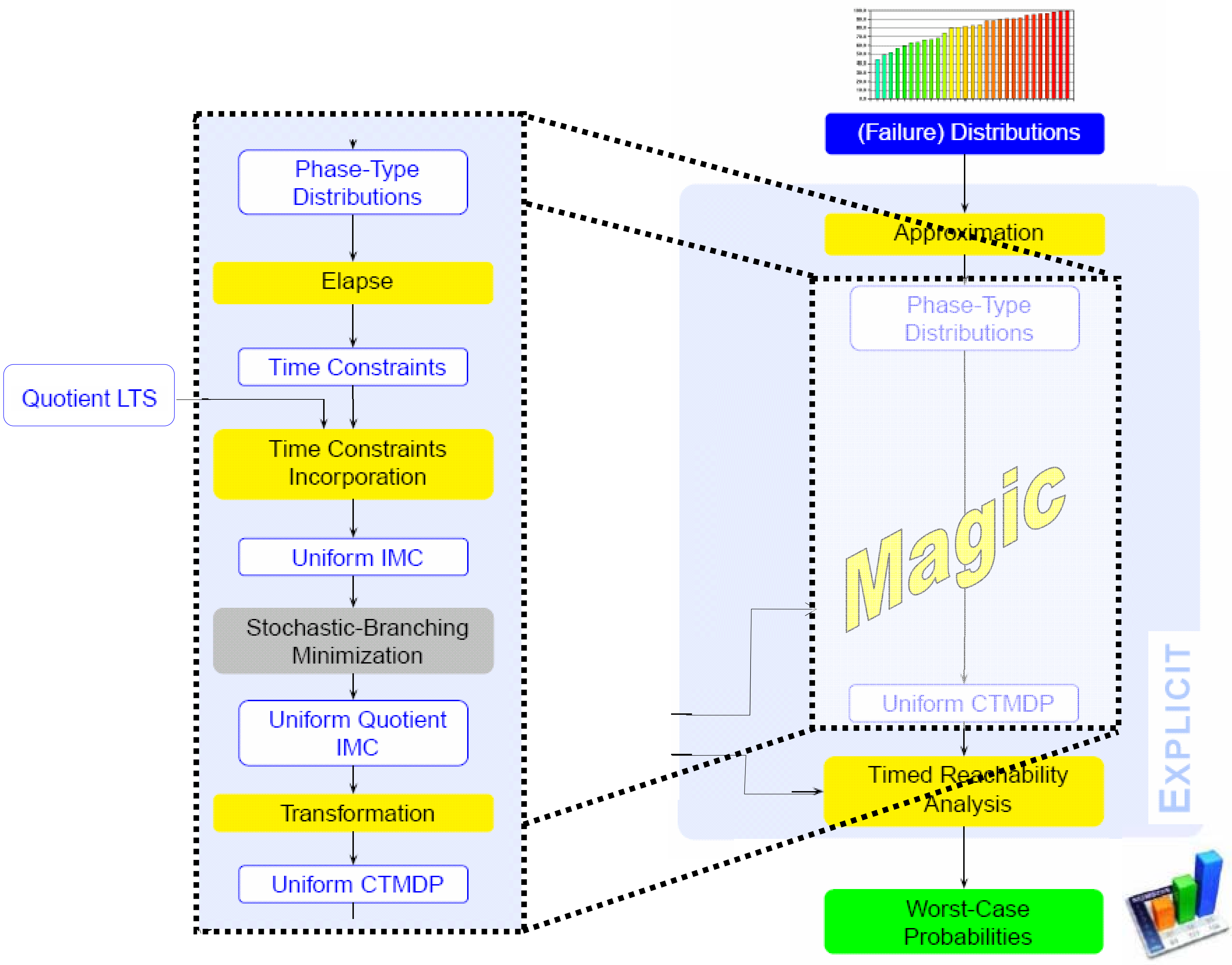
Sigref

- symbolic
- signature based
- partition refinement

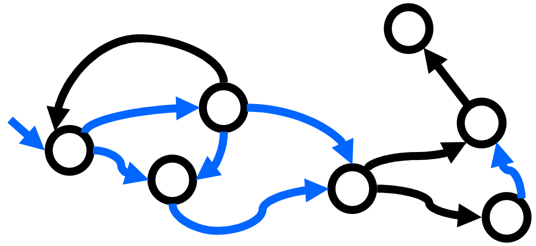
toolbox for various bisimulations

[Wimmer]


STATEMATE Description
(Failure) Events
(Failure) Distributions
Transition Labelling
Cone-of-Influence Reduction
LTS Computation
Symbolic LTS
Branching Minimization
Quotient LTS
SYMBOLIC
Approximation
Phase-Type Distributions
Magic
Uniform CTMDP
Timed Reachability Analysis
EXPLICIT
Safety Requirements
Time Bound
No safety critical problem prior to next inspection!
Worst-Case Probabilities


Phase-Type Distributions
Elapse
Time Constraints
Quotient LTS
Time Constraints Incorporation
Uniform IMC
Stochastic-Branching Minimization
Uniform Quotient IMC
Transformation
Uniform CTMDP
(Failure) Distributions
Approximation
Phase-Type Distributions
Magic
Uniform CTMDP
Timed Reachability Analysis
EXPLICIT
Worst-Case Probabilities

Interactive Markov chains



Model level

- An orthogonal extension
  - of labelled transition systems
  - of CTMCs
- two types of transititions
  - → bla
  - → ν

  in the state space
- equipped with semantic equivalence notions

Syntax level

- A super-algebra
  - of standard process algebra
  - of CTMC algebra
- two types of ‘actions’
  - actions
  - Markov delays

  in the specification
- equipped with the necessary compositional theory

Interactive Markov Chains

What do we like about this model?

- Is (rather) easy to model with
- Fits well to concurrency: Interleaving Semantics
  - Message passing,
  - handshake communication, and
  - shared variable communication

  all behave naturally
- Is compositional
- Has `state-local´ clocks (if at all)

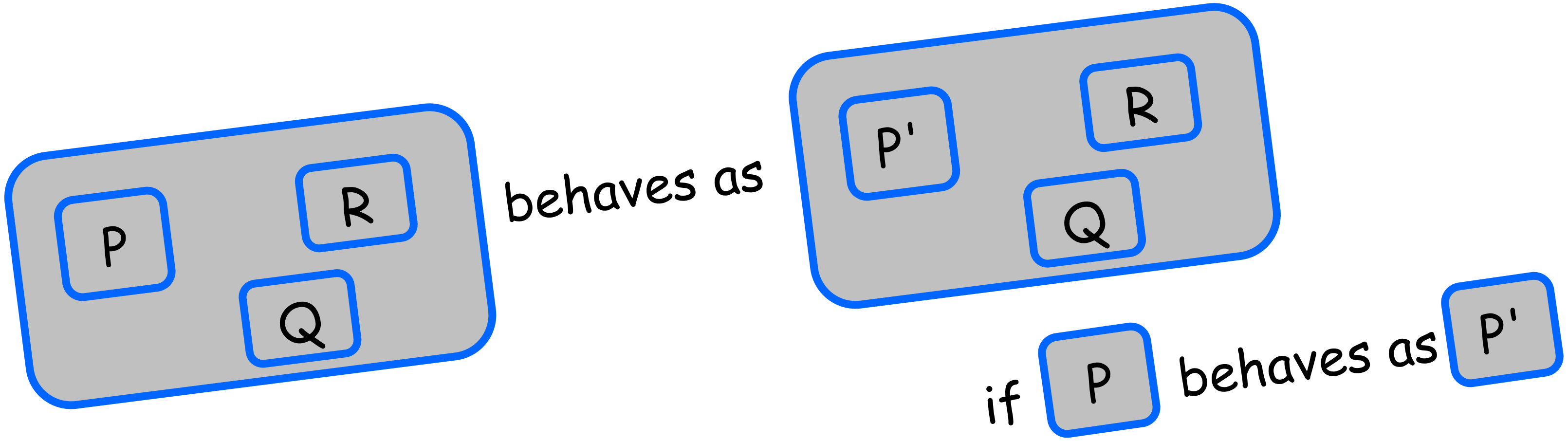
Correspond to CTMDPs

Semantic 'workhorse'

Interactive Markov chains

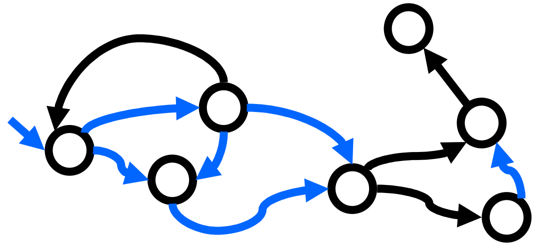
- Orthogonal superposition of
  - automata (or LTS), and
  - Continuous Time Markov Chains,
- with a compositional semantics,
- and *substitutive* notions of equivalence

heavily exploited
in the construction
(compositional minimisation)



What is the relation to CTMDPs? How about uniformity?

IMCs to CTMDPs



An on-the-fly construction:

- cut according to maximal progress assumption,
- split sequences of Markov transitions,
- transitive closure of action transitions.

Let IMC $\mathcal{M}$ be given and let $\mathcal{C}$ be its underlying CTMDP.

Theorem 7.1 Given scheduler D over IMC $\mathcal{M}$, it holds for all traces $\beta \in Words_L$ of CTMDP $\mathcal{C}$ that there exists scheduler D' over $\mathcal{C}$ such that

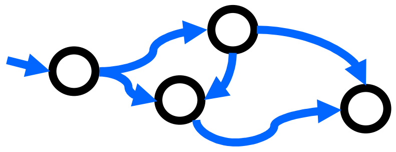
$$Pr^{\omega}_{\mathcal{M},D}\left(C_{trace^{-1}_{\setminus\top}(\beta)}\right) = Pr^{\omega}_{\mathcal{C},D'}\left(C_{trace^{-1}(\beta)}\right) .$$

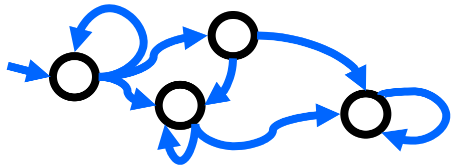
Theorem 7.2 Given scheduler D over CTMDP $\mathcal{C}$, it holds for all traces $\beta \in Words_L$ of CTMDP $\mathcal{C}$ that there exists scheduler D' over $\mathcal{M}$ such that

$$Pr^{\omega}_{\mathcal{C},D}\left(C_{trace^{-1}(\beta)}\right) = Pr^{\omega}_{\mathcal{M},D'}\left(C_{trace^{-1}_{\setminus\top}(\beta)}\right) .$$

[Johr]

Uniformity





- Uniformisation is a CTMC transformation that preserves its probabilistic behaviour, but makes the jumps occur at Poisson intervals.

- We apply this to the input CTMCs, modelling PH distributions.

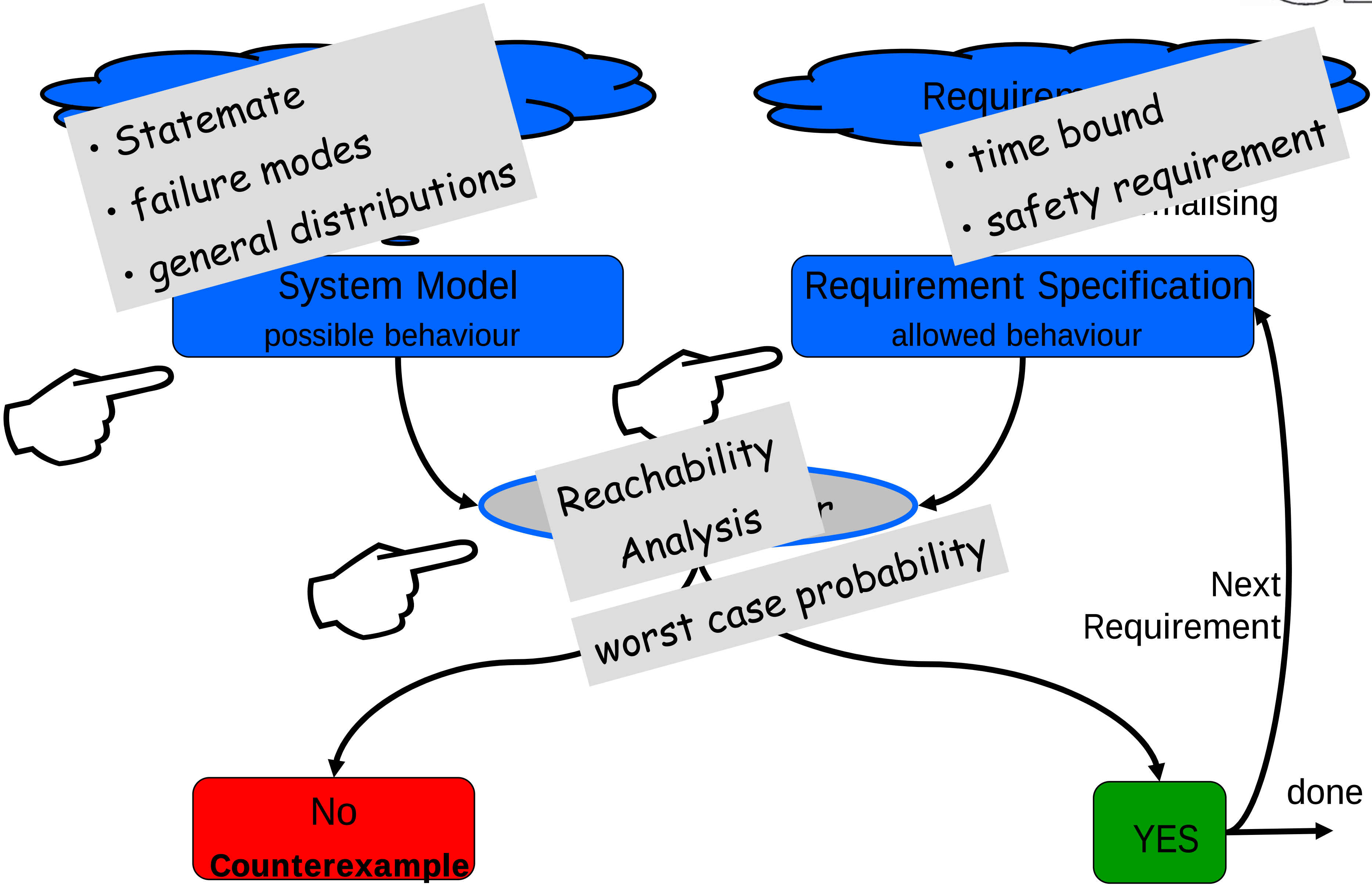
- LTS are uniform.

- The composition, minimisation and transformation algorithms are all ensured to preserve uniformity, if the input models are.

Theorem 6.1 Let IMC $\mathcal{M}$ be given and let $\mathcal{C}$ be its underlying CTMDP. The reachable states of $\mathcal{M}$ are uniform iff the reachable states of $\mathcal{C}$ are uniform.

[Johr]

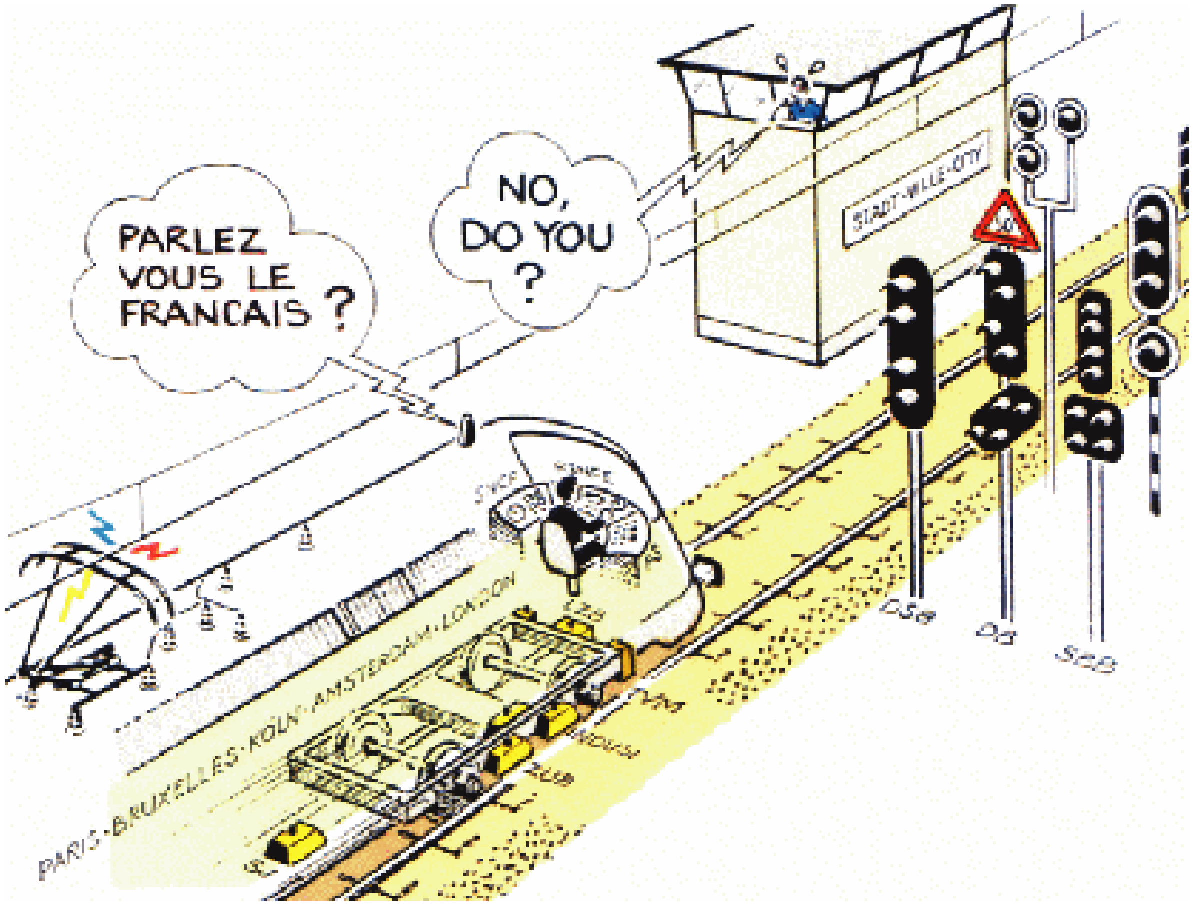
Where we stand.



An exemplary case:
Future European Train Coordination

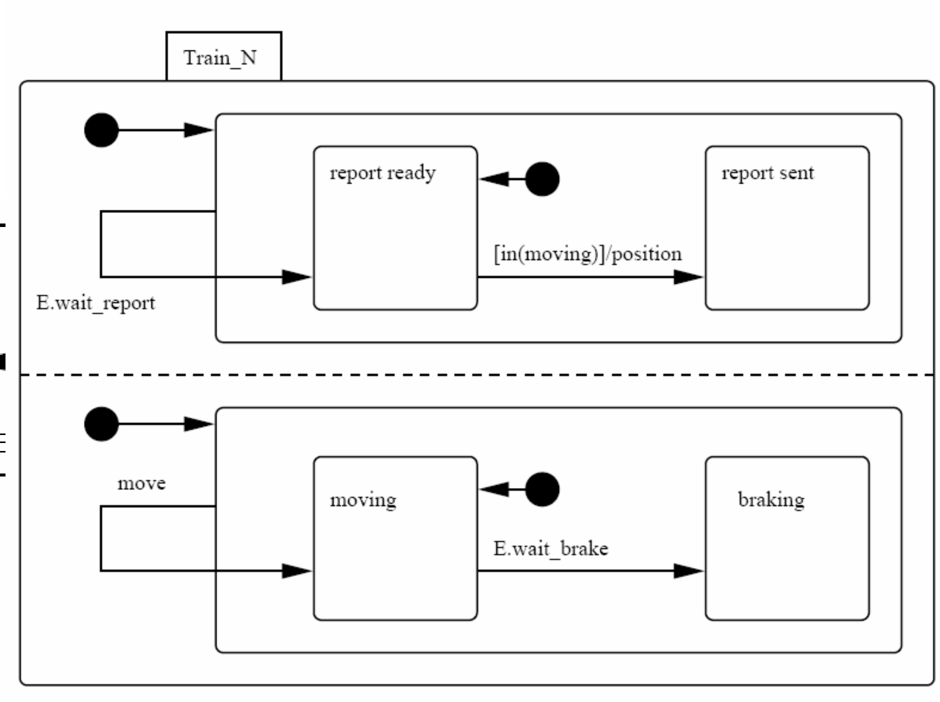
'ETCS'

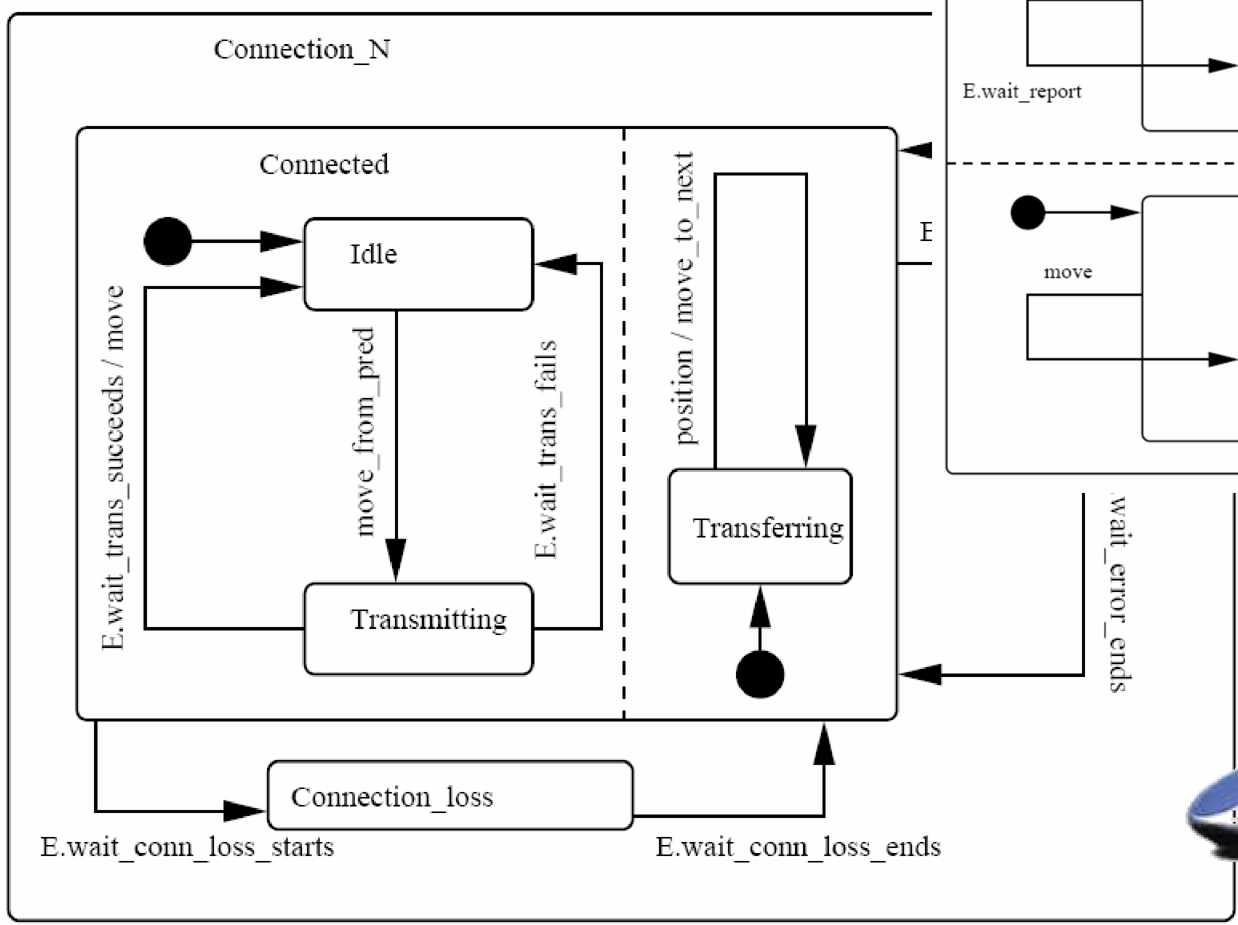
- Future standard for cross-European trains
- Defines train interoperability
- Moves functionality inside train, to improve track utilisation

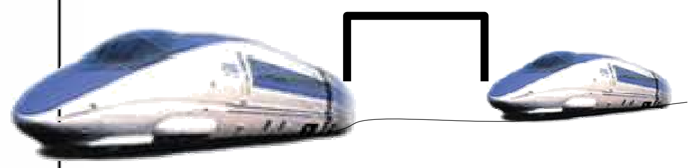


- Train-trackside communication via ‘GSM-R’

An exemplary case







- A sequence of N trains following each other
- What is the risk of having to brake within (10 or) 180 sec?

An exemplary case

MONOLITHIC CONSTRUCTION FOR ETCS WITH 2 TRAINS

Phases	Monolithic Construction			
	States	Transitions	G Time (sec.)	M Time (sec.)
1	33600	518464	12	3
5	302400	4142016	22	402
10	1016400	13521376	46	5154

EXPLICIT STEPS: COMPOSITION AND MINIMIZATION STATISTICS

Trains	Phases	Compositional Construction			Final Quotient IMC	
		States	Transitions	G + M Time (sec.)	States	Transitions
2	1	600	2505	42	355	1590
	5	10000	53625	61	5875	39500
	10	37500	207500	511	20000	154750
3	1	3240	16064	58	1375	5225
	5	64440	354100	813	36070	159119
	10	249480	1382900	10666	113650	533500
4	1	2870	11260	53	1435	5475
	5	57950	260350	420	30575	141000
	10	224900	1022700	7391	119650	558500

An exemplary case

MONOLITHIC CONSTRUCTION FOR ETCS WITH 2 TRAINS

Phases	Monolithic Construction			
	States	Transitions	G Time (sec.)	M Time (sec.)
1	33600	518464	12	3
5	302400	4142016	22	402
10	1016400	13521376	46	5154

EXPLICIT STEPS: CTMDP TRANSFORMATION AND ANALYSIS STATISTICS

Trains	Phases	Uniform CTMDP		Time for Analysis of Formula (sec.)	
		States	Transitions	$\sup_D \Pr_D(s, \overset{\leq 10}{\rightsquigarrow} B)$	$\sup_D \Pr_D(s, \overset{\leq 180}{\rightsquigarrow} B)$
2	1	227	352 (1.75)	0.06	0.44
	5	3127	3752 (4.60)	0.54	7.00
	10	11252	12502 (5.52)	2.23	31.15
3	1	787	1347 (1.10)	0.14	2.01
	5	21722	35942 (1.55)	6.24	89.39
	10	56452	90402 (1.84)	17.95	254.29
4	1	817	1457 (1.01)	0.16	2.28
	5	15477	26577 (1.57)	4.43	62.83
	10	59452	101402 (1.64)	19.94	280.88

Conclusion and Outlook

- Overview of the probabilistic behavioural model spectrum

- State-of-the-practice modelling
  combined with state-of-the-art analysis algorithm.

- More examples needed.

- Restricted to timed bounded reachability, no costs yet.
- Tool chain is long and prototypical, not easy to handle.

Make more of tool chain symbolic

- What I did not discuss:
  - Why symbolic vs. explicit this way