Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Quantitative Models and Analysis for Reactive Systems

PhD Defence

Claus Rørbæk Thrane

Distributed and Embedded Systems unit,
Department of Computer Science, Aalborg University

November 18, 2011

1 Introduction
- The big picture
- Specifications and Reactive Systems
- Research Hypothesis and Goals

2 Models, Specifications, and Verification
- Timed Automata and Semantics
- Specification Languages: CTL

3 Robustness and Quantitative Analysis
- From equivalences to distances
- Approximating properties of systems
- Approximating Specifications
- Robustness and Implementability

4 Conclusion & Final Remarks

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Understanding systems, and making useful predictions

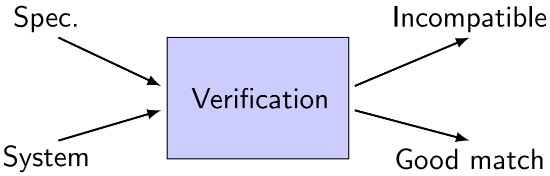
- Errors persist in mission and safety critical systems.
- Failures are expensive or tragic.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Understanding systems, and making useful predictions

- Errors persist in mission and safety critical systems.
- Failures are expensive or tragic.

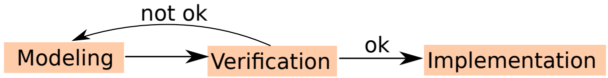
- Solution: *Formal methods*
  - Models of systems and requirements
  - Verification: manual or automated.



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Understanding systems, and making useful predictions

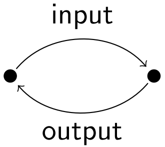
- Errors persist in mission and safety critical systems.
- Failures are expensive or tragic.
- Solution: *Formal methods*
  - Models of systems and requirements
  - Verification: manual or automated.



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Reactive systems

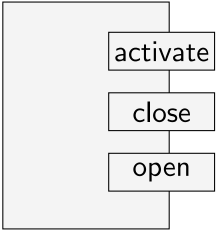
- Reactive systems [Pnueli'85] & [Milner '89]
  - non-terminating, communicating



- Control systems.
- Embedded systems.
- Distributed and communicating systems.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications



Qualitative

1. If **open** occurs an **activate** must have been performed.
2. If **open** occurs **close** must follow.

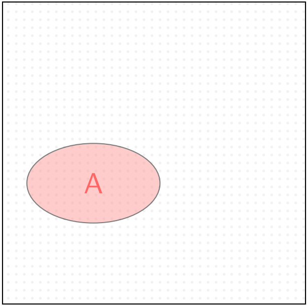
Quantitative

3. **open** will occur at most $12ms$ after **activate**.
4. ...

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications and Systems

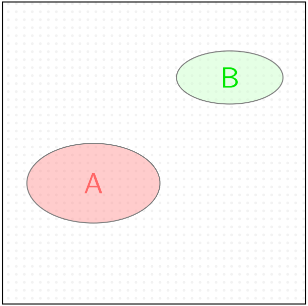
- Systems are solutions to specifications



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications and Systems

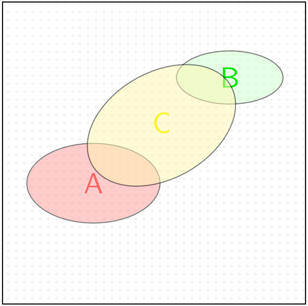
- Systems are solutions to specifications



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications and Systems

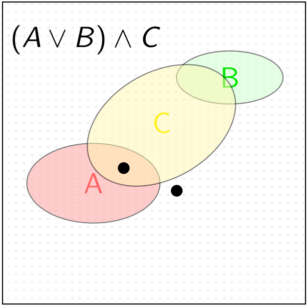
- Systems are solutions to specifications



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications and Systems

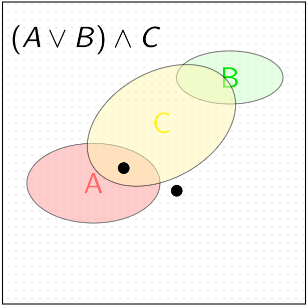
- Systems are solutions to specifications



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specifications and Systems

- Systems are solutions to specifications



- What if $(A \vee B) \wedge C$ is empty?
- How can we choose between systems for a specification?
- **Idea:** Specifications rate systems e.g. $A(s) = 0.7$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Thesis

Research Hypothesis

Using **quantitative techniques** and **game** theoretic approaches, it is possible to leverage the limitations of the Boolean framework for formal verification of reactive systems.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

1 Introduction
- The big picture
- Specifications and Reactive Systems
- Research Hypothesis and Goals

2 Models, Specifications, and Verification
- Timed Automata and Semantics
- Specification Languages: CTL

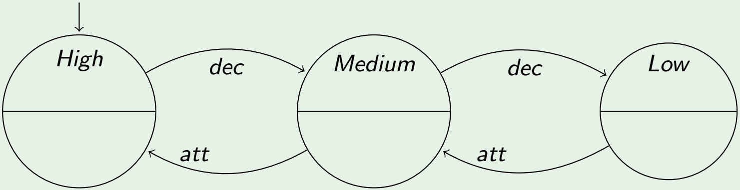
3 Robustness and Quantitative Analysis
- From equivalences to distances
- Approximating properties of systems
- Approximating Specifications
- Robustness and Implementability

4 Conclusion & Final Remarks

Real-Time Reactive Systems

- Timed automaton [Alur and Dill '94]
- Weighted timed automaton [Alur+, Behrmann+ '01]

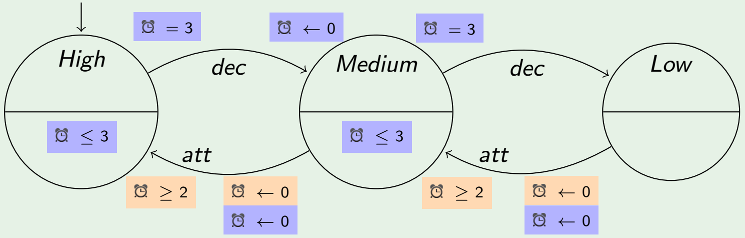
A simple production system



Real-Time Reactive Systems

- Timed automaton [Alur and Dill '94]
- Weighted timed automaton [Alur+, Behrmann+ '01]

A simple production system

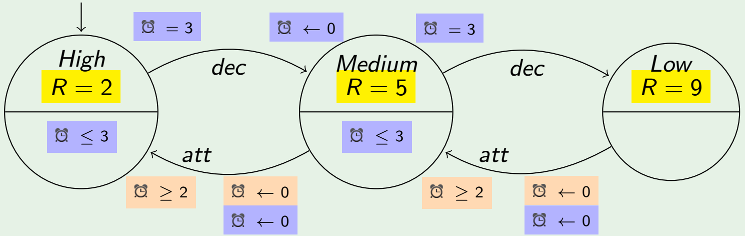


Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Real-Time Reactive Systems

- Timed automaton [Alur and Dill '94]
- Weighted timed automaton [Alur+, Behrmann+ '01]

A simple production system

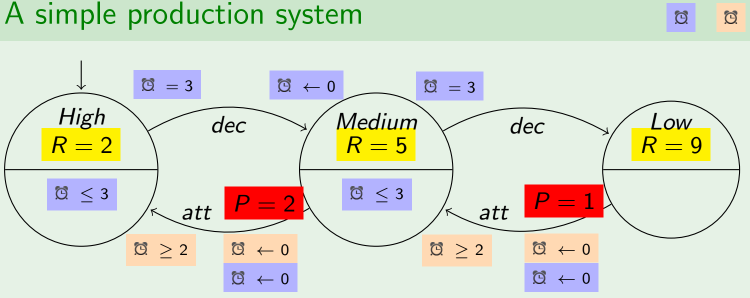


Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Real-Time Reactive Systems

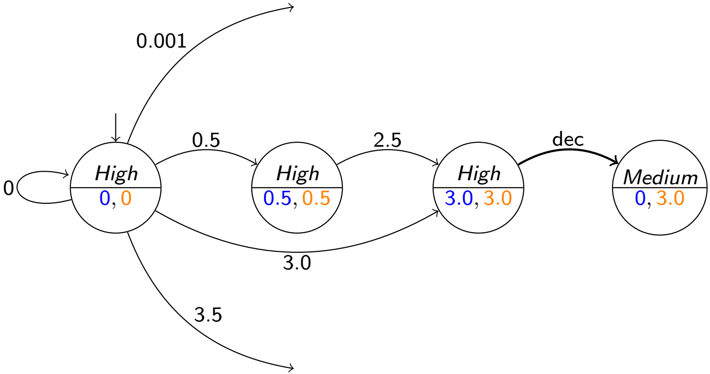
- Timed automaton [Alur and Dill '94]
- Weighted timed automaton [Alur+, Behrmann+ '01]

A simple production system



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Semantics



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Ehrenfeucht-Fraïssé games

- Characterizes Bisimulation [Milner '89]
- A blind Attacker gives us trace equivalence [Hoare '85]
- Time abstract games!

Bisimulation game [Stirling '95]

An *Attacker* and *Defender* plays a round from s and t.

1. Attacker chooses s or t and a move, e.g. $s \xrightarrow{a} s'$
2. Defender proposes a matching move, e.g. $t \xrightarrow{a} t'$, from s or t opposite the attacker.

Another round is played from s' and t' if a match was found.

- $s \not\sim t$ if the attacker can *win*, and $s \sim t$ otherwise.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Specification languages: CTL

- Safety properties & invariants

CTL Specifications [Clarke, Emerson '81]

- AGEX(*true*) — Non-termination.
- EF(*error*) — Reachability.
- AGEF(EX(*Medium*) $\vee$ EX(*Low*)) — "invariant choice".

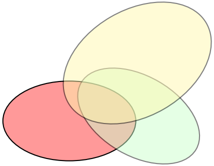
TCTL Specifications [Alur, Courcoubetis Dill '93]

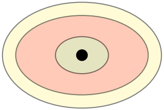
- $High \vee EF_{[0,4]} High$ — If the production level is not H, can it be obtained within 4 time units?

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Properties of CTL

[Brown, Clarke, Grümberg '87]





- Adequacy: CTL can distinguish (only) inequivalent systems.
- Expressivity: can express specifications with exactly one solution (up to $\sim$)!

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

1 Introduction
- The big picture
- Specifications and Reactive Systems
- Research Hypothesis and Goals

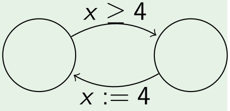
2 Models, Specifications, and Verification
- Timed Automata and Semantics
- Specification Languages: CTL

3 Robustness and Quantitative Analysis
- From equivalences to distances
- Approximating properties of systems
- Approximating Specifications
- Robustness and Implementability

4 Conclusion & Final Remarks

Towards Quantitative Analysis

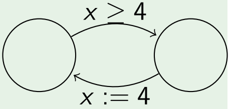
Models ✓



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Towards Quantitative Analysis

Models ✓



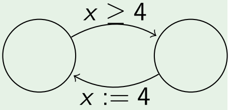
Specifications ✓

$\mathrm{AF}_{[0,4]} \mathit{High}$

Towards Quantitative Analysis

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Models ✓



Specifications ✓

$AF_{[0,4]} High$

Verification

$[\![\phi]\!](s) = 3.14$

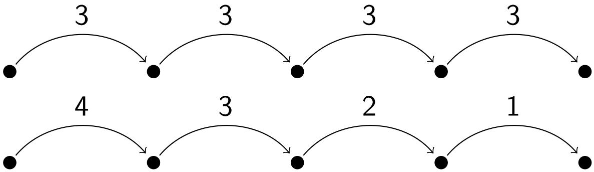
$d(s,t) = 42$

Boolean world	**"Quantification"**
Trace equivalence $\equiv$	Linear distance d_L
Bisimilarity $\sim$	Branching distance d_B
$s \sim t$ implies $s \equiv t$	$d_L(s,t) \leq d_B(s,t)$
Satisfaction $s \models \phi$	Multi-valued $[\![\phi]\!](s) \in \mathbb{R}$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Behavior revisited

- Games are no longer win/loose but have values.
- Players try to optimize the value of the game.

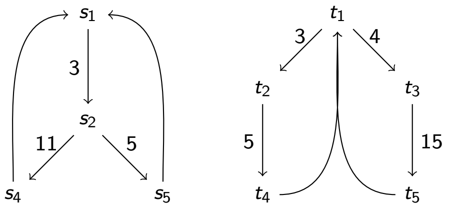


- Point-wise distance 2
- Hamming distance 3
- Accumulating (discounted) distance 4
- Maximum-lead distance 2

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



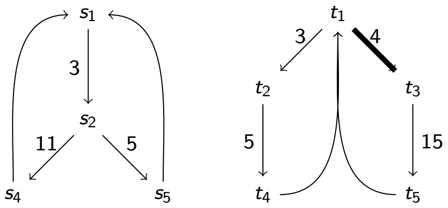
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



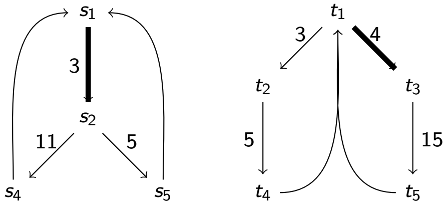
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



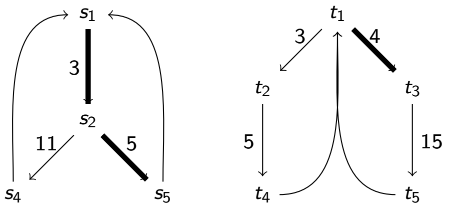
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



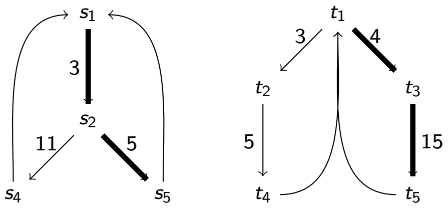
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



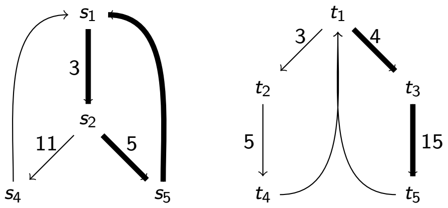
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



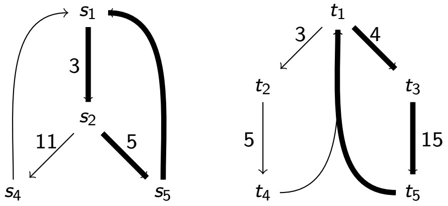
$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

The accumulating distance, discounted by $\lambda = 0.9$



$$d_L(s_1, t_1) = \sum_i (1 + 4\lambda)\lambda^{3i} \approx 17.0$$

$$d_B(s_1, t_1) = 1 + 10\lambda + \lambda^3 d_B(s_1, t_1) \approx 36.9$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Measuring the dissimilarity between systems

- In paper A

Theorem

Branching distances bound linear distances.

Theorem

For discounting factor $\lambda < 1$, accumulating branching distance from deterministic to non-deterministic weighted timed automata is computable.

- In paper B

Theorem

Computing accumulating distance is polynomial-time equivalent to computing the payoff for discounted games.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Interpreting WCTL quantitatively

Syntax & Semantics

$$\Phi ::= \mathsf{p} \mid \neg \mathsf{p} \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \mathsf{E}\Psi \mid \mathsf{A}\Psi$$

$$\Psi ::= \mathsf{X}_c \Phi \mid \mathsf{G}_c \Phi \mid \mathsf{F}_c \Phi \mid [\Phi_1 \mathsf{U}_c \Phi_2]$$

Every ϕ is interpreted $[\![\phi]\!]$: as a function in $[S \to \mathbb{R}_{\geq 0}]$

Example: $\phi = \mathsf{AG}(\mathit{High} \vee \mathsf{EF}_2 \mathit{High})$

$$[\![\phi]\!](s) = \sup_{\sigma \in P(s), k} \min \begin{cases} [\![\mathit{High}]\!](\sigma^k) \\ \inf_{\sigma' \in P(\sigma^k), k'} \sum_{j=0}^{k} |\sigma'(j)_w - 2| + [\![\mathit{High}]\!](\sigma'^{k'}) \end{cases}$$

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

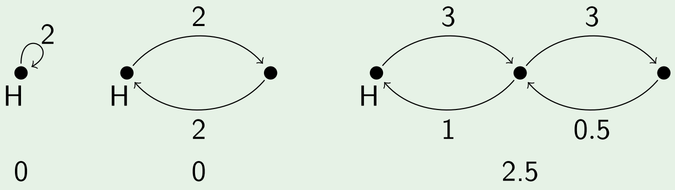
Interpreting WCTL quantitatively

Syntax & Semantics

$$\Phi ::= p \mid \neg p \mid \Phi_1 \wedge \Phi_2 \mid \Phi_1 \vee \Phi_2 \mid \mathsf{E}\Psi \mid \mathsf{A}\Psi$$
$$\Psi ::= \mathsf{X}_c\Phi \mid \mathsf{G}_c\Phi \mid \mathsf{F}_c\Phi \mid [\Phi_1 \mathsf{U}_c \Phi_2]$$

Every ϕ is interpreted $[\![\phi]\!]$: as a function in $[S \to \mathbb{R}_{\geq 0}]$

Example: $\phi = \mathsf{AG}(High \vee \mathsf{EF}_2 High)$

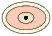


Properties of WCTL

- In paper D

Quantitative Adequacy [de Alfaro, Faella, Stoelinga'04]

For every S and T $d_B(S,T) \leq \epsilon$ if and only if, for every property ϕ in WCTL $|[\![\phi]\!](S) - [\![\phi]\!](T)| \leq \epsilon$



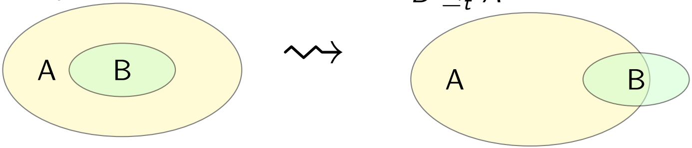
Quantitative Expressiveness

For each S, and every $\gamma > 0$, there is a (single) characteristic property ϕ_S^γ in WCTL, such that: $[\![\phi_S^\gamma]\!](T) \in [\epsilon - \gamma, \epsilon + \gamma]$ if and only if $d_B(S,T) \leq \epsilon$

Approximating Specifications: MTS & Quantitative Refinement

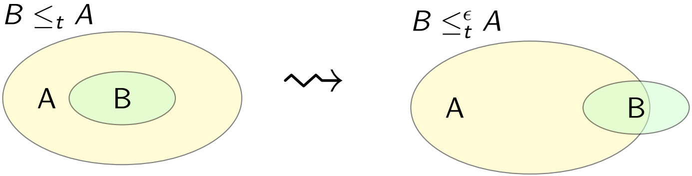
- Modal Transition Systems [Larsen & Thomsen '88]
  - (De)Composition of specifications: $A \parallel B$ and $A \setminus\!\!\setminus B$

$B \le_t A$ $\quad\leadsto\quad$ $B \le_t^{\epsilon} A$



Approximating Specifications: MTS & Quantitative Refinement

- Modal Transition Systems [Larsen & Thomsen '88]
  - (De)Composition of specifications: $A \parallel B$ and $A \setminus\!\!\setminus B$

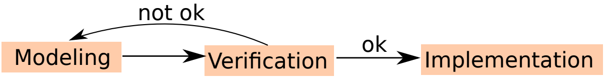


Results: Paper E

+ EXPTIME-hard to decide $B \leq_t^\epsilon A$, given $\epsilon > 0$
+ $B \leq_m^\epsilon A$ is decidable in $NP \cap coNP$, given $\epsilon > 0$
− No suitable conjunction operator ($\wedge$) is definable.

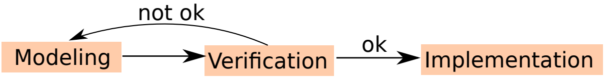
Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation Issues



Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation Issues



- Digital clock suffers from drift and finite precision.
- Digital hardware has finite execution speed.

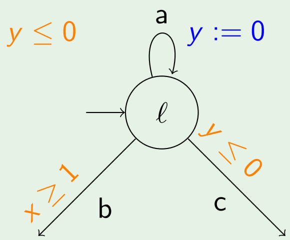
Dynamical properties [Puri'98]

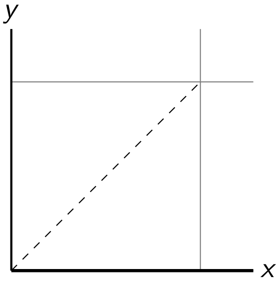
The effects of physical hardware corresponds to implicitly statically *enlarging* all constraints by some small $\Delta > 0$.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation Issues

Example $\mathcal{B}$

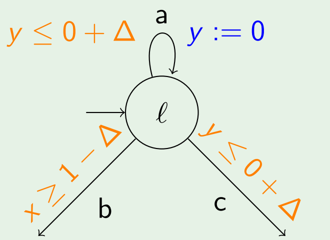


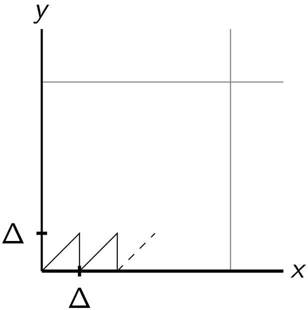


Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation Issues

Example $\mathcal{B}_\Delta$

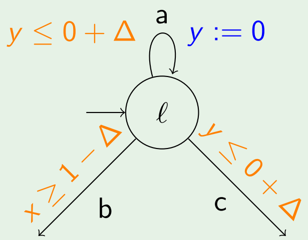


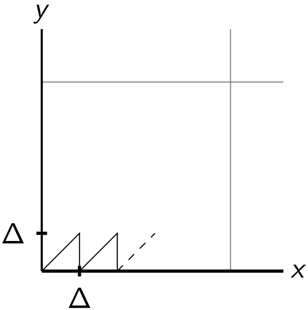


Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation Issues

Example $\mathcal{B}$

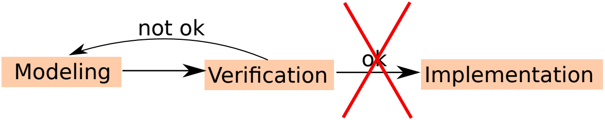




- The Attacker wins the untimed game: **b** and **c** are available

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Implementation issues



- Enlargement may induce extra (discrete) behavior
- Hence the formalism lacks **robustness**.

Even if models were *robust* what can we guarantee about the timing of implementations – in the presence of Δ?

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Resolving implementation issues

Safety robustness [Puri'98]

A timed automata $\mathcal{A}$ is safety robust w.r.t. locations B if there exist a $\Delta > 0$ such that $\mathcal{A}_\Delta$ is safe for B.

- in paper F we consider a stronger notion, capturing also *reactive* expectations.

ϵ-robustness

A timed automata $\mathcal{A}$ is ϵ-robust, for $\epsilon > 0$, if there exist a $\Delta > 0$ such that $d_B(\mathcal{A}, \mathcal{A}_\Delta) \leq \epsilon$.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Resolving implementation issues

- in paper F

Theorem: Implementability of Timed Automata

Let $\mathcal{A} = (\mathcal{L}, \mathcal{C}, \Sigma, l_0, E)$ be a TA, safe w.r.t. $B \subseteq \Sigma$, then:

1. it has safety robust implementation with the same clocks and locations, and at most $|E| \cdot |Reg(\mathcal{A})|$ edges.
2. For all $\epsilon > 0$, it has a
   - ϵ-robust implementation w.r.t. $\sim_0$
   - ϵ-sampled and ϵ-robust implementation w.r.t. $\approx_{0^+}$.

- Meaning all WCTL properties are transferable between the intended design and the implementation.

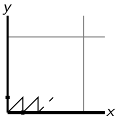
Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Robust implementations

Given a timed automaton $\mathcal{A}$, construct $\mathcal{A}'$ such that

- $\mathcal{A}$ has the same behaviour as $\mathcal{A}'$,
- $\mathcal{A}'$ is robust, i.e. $\mathcal{A}'$ has approximately the same behaviour as $\mathcal{A}'_\Delta$ for some $\Delta > 0$.

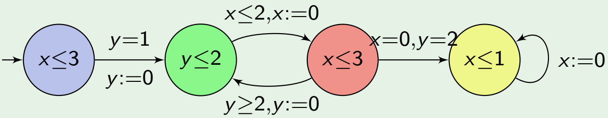
Notice: $\mathcal{B}_\Delta$ didn't respect the region automaton.

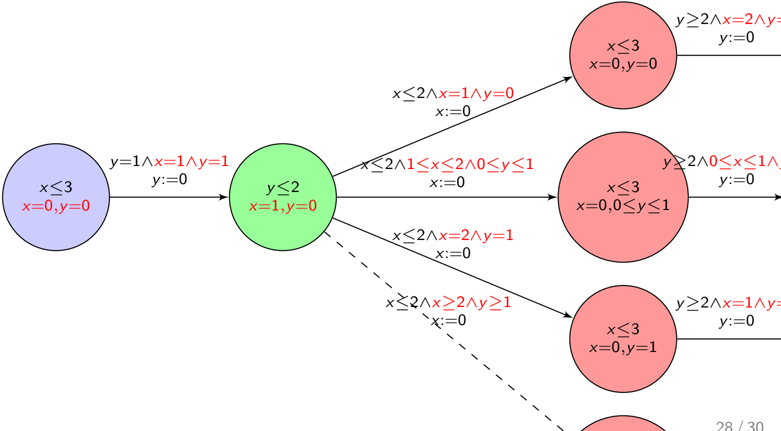


Basic idea: Enforce the region automaton: encoding regions in locations.

Robust implementations

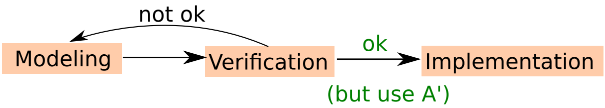
Example $\mathcal{A}$





Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Consequences of (point-wise) ϵ-robustness



Reuse of tools

We need not rebuild existing tools providing automated verification. Rather the **code generation** step, will need to apply our construction.

Reliable fault detectors

A point-wise deviation may provide a upper bound in delays for each step of a communication protocol.

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Conclusion & Final Remarks

- Distances yield meaningful approximations of equality, system properties, and specifications by other specifications.
- Games turn out to be useful in defining distances.

What else?

- Quantitative analysis preserves expressivity, and the hierarchy of equivalences.
- What about other types of qualitative behavior?
- What about measuring Cost, Energy, Radiation?
- What about stochastic and probabilistic systems?

Introduction
The big picture
Reactive Systems
Goals
Formal Methods
Timed Systems
Specifications
Quantitative Analysis
Distances
Quantitative Specifications
Approximating Specifications
Robustness
Conclusion

Conclusion & Final Remarks

- Distances yield meaningful approximations of equality, system properties, and specifications by other specifications.
- Games turn out to be useful in defining distances.

What else?

- Quantitative analysis preserves expressivity, and the hierarchy of equivalences.
- What about other types of qualitative behavior?
- What about measuring Cost, Energy, Radiation?
- What about stochastic and probabilistic systems?

Thank you!