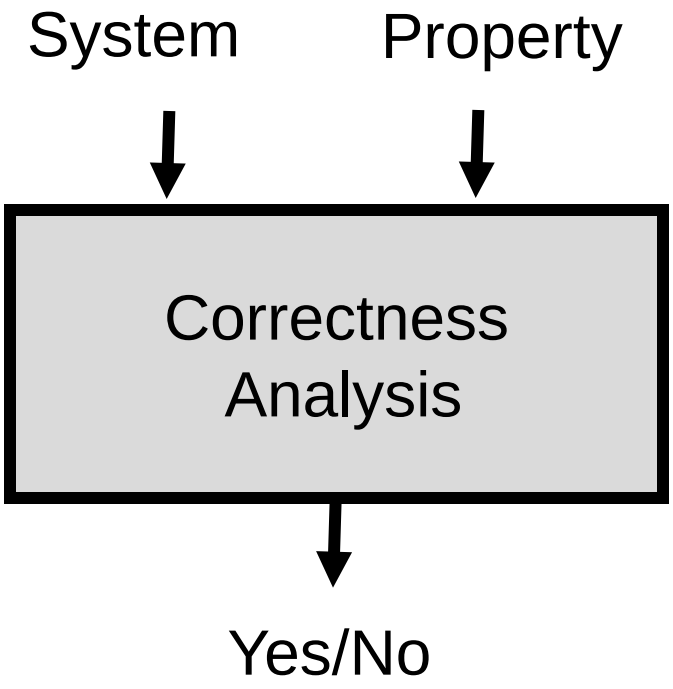
The Quest for Average Response Time

Tom Henzinger
IST Austria

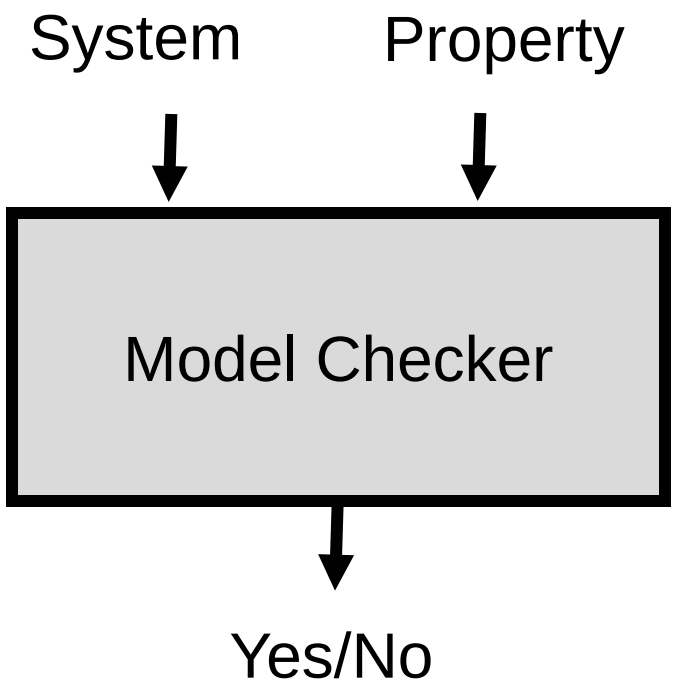
Joint work with Krishnendu Chatterjee and Jan Otop.

Formal Verification



Formal Verification

Transition System



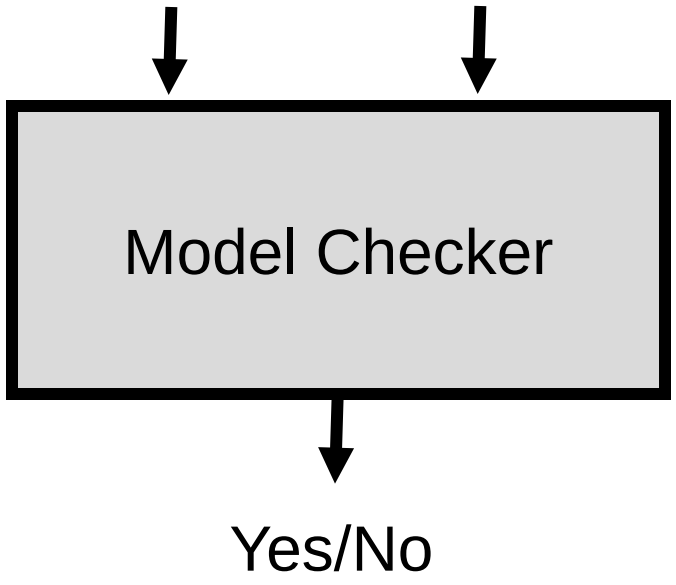
$\Box (r \Rightarrow \Diamond g)$

Formal Verification

Timed Automaton

Quantitative Property

$\Box (r \Rightarrow \Diamond_{\leq 5} g)$

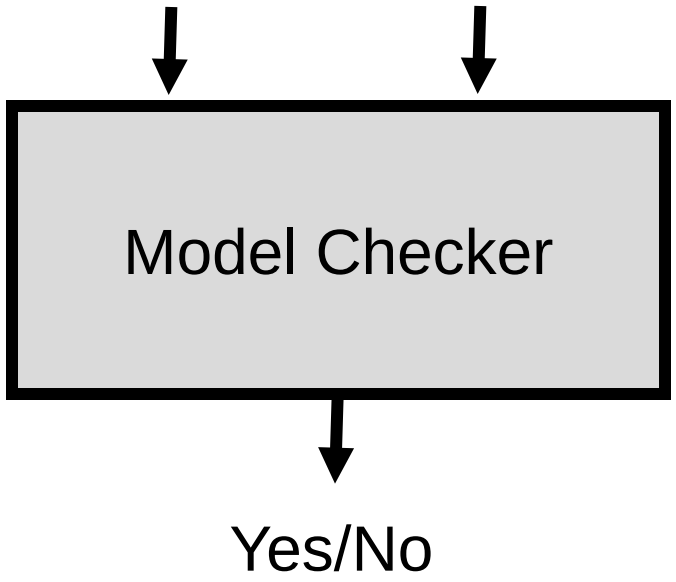


Formal Verification

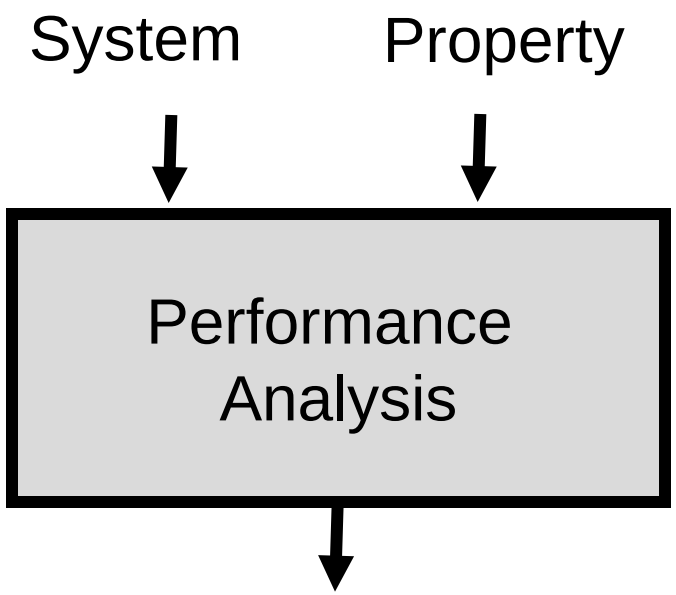
Markov Process

Quantitative Property

$\forall\Box\,(r \Rightarrow Pr(\Diamond g) \geq 0.5)$



Quantitative Formal Methods



$\Box (r \Rightarrow \Diamond g)$

Worst or average response time

Quantitative Formal Methods

From checking correctness
to measuring performance and robustness of systems:

Quantitative temporal logics
Quantitative automata
Quantitative abstractions
Quantitative synthesis
etc.

Quantitative Formal Methods

From checking correctness
to measuring performance and robustness of systems:

Quantitative temporal logics
Quantitative automata
Quantitative abstractions
Quantitative synthesis
etc.

None of these have captured average response time.

Observations

r	request
g	grant
t	tick
x	neither

Σ = {r,g,t,x}

System Behavior = Observation Sequence

x t t x x r x x t x x x r x t x t x x t g r t t g g x t x t x ...

Response Times

x t t x x r x x t x x x r x t x t x x t g r t t g g x t x t x ...

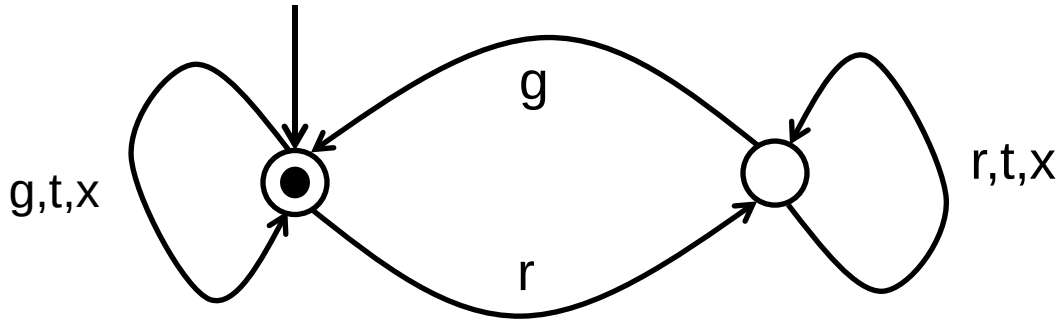
4,3,2

Response Times

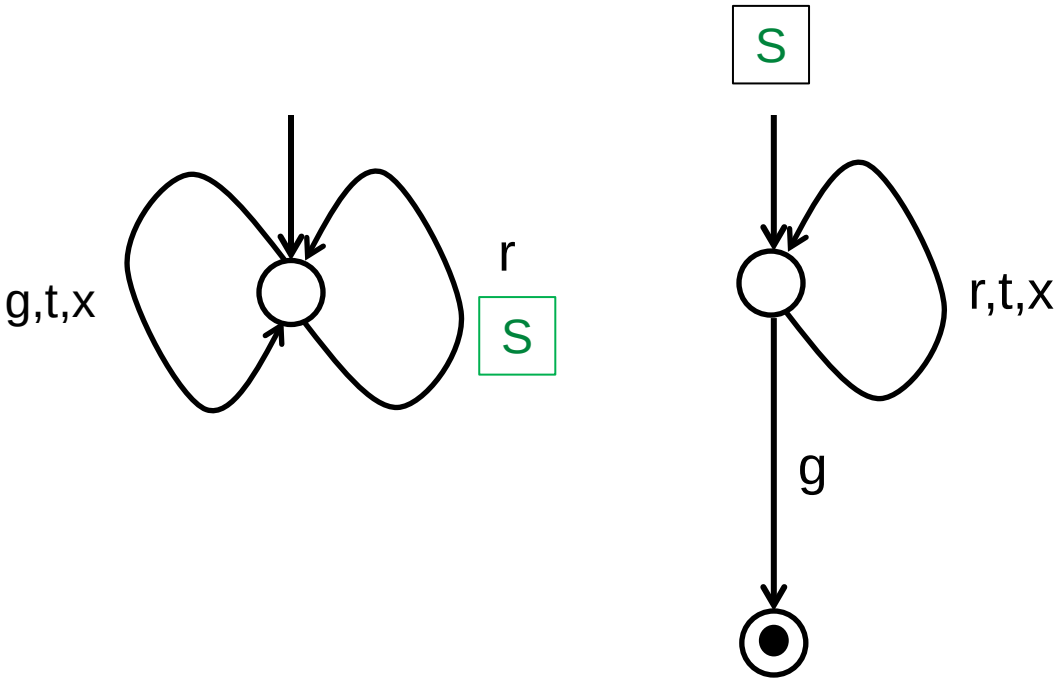
x t t x x r x x t x x x r x t x t x x t g r t t g g x t x t x ... 4,3,2

r t t t t t t t ... ∞

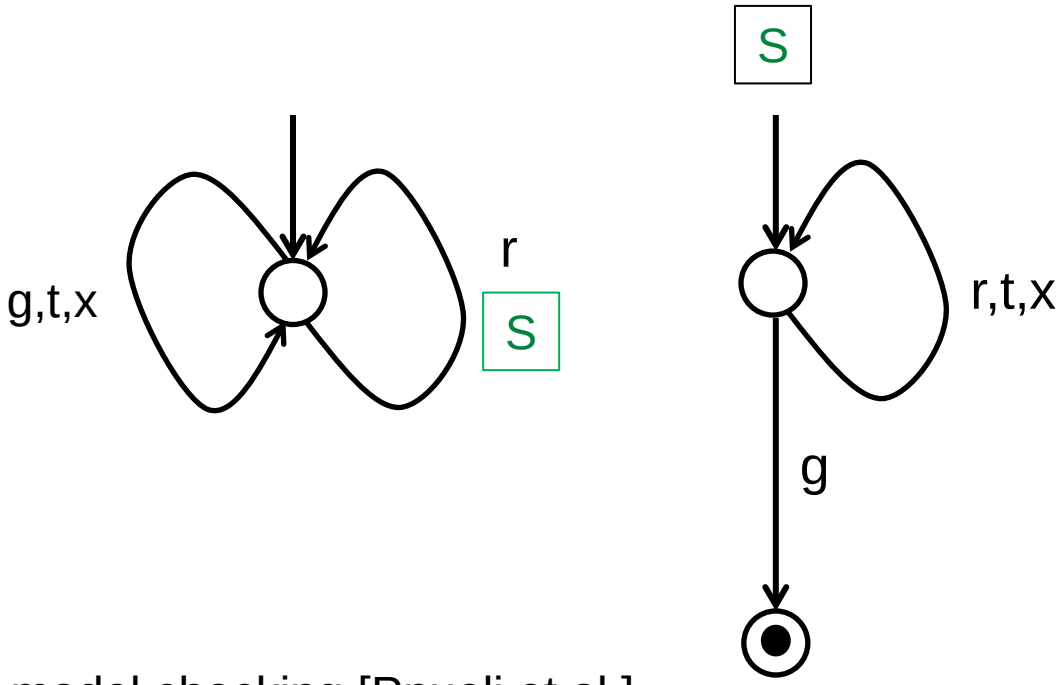
Response Property



Response Monitor

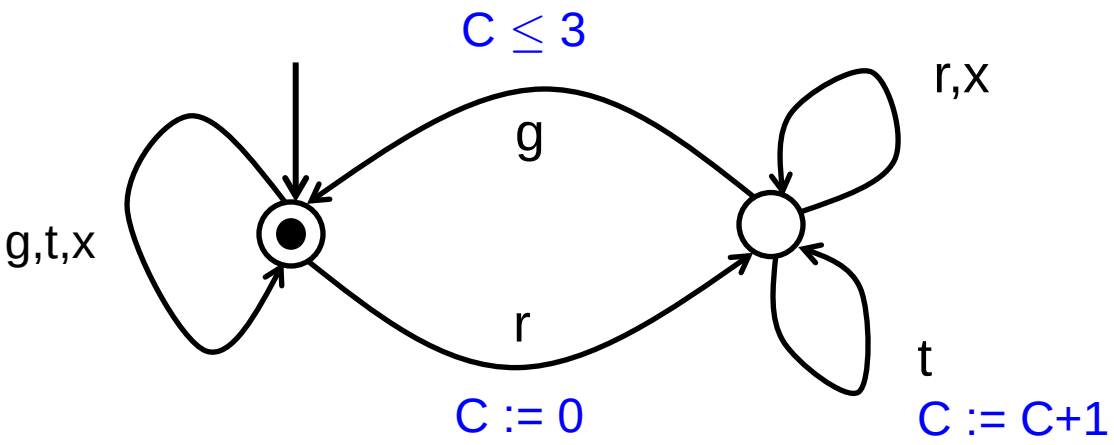


Response Monitor

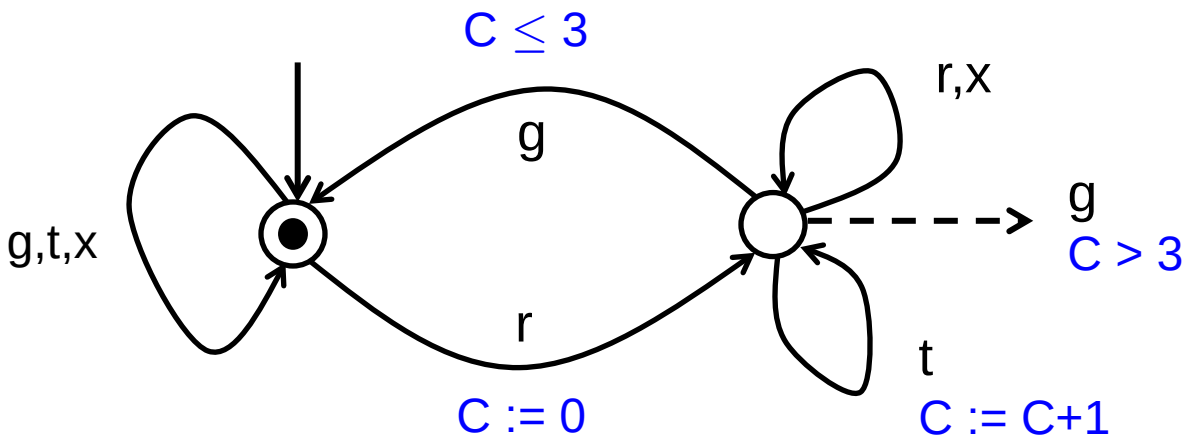


Decomposing model checking [Pnueli et al.]
Alternating automata
Run-time verification

Bounded Response

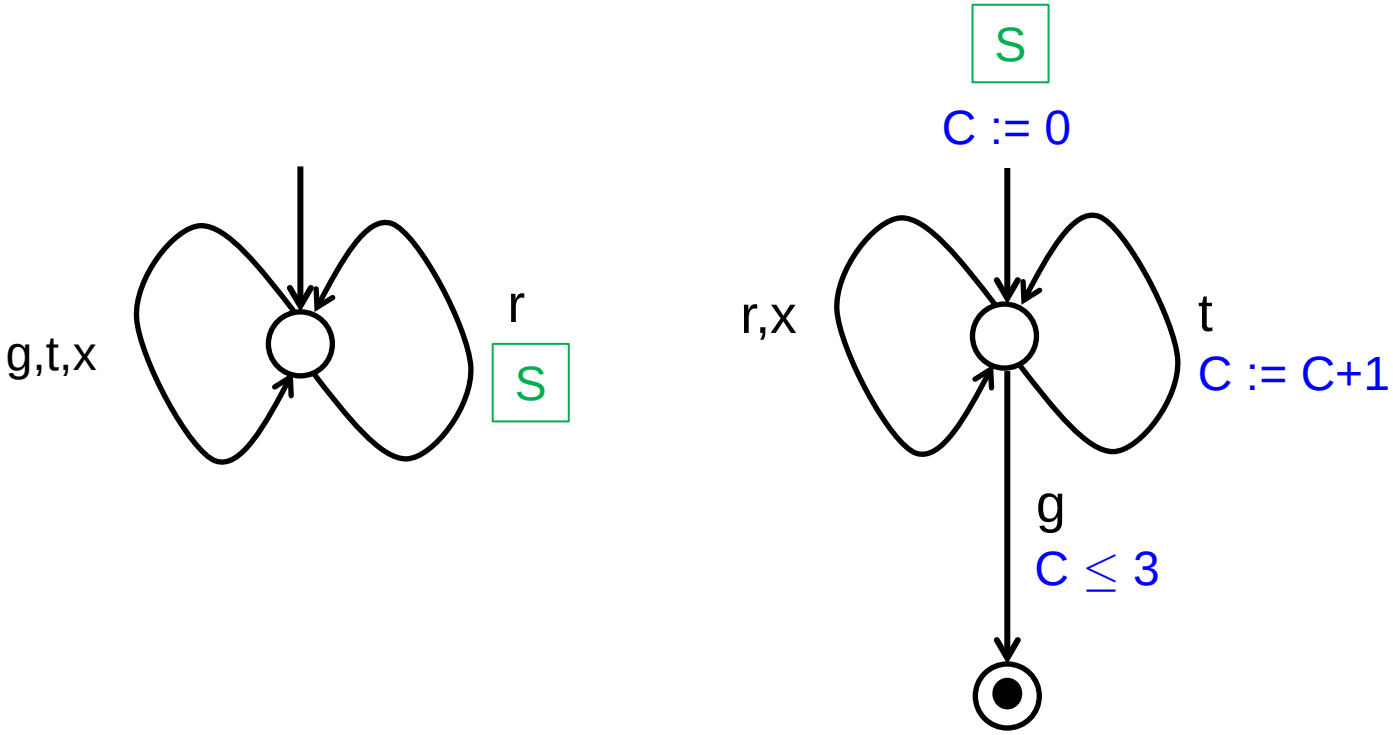


Bounded Response

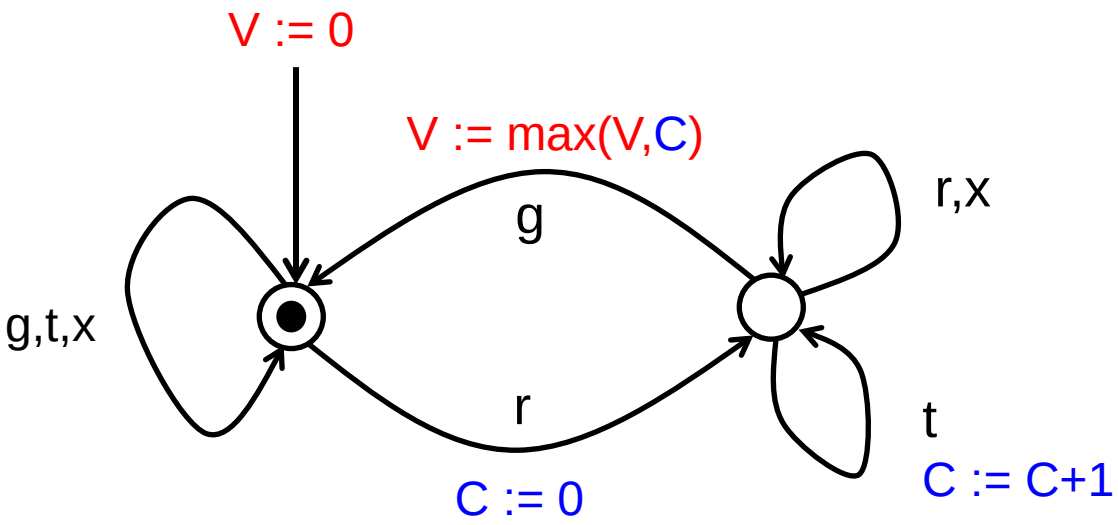


(Discrete) clocks exponentially succinct,
but not more expressive than finite state.

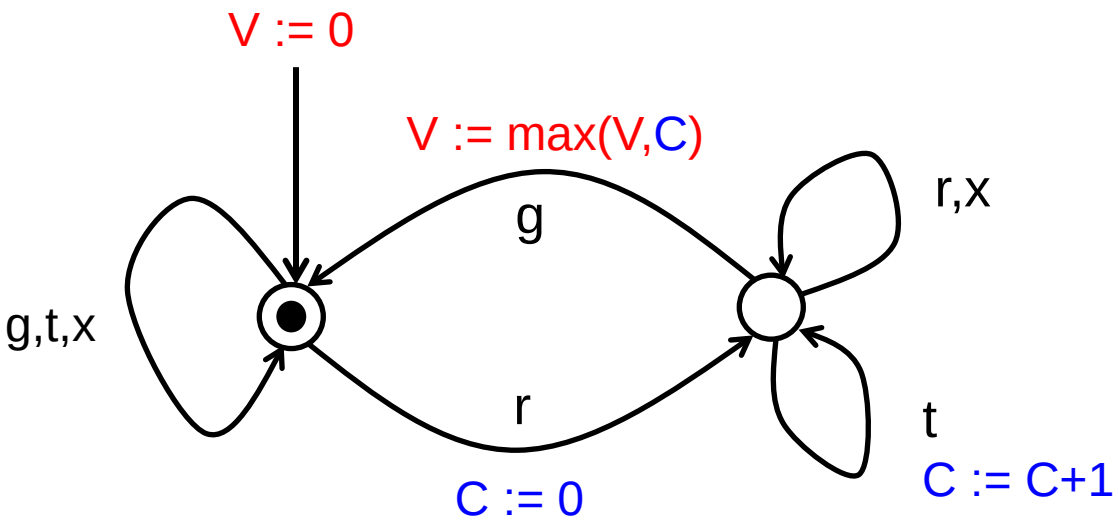
Bounded Response Monitor



Maximal Response

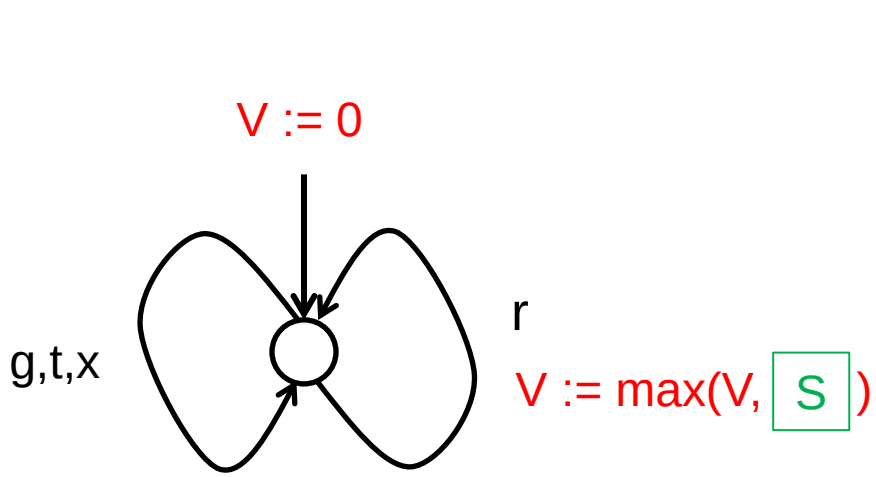


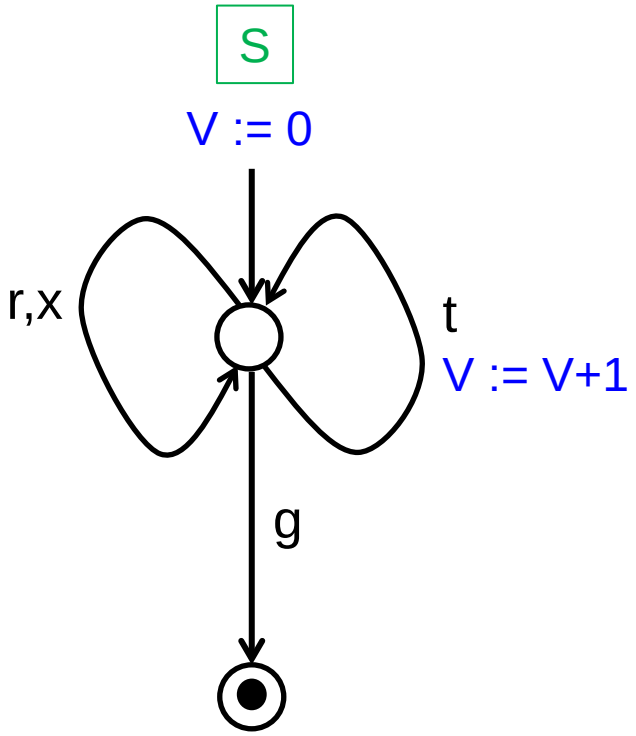
Maximal Response



Value of an infinite run is liminf of V.

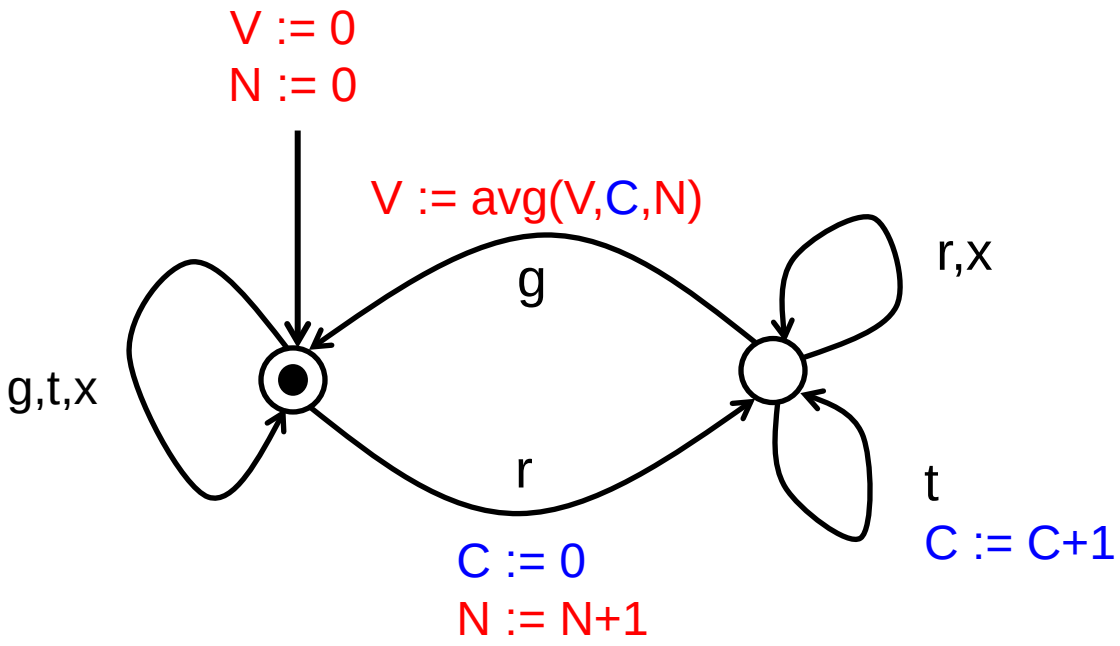
Maximal Response Monitor





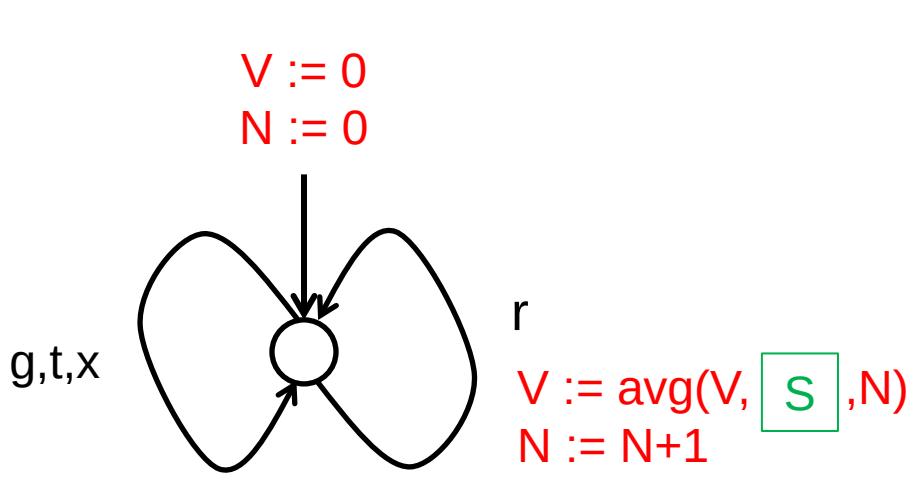
S is final value of V.

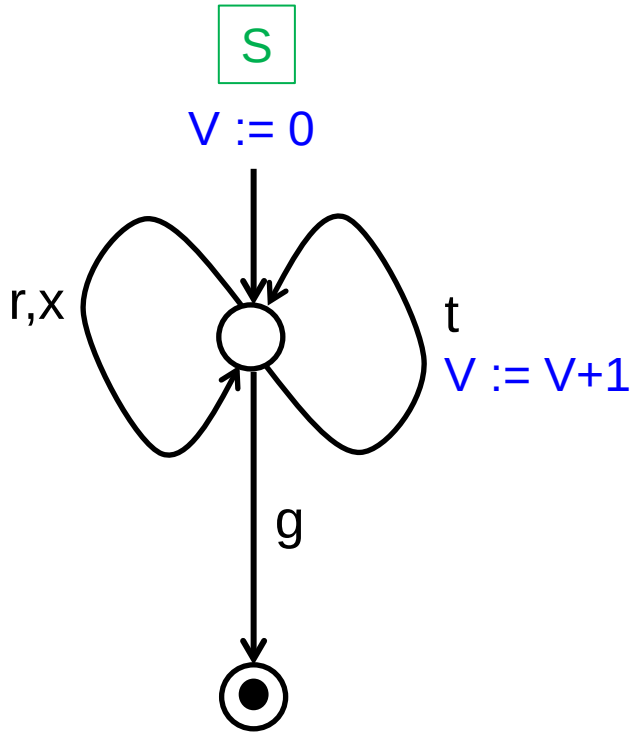
Average Response



avg(V,C,N) = (V·(N-1)+C) / N

Average Response Monitor





(max,inc) automata:

Master automaton maintains the max of values returned by slaves (1 max register).

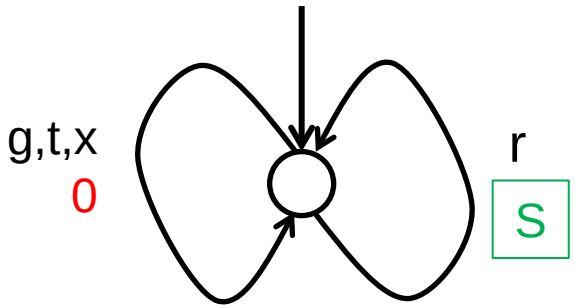
Each slave automaton counts occurrences of t (1 inc register).

(avg,inc) automata:

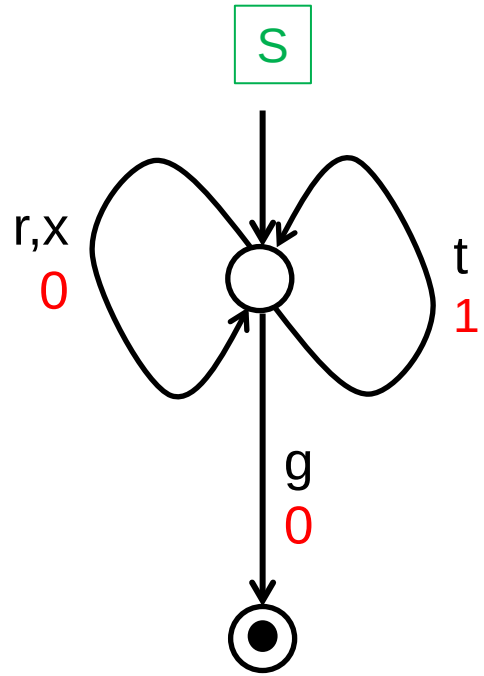
Master automaton maintains the avg of values returned by slaves (1 avg register).

Slaves as above.

Nested Weighted Automata



limavg of weights



sum of weights

Function on weights instead of registers.

Unlike in the qualitative case,

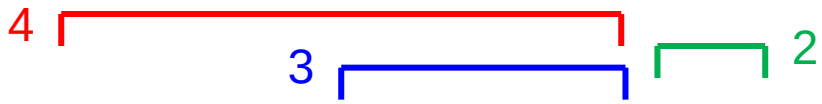
nested weighted automata more expressive than
flat weighted automata:

1. value of flat limavg automaton bounded by largest weight
   (cannot specify average response time)

2. flat automata have constant “width” (number of registers)

Deterministic qualitative automaton A: $\Sigma^\omega \to \mathbb{B}$
Deterministic quantitative automaton A: $\Sigma^\omega \to \mathbb{R}$

Deterministic qualitative automaton A: $\Sigma^\omega \to \mathbb{B}$
Deterministic quantitative automaton A: $\Sigma^\omega \to \mathbb{R}$



ω = x t t x x r x x t x x x r x t x t x x t g r t t g g x t x t x ...

Response(ω) = 1
BoundedResponse(ω) = 0

MaximalResponse(ω) = 4
AverageResponse(ω) = 3

Nondeterministic qualitative automaton A: $\Sigma^\omega \to \mathbb{B}$
A(ω) = max{ value(ρ) | ρ run of A and obs(ρ) = ω }

Nondeterministic quantitative automaton A: $\Sigma^\omega \to \mathbb{R}$
A(ω) = sup{ value(ρ) | ρ run of A and obs(ρ) = ω }

Nondeterministic qualitative automaton A: $\Sigma^\omega \to \mathbb{B}$
A(ω) = max{ value(ρ) | ρ run of A and obs(ρ) = ω }

Emptiness: $\exists\omega$. A(ω) = 1
Universality: $\forall\omega$. A(ω) = 1

Nondeterministic quantitative automaton A: $\Sigma^\omega \to \mathbb{R}$
A(ω) = sup{ value(ρ) | ρ run of A and obs(ρ) = ω }

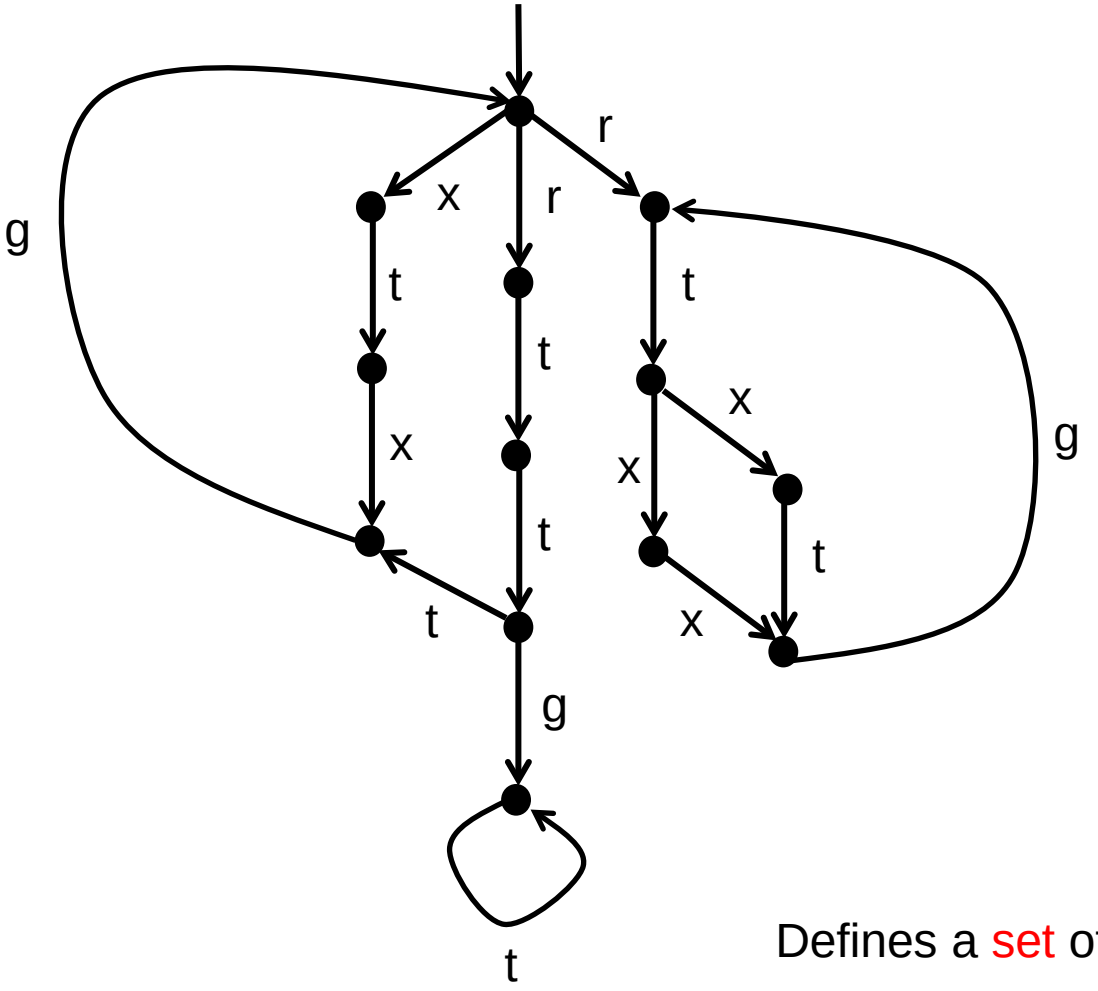
Nondeterministic qualitative automaton A: $\Sigma^\omega \to \mathbb{B}$
A(ω) = max{ value(ρ) | ρ run of A and obs(ρ) = ω }

Emptiness: $\exists\omega$. A(ω) = 1
Universality: $\forall\omega$. A(ω) = 1

Nondeterministic quantitative automaton A: $\Sigma^\omega \to \mathbb{R}$
A(ω) = sup{ value(ρ) | ρ run of A and obs(ρ) = ω }

Emptiness: $\exists\omega$. A(ω) $\geq \lambda$
Universality: $\forall\omega$. A(ω) $\geq \lambda$

Transition System = Labeled Graph



Defines a set of behaviors.

Quantitative Analysis of Transition Systems

Given transition system K and quantitative property A,

Q1. does some run of K correspond to a run of A with value $V \geq \lambda$?
[emptiness of $K \times A$]

Q2. does every run of K correspond to a run of A with $V \geq \lambda$?
[as hard as universality of A]

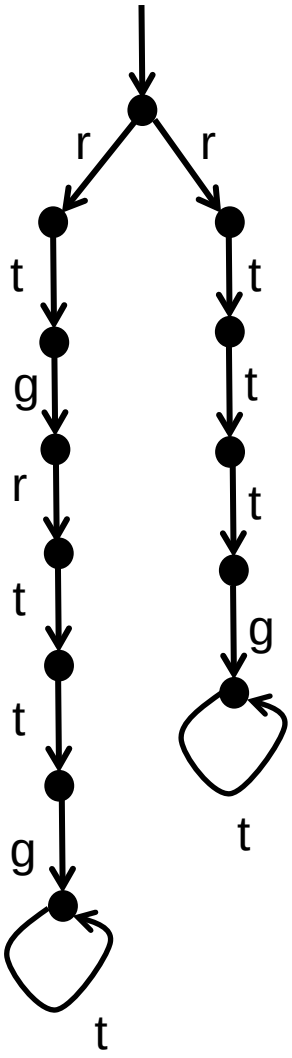
Quantitative Analysis of Transition Systems

Given transition system K and quantitative property A,

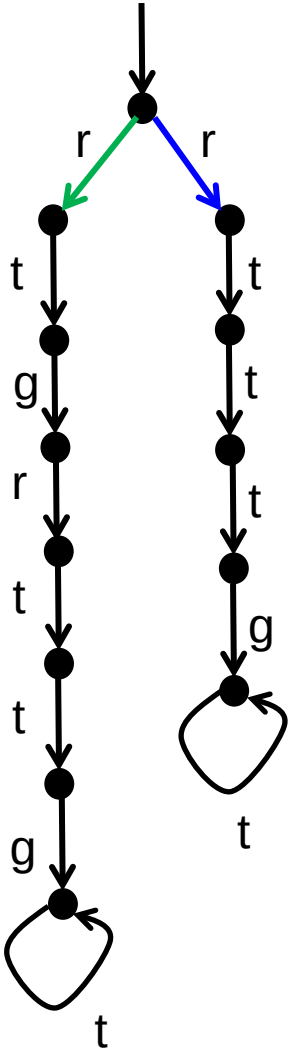
Q1. does some run of K correspond to a run of A with value $V \geq \lambda$?
[emptiness of $K \times A$]

Q2. does every run of K correspond to a run of A with $V \geq \lambda$?
For deterministic A, equivalently:
does some run of K correspond to a run of A with $V < \lambda$?
[emptiness of $K \times -A$]

Example



Example

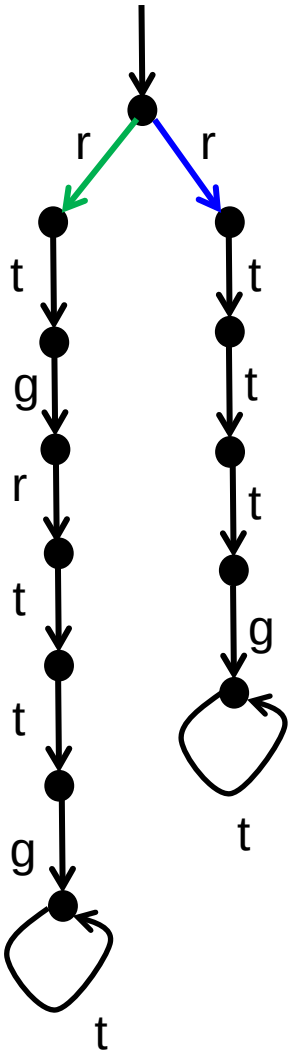


Best maximal response time: 2

Worst maximal response time: 3

Emptiness of product with (max,inc) automaton

Example



Best maximal response time: 2
Worst maximal response time: 3

Emptiness of product with (max,inc) automaton

Best average response time: 1.5
Worst average response time: 3

Emptiness of product with (avg,inc) automaton

Results on (max,inc) Automata

	Nondet (max,inc)	Det (max,inc)
Emptiness	PSPACE	PSPACE
Universality	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE

Qualitative nondeterministic universality PSPACE, emptiness and deterministic universality PTIME.

Results on (avg,inc) Automata

	Nondet (avg,inc)	Det (avg,inc)
Emptiness	$\leq$ EXPSPACE $\geq$ PSPACE	$\leq$ EXPSPACE $\geq$ PSPACE
Universality	undecidable	$\leq$ EXPSPACE $\geq$ PSPACE

Results on (avg,inc) Automata

	Nondet (avg,inc)	Det (avg,inc)	Det bounded width (avg,inc)	Det constant width (avg,inc)
Emptiness	$\leq$ EXPSPACE $\geq$ PSPACE	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE	PTIME
Universality	undecidable	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE	PTIME

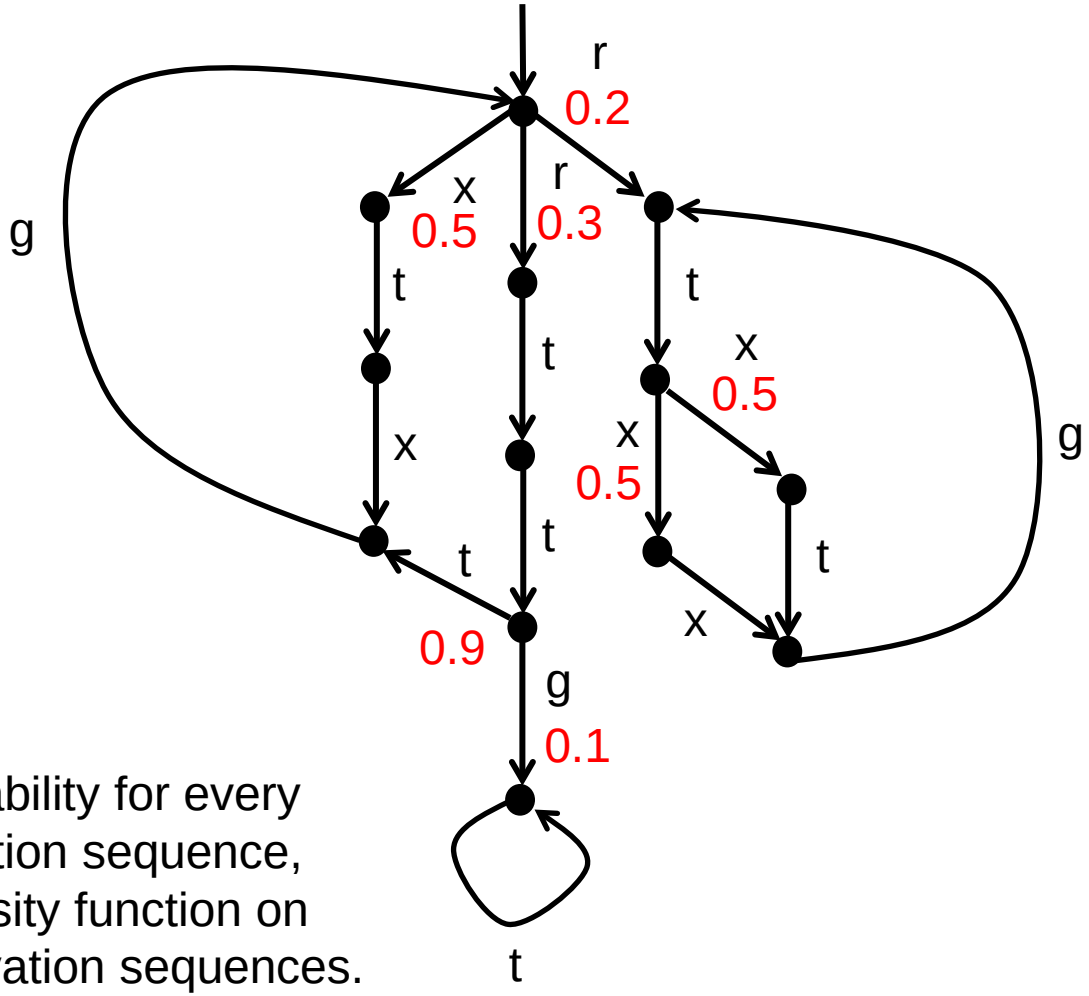
How many overlapping requests?

Results on (avg,inc) Automata

	Nondet (avg,inc)	Det (avg,inc)	Det bounded width (avg,inc)	Det constant width (avg,inc)
Emptiness	$\leq$ EXPSPACE $\geq$ PSPACE	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE	PTIME
Universality	undecidable	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE	PTIME

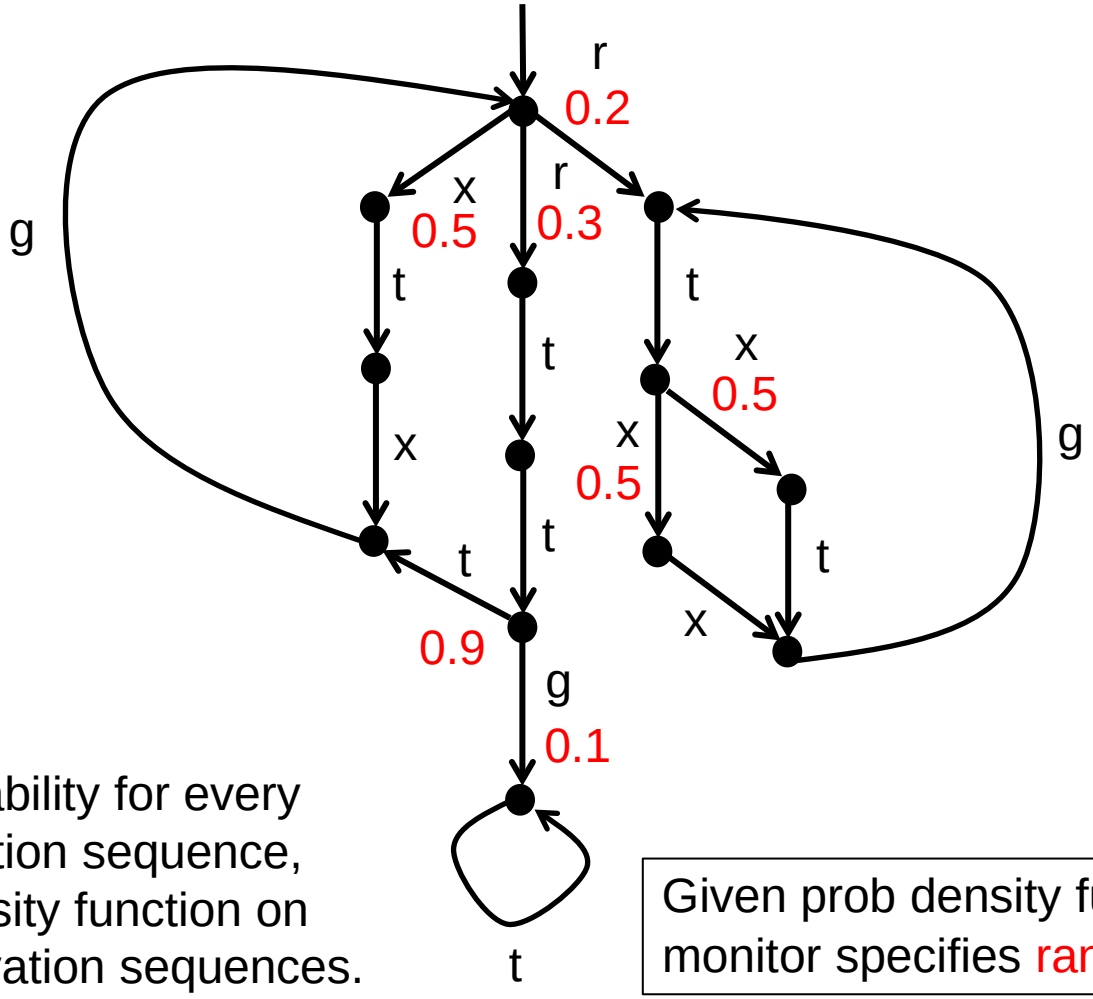
PTIME also for deterministic slaves of constant size (e.g. average response time).

Probabilistic System = Markov Chain



Defines probability for every finite observation sequence, and prob density function on infinite observation sequences.

Probabilistic Transition System = Markov Chain



Defines probability for every finite observation sequence, and prob density function on infinite observation sequences.

Given prob density function on Σ^ω, monitor specifies random variable V.

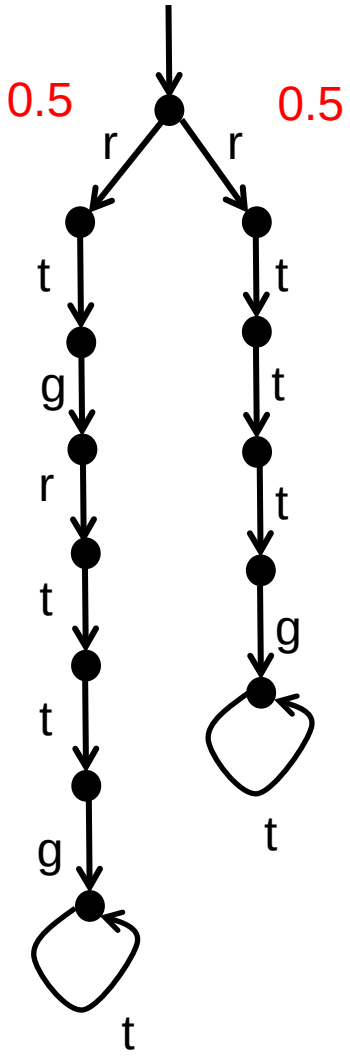
Probabilistic Analysis of Transition Systems

Given probabilistic system M and quantitative property A,

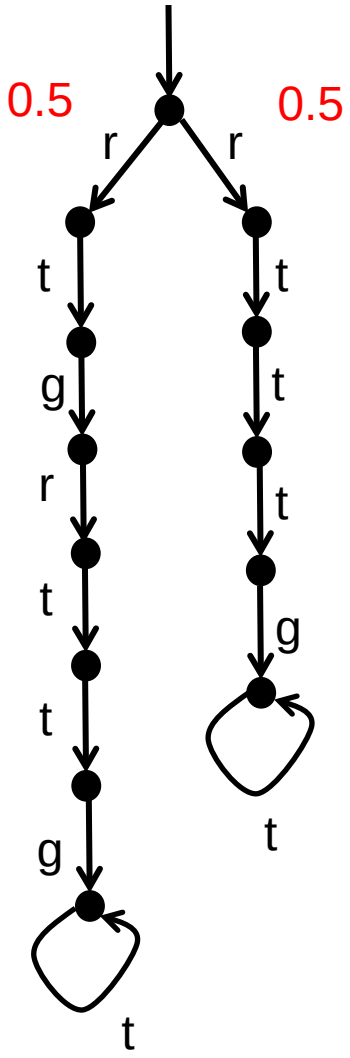
Q1. compute the expected value of V on the runs of M $\times$ A
[moment analysis]

Q2. compute the probability of V $\geq \lambda$ on the runs of M $\times$ A
[distribution analysis]

Probabilistic Example



Probabilistic Example

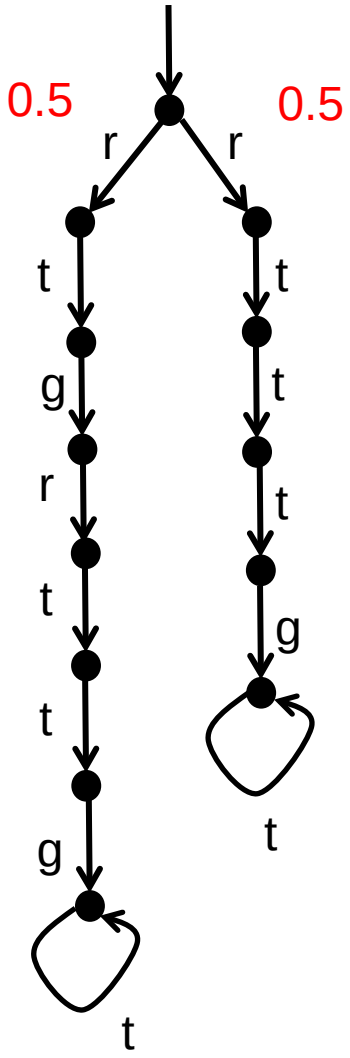


Expected maximal response time: 2.5

Prob of maximal response time at most 2: 0.5

Probabilistic analysis of (max,inc) automata

Probabilistic Example



Expected maximal response time: 2.5
Prob of maximal response time at most 2: 0.5

Probabilistic analysis of (max,inc) automata

Expected average response time: 2.25
Prob of average response time at most 2: 0.5

Probabilistic analysis of (avg,inc) automata

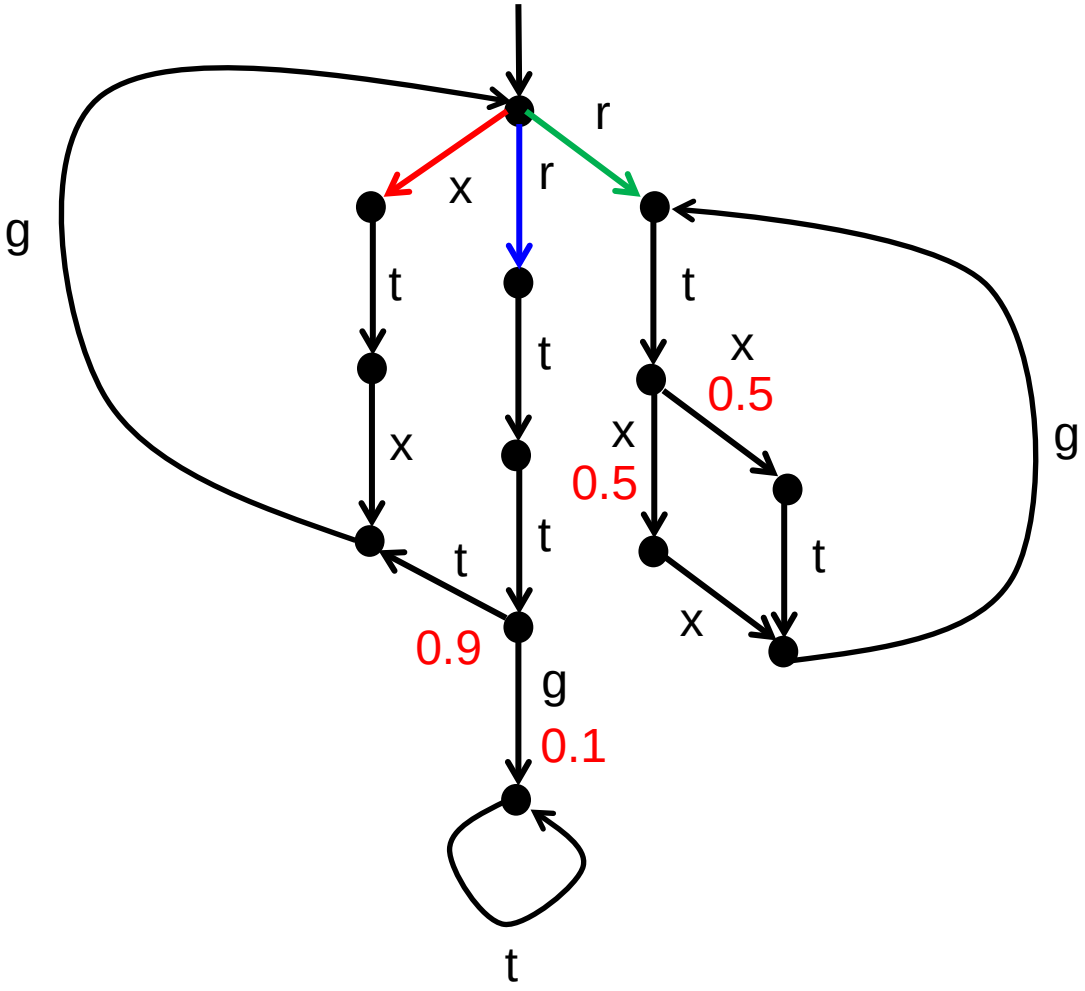
Results on (max,inc) Automata

	Nondet (max,inc)	Det (max,inc)
Emptiness	PSPACE	PSPACE
Universality	$\leq$ EXPSPACE $\geq$ PSPACE	PSPACE
Expectation	?	$\leq$ EXPSPACE $\geq$ PSPACE
Probability	?	$\leq$ EXPSPACE $\geq$ PSPACE

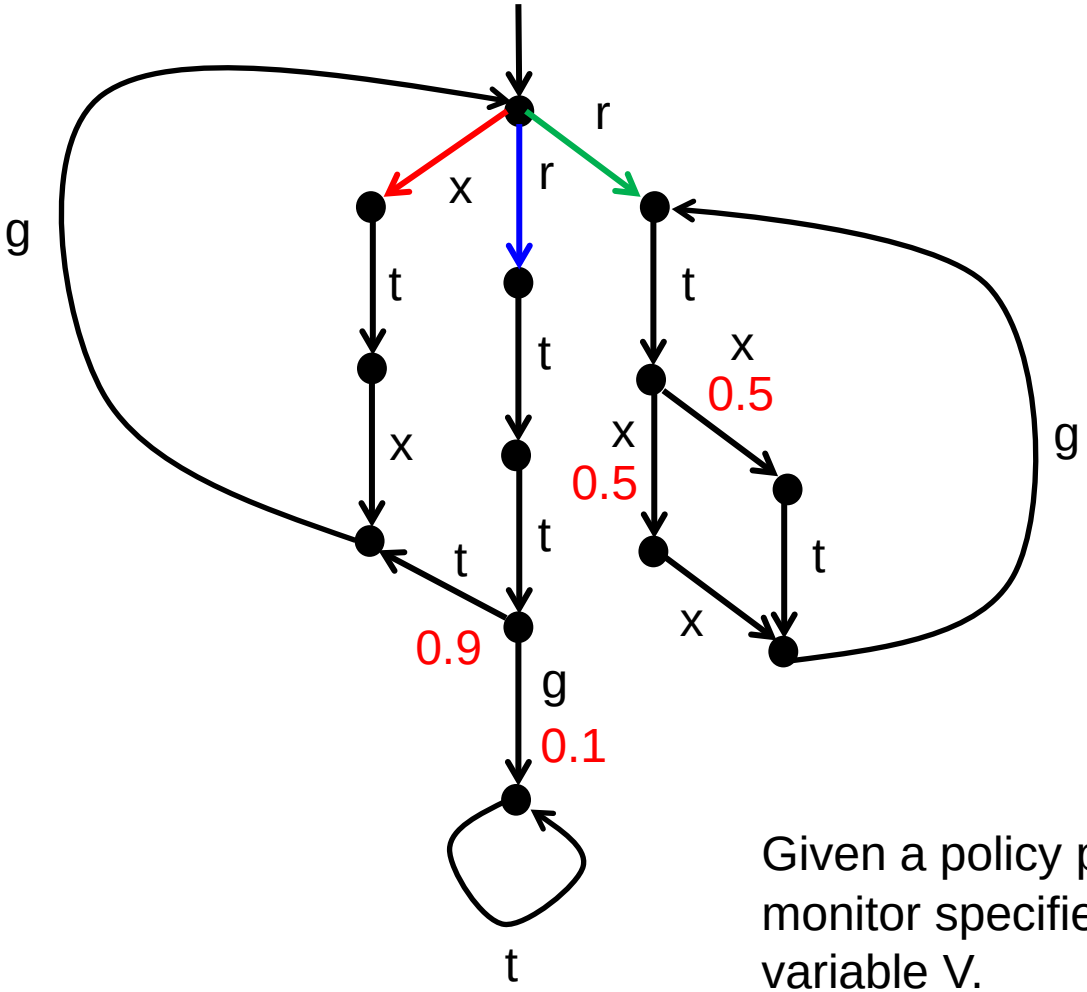
Results on (avg,inc) Automata

	Nondet (avg,inc)	Det (avg,inc)
Emptiness	$\leq$ EXPSPACE $\geq$ PSPACE	$\leq$ EXPSPACE $\geq$ PSPACE
Universality	undecidable	$\leq$ EXPSPACE $\geq$ PSPACE
Expectation	?	PTIME
Probability	?	PTIME

Markov Decision Process



Markov Decision Process

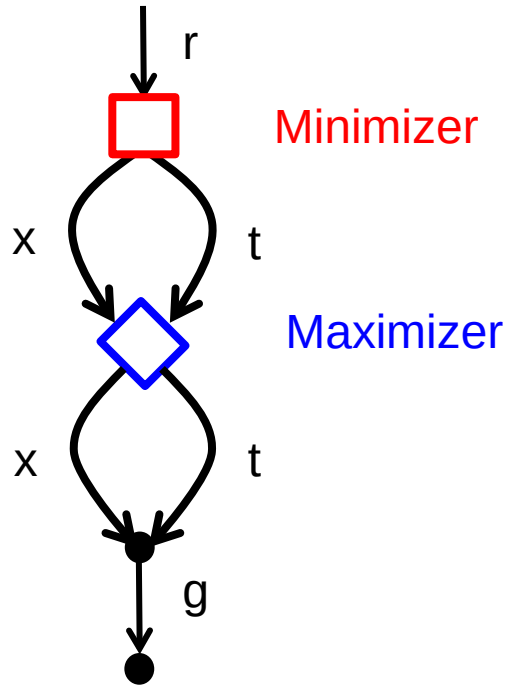


Given a policy p:$\Sigma^{\omega} \rightarrow$ { $,$,$ }, monitor specifies random variable V.

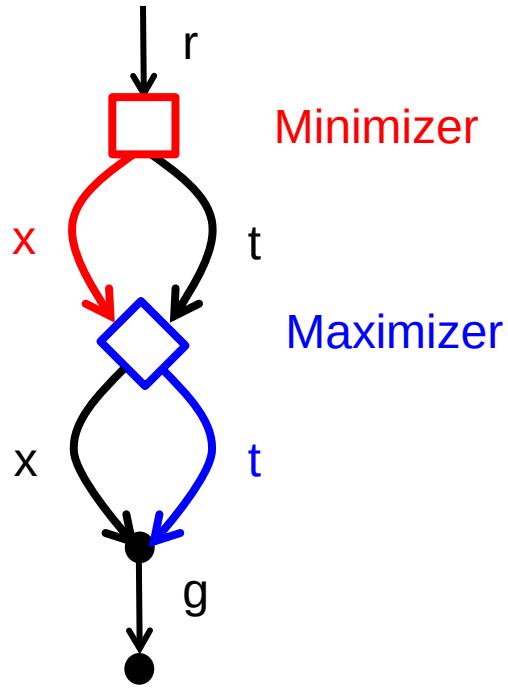
Many Open Questions …

E.g., given MDP M and deterministic (avg,inc) automaton A, compute the policy that maximizes the expected value of M on A (generalization of mean-payoff game).

Adversarial System = Graph Game

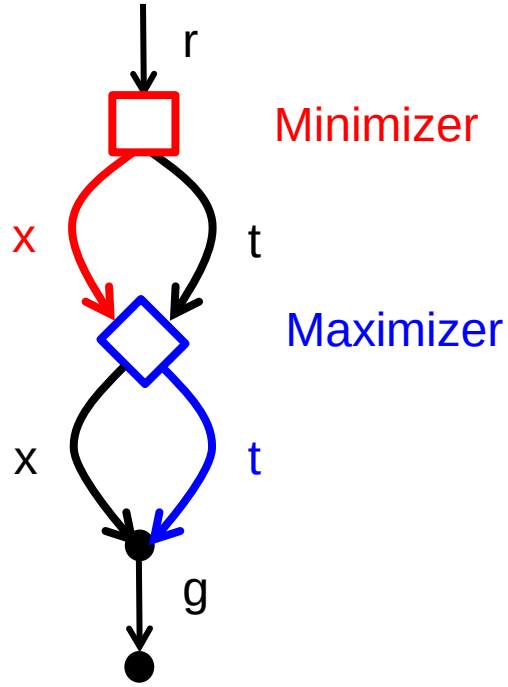


Adversarial System = Graph Game

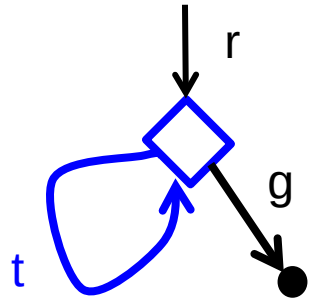


Optimal response time 1

Adversarial System = Graph Game

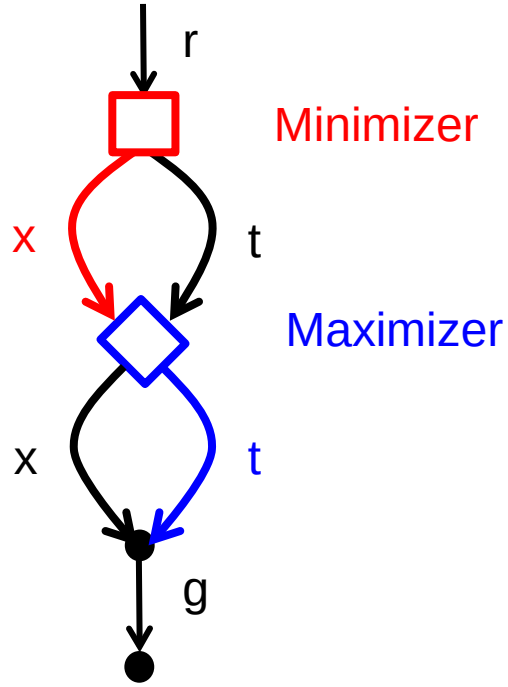


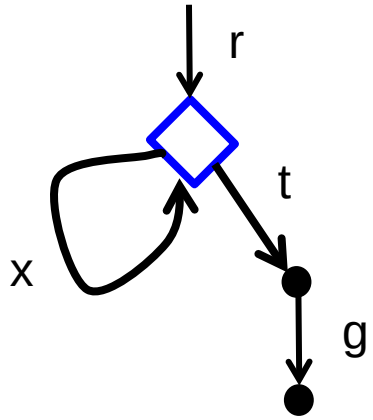
Optimal response time 1



Optimal response time ∞

Adversarial System = Graph Game

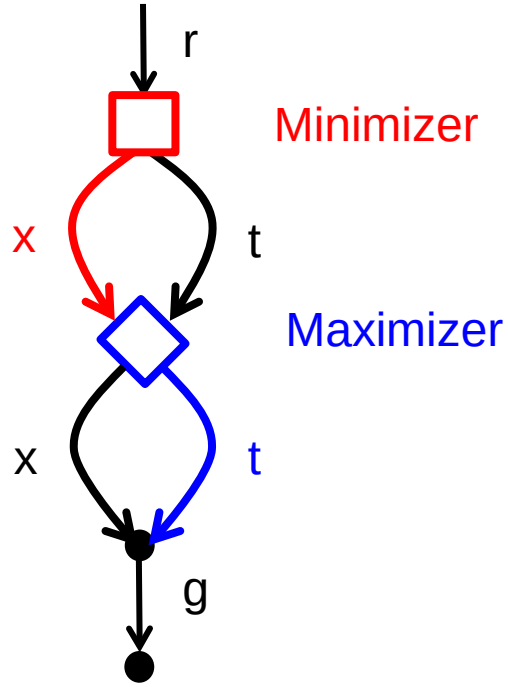


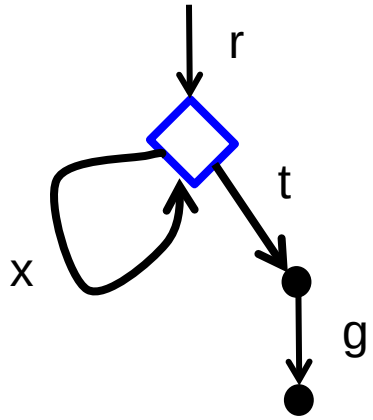


Optimal response time 1

Optimal response time 1

Adversarial System = Graph Game



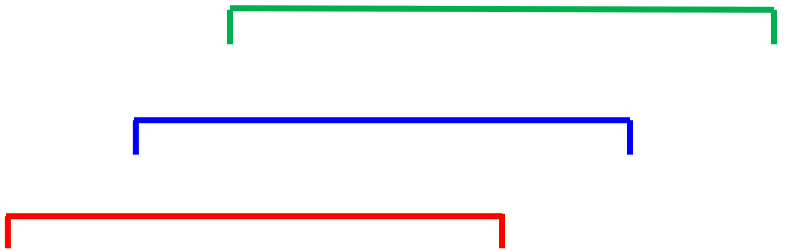


Optimal response time 1

Optimal response time 1

Streett objective $\square\diamondsuit(t \Rightarrow \square(r \Rightarrow \diamondsuit g))$
instead of Buchi objective $\square(r \Rightarrow \diamondsuit g)$

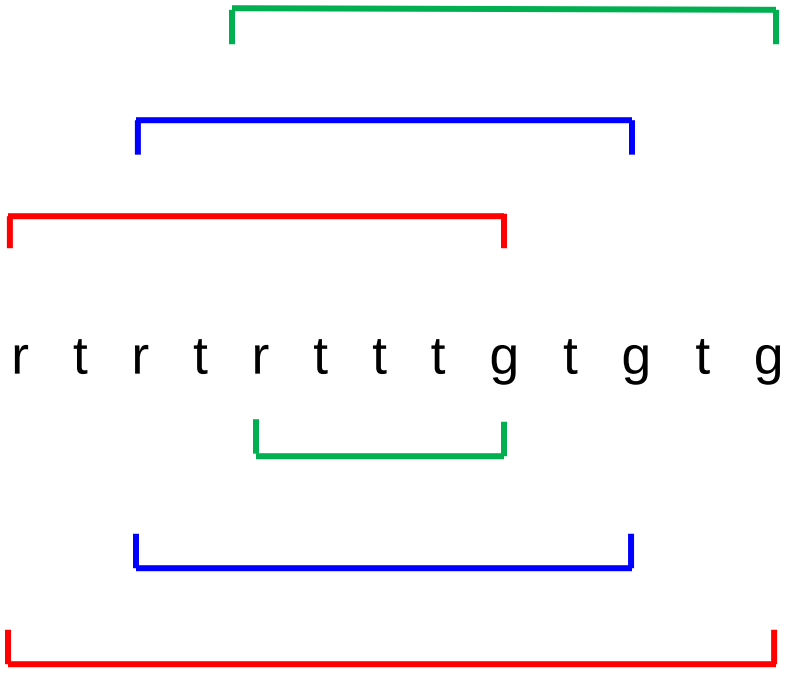
Matching Requests and Grants



5,5,5

r t r t r t t t g t g t g

Matching Requests and Grants

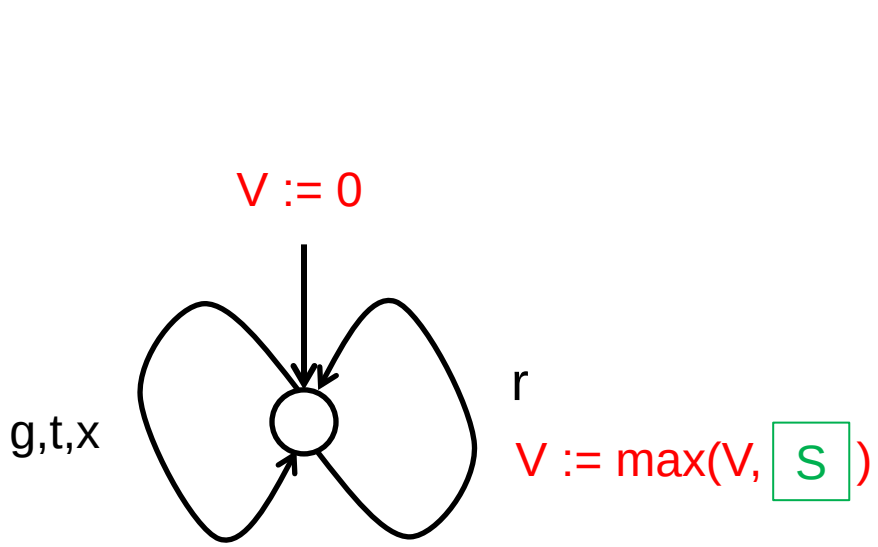


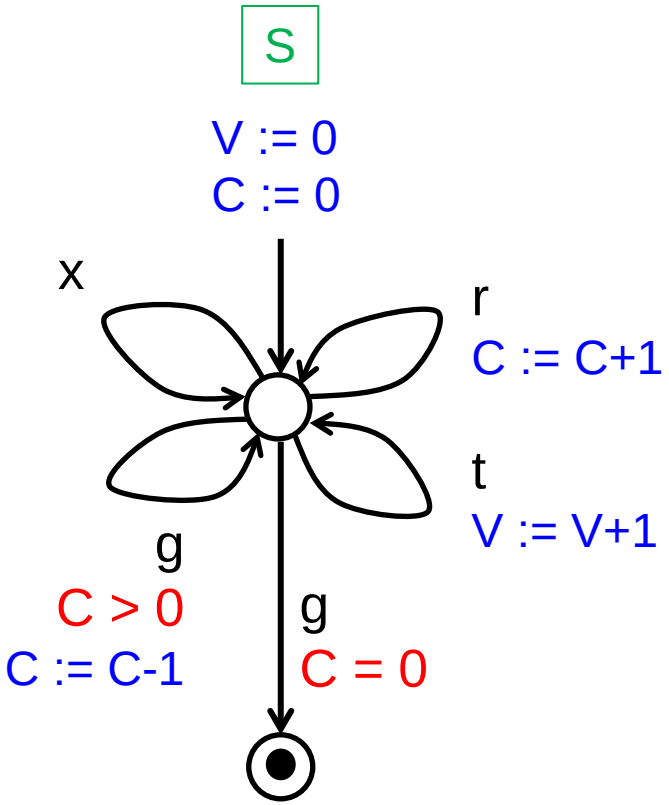
5,5,5

7,5,3

Quantitative pushdown monitors?

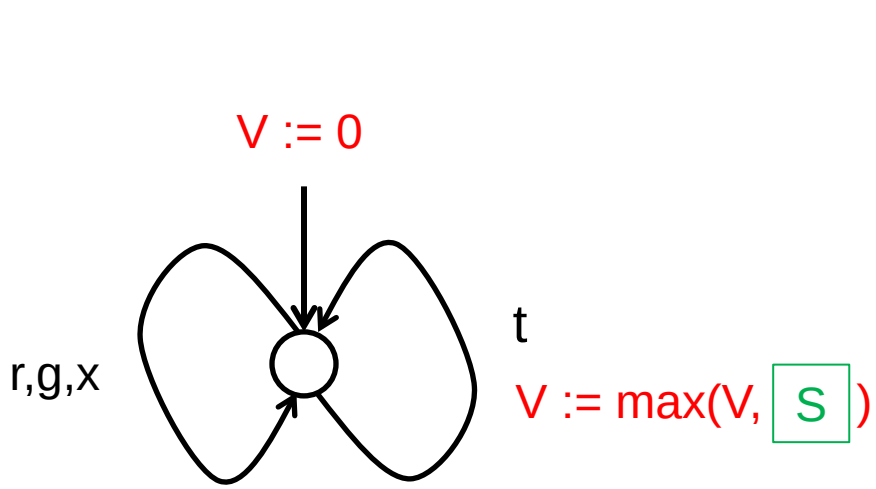
Counter Machine

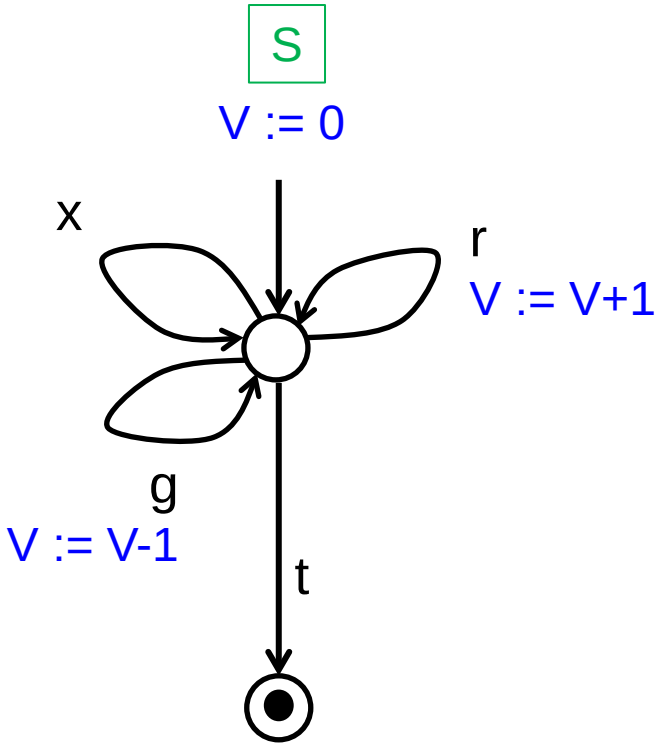




Emptiness for two counters is in general undecidable.

Counter Monitor





No test on counter values.

width = 1

Results on (max, inc+dec) Automata

	Nondet (max,inc+dec)	Det (max,inc+dec)
Emptiness	PSPACE	PSPACE
Universality	undecidable	undecidable
Expectation	undecidable	undecidable
Probability	undecidable	undecidable

Results on (avg, inc+dec) Automata

	Nondet (avg,inc+dec)	Det (avg,inc+dec)
Emptiness	?	?
Universality	undecidable	?
Expectation	?	PTIME
Probability	?	PTIME

Quantitative Monitors = Deterministic Nested Weighted Automata

Unbounded width allows for natural decomposition of specifications (incl. average response time).

More expressive and more succinct than flat weighted automata.

Emptiness decidable and sufficient for quantitative monitor verification (universality can be undecidable, even for constant width).

Probabilistic analysis polynomial for (avg, inc+dec) monitors.

References

Nested Weighted Automata: LICS 2015

Quantitative Automata under Probabilistic Semantics: LICS 2016