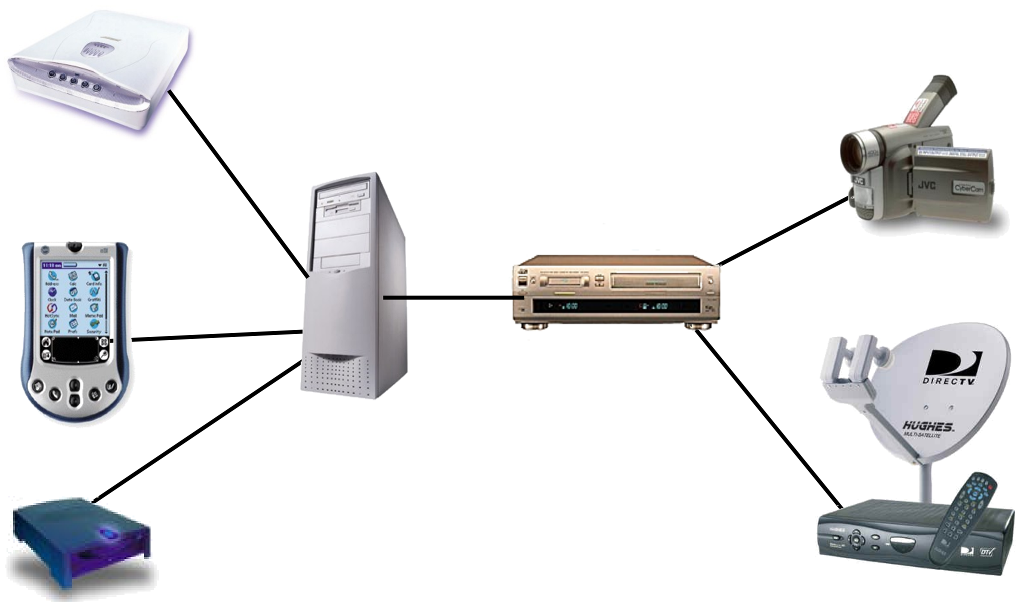
Probabilistic Timed Automata

Jeremy Sproston
Università di Torino

PaCo kick-off meeting, 23/10/2008

FireWire root contention protocol



- Leader election: create a tree structure in a network of multimedia devices
- Symmetric, distributed protocol
- Uses electronic coin tossing (symmetry breaker) and timing delays

FireWire root contention protocol

- If two nodes try to become root at the same time:
  - Both nodes toss a coin
  - If heads: node waits for a "long" time ($\geq$1590ns, $\leq$1670ns)
  - If tails: node waits for a "short" time ($\geq$760ns, $\leq$850ns)
- The first node to finish waiting tries to become the root:
  - If the other contending node is not trying to become the root (different results for coin toss), then the first node to finish waiting becomes the root
  - If the other contending node is trying to become the root (same result for coin toss), then repeat the probabilistic choice

FireWire root contention

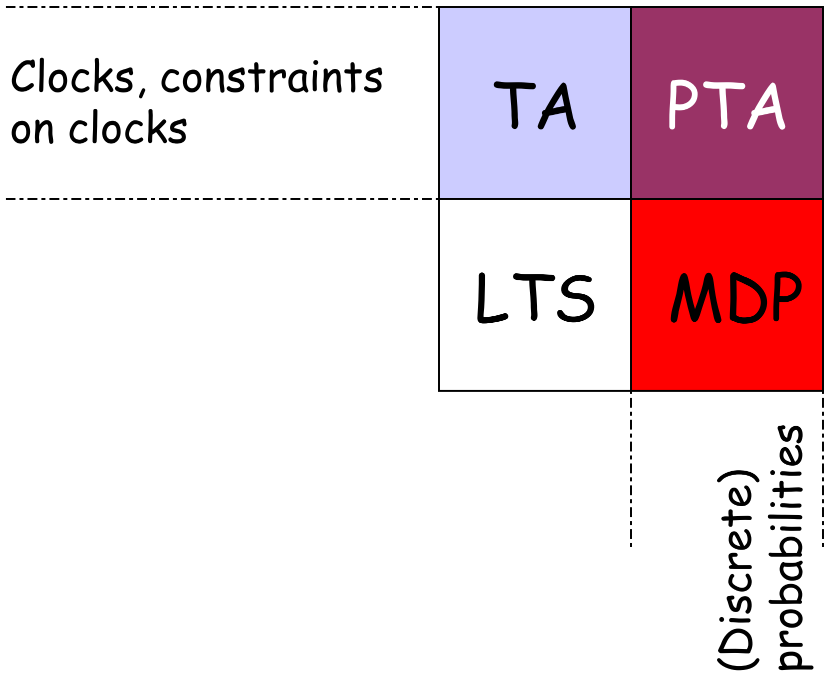
- Description of protocol:
  - Time
  - (Discrete) probability
  - Nondeterminism:
    - Exact time delays are not specified in the standard, only time intervals
- Probabilistic timed automata - formalism featuring:
  - Time
  - (Discrete) probability
  - Nondeterminism

PTA: other case studies

- IEEE 802.11 backoff strategy [KNS02]
  - Wireless Local Area Networks
- IEEE 802.15.4 CSMA/CA protocol [Fru06]
- IPv4 Zeroconf protocol [KNPS03]
  - Dynamic self-configuration of network interfaces
- Security applications [LMT04, LMT05]
- PC-mobile downloading protocol [ZV06]
- Publish-subscribe systems [HBGS07]

Probabilistic timed automata

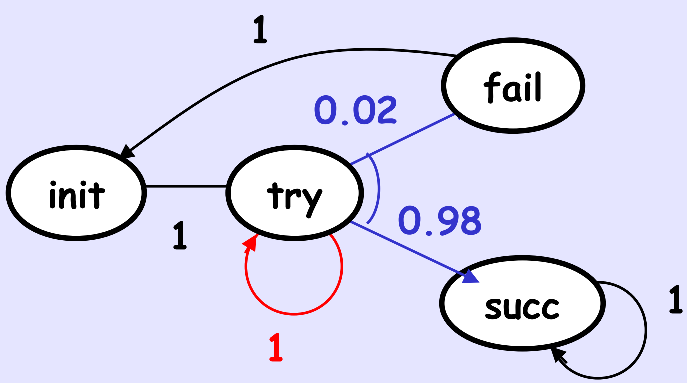
- Probabilistic timed automata:
  - An extension of Markov decision processes with clocks and constraints on clocks
  - An extension of timed automata with (discrete) probabilistic choice



Timed automata

- Timed automata [Alur & Dill'94]: formalism for timed + nondeterministic systems
  - Finite graph, clocks (real-valued variables increasing at same rate as real-time), constraints on clocks

Markov decision processes

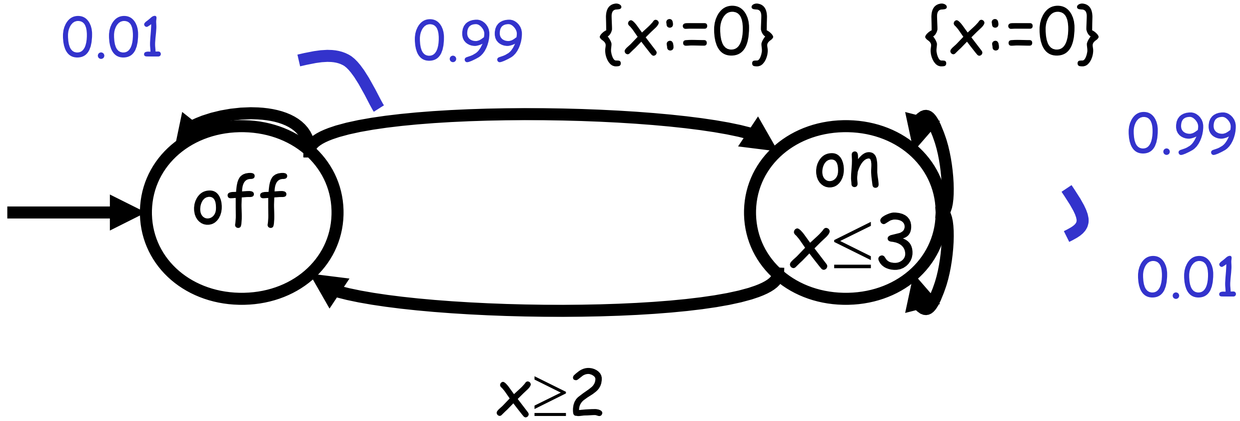


State-to-state transition:

1. Nondeterministic choice over the outgoing probability distributions of the source state
2. Probabilistic choice of target state according to the distribution chosen in step 1.

- Markov decision process: MDP = $(S, s_0, Steps)$:
  - S is a set of states with the initial state s_0
  - Steps: $S \rightarrow 2^{Dist(S)} \backslash \{\emptyset\}$ maps each state s to a set of probability distributions μ over S

Probabilistic timed automata



- Recall clocks: real-valued variables which increase at the same rate as real-time
- Clock constraints $CC(X)$ over set X of clocks:

$$g ::= x \sim c \mid g \wedge g$$

where

$x \in X$, $\sim \in \{<, \leq, \geq, >\}$ and c is a natural

Probabilistic timed automata

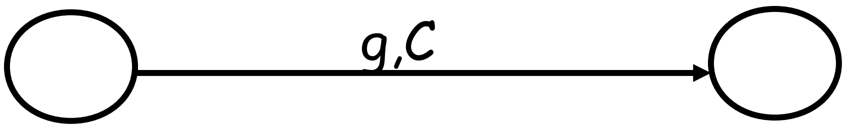
Formally, PTA = $(Q, q_0, X, Inv, prob)$:

- Q finite set of locations with q_0 initial location
- X is a finite set of clocks
- $Inv: Q \rightarrow CC(X)$ maps locations q to invariant clock constraints
- $prob \subseteq Q \times CC(X) \times Dist(2^X \times Q)$ is a probabilistic edge relation: yields the probability of moving from q to q', resetting specified clocks

Probabilistic timed automata

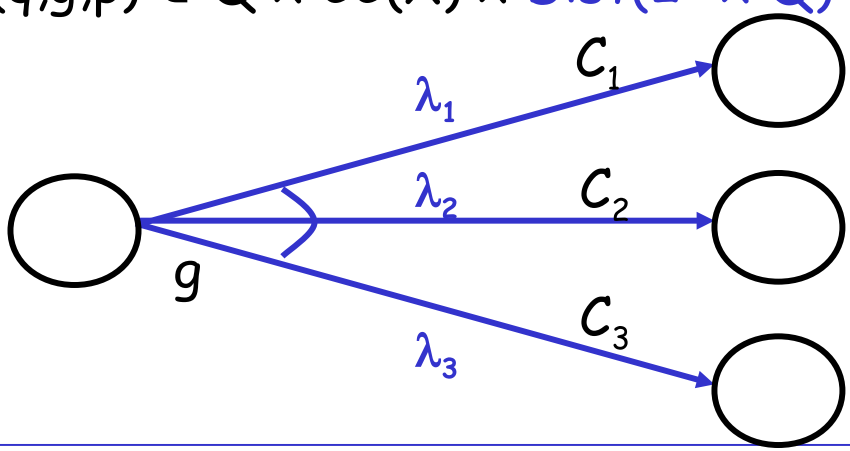
Discrete transition of timed automata:

$(q,g,C,q') \in Q \times CC(X) \times 2^X \times Q$



Discrete transition of probabilistic timed automata:

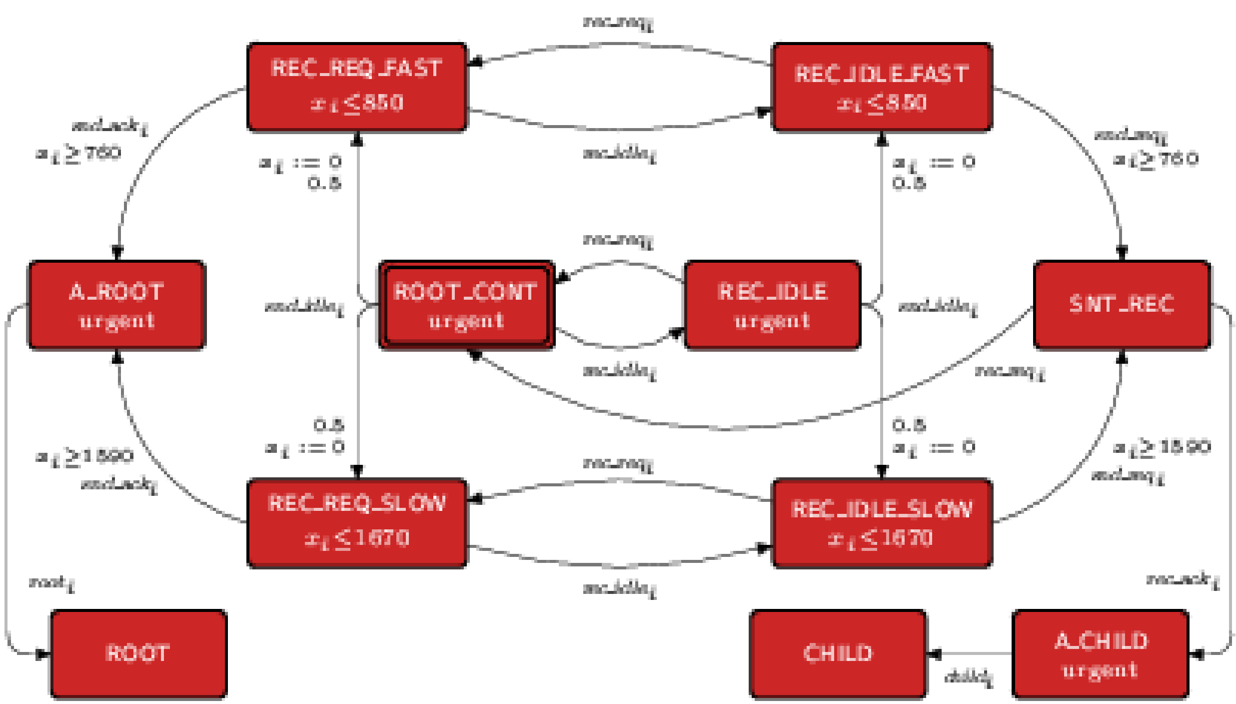
$(q,g,p) \in Q \times CC(X) \times \mathrm{Dist}(2^X \times Q)$



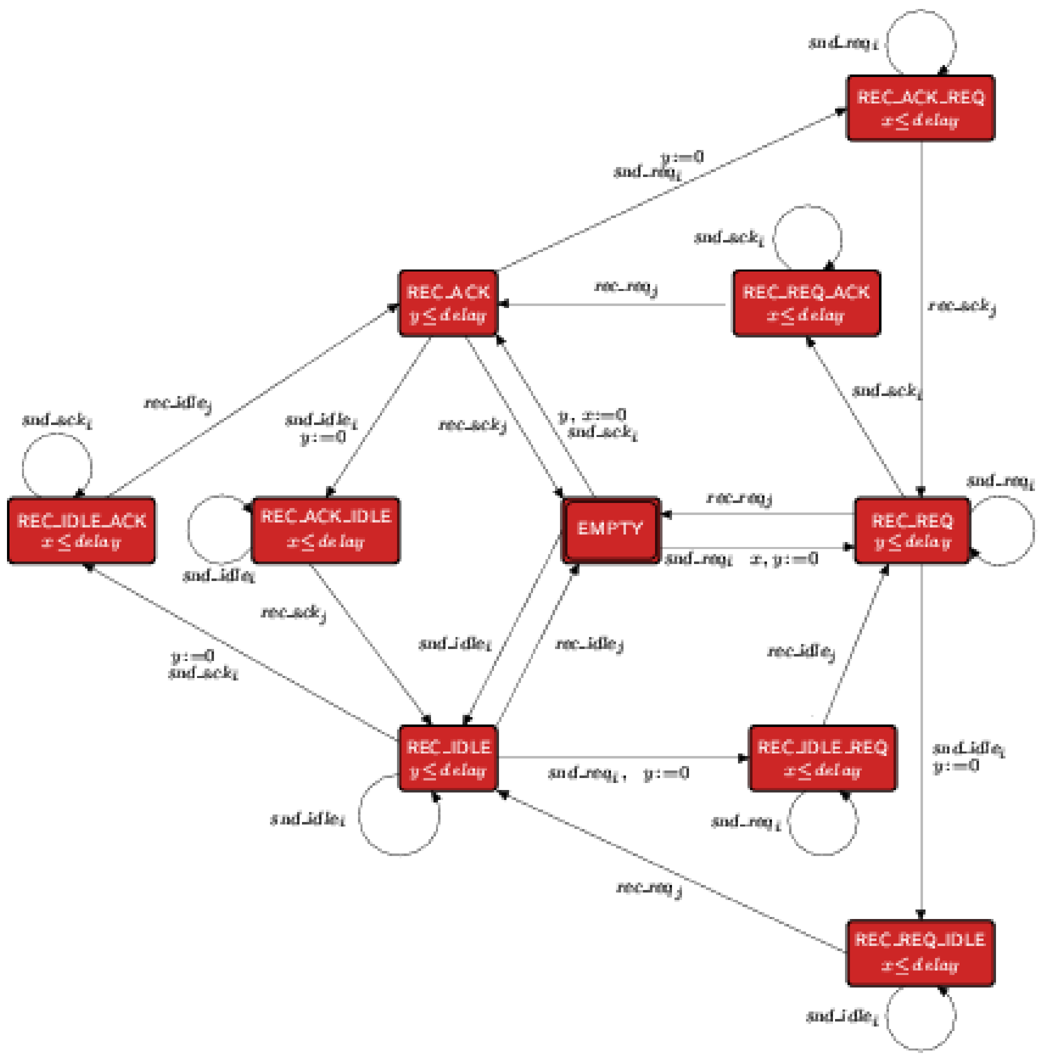
FireWire: node PTA

Modelling:

- Four PTA (2 nodes, 2 wires)



FireWire: wire PTA



Probabilistic Timed CTL

- To express properties such as:
  - "under any policy, with probability >0.98, the message is delivered within 5 ms"
- Choices for the syntax:
  - Time-bound (TCTL of [ACD93]):

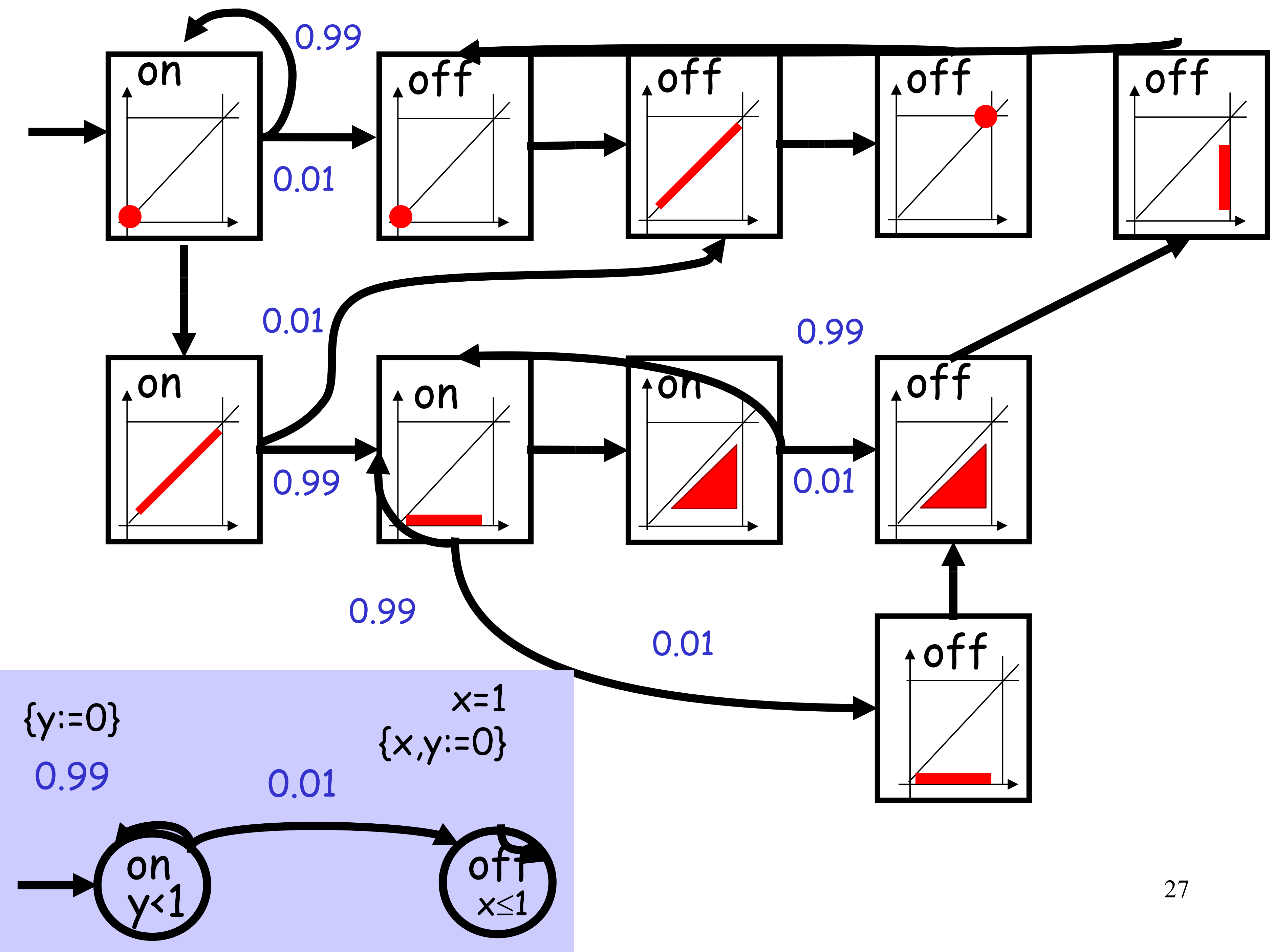
    $$P_{>0.98}[\ \Diamond_{\leq 5} \text{ delivered}]$$

  - Reset quantifier (TCTL of [HNSY94]):

    $$z.[P_{>0.98}[\ \Diamond\ (\text{delivered} \wedge z \leq 5)]$$

Model checking for PTA

- Common characteristics:
  - Semantics of a PTA is an infinite-state MDP, so construct a finite-state MDP
    - E.g., “region graph”
    - E.g., discrete-time semantics (for certain classes of PTA/properties, equivalent to continuous-time semantics)
  - Apply the algorithms for the computation of maximum/minimum reachability probabilities to the finite-state MDP



on
off
off
off
off
0.99
0.01
on
on
on
off
0.01
0.99
0.99
0.01
0.99
0.01
off
{y:=0}
0.99
0.01
x=1
{x,y:=0}
on
y<1
off
x≤1

Complexity of model checking PTA

- Model checking for PTA:
  - EXPTIME-algorithm [KNSS02]
  - Construct finite-state MDP: exponential in the encoding of the PTA
  - Run the polynomial time algorithm for model checking finite-state MDPs [BdA95]

Complexity of model checking PTA

- Comparison:
  - TCTL model checking (and reachability) for timed automata is PSPACE-complete [ACD93, AD94]
  - CTL model-checking problem for transition systems operating in parallel is PSPACE-complete [KVW00]
  - TATL (and alternating reachability) for timed games is EXPTIME-complete [HK99,HP06]

TA with one or two clocks

- Restricting the number of clocks in timed automata [LMS04]:
  - Reachability for one-clock timed automata is NLOGSPACE-complete
  - Reachability for two-clock timed automata is NP-hard
  - Model checking "deadline" properties for one-clock timed automata is PTIME-complete