Logic & Formal Verification

Kim G. Larsen
CISS – Aalborg University
DENMARK

Quantitative & Compositional Model Checking

Kim G. Larsen

CISS – Aalborg University

DENMARK

Prior Knowledge

- Finite Automata
- Transition Systems
- Temporal Logics
- Fixpoint Theory (a'la Tarski)
- Model Checking
- Proces Algebra
- Bisimilarity
- Timed Automata
- Markov Chains

Topics

Behavioural Model $\mathcal{M}$
(Transition Systems)

—————

Logical Specification $\mathcal{S}$
(Temporal Logic)

$$\mathcal{M}_1 | \ldots | \mathcal{M}_n \models \mathcal{S}$$

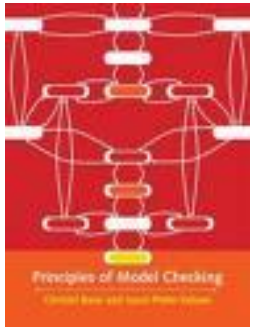
- Quantitative Model Checking
- On-the-fly Model Checking
- Compositional Model Checking
- Modal Transition Systems

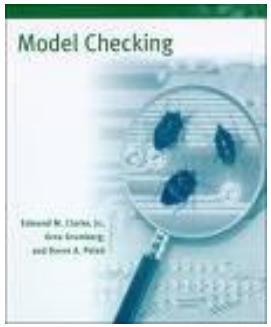
Model Checking

Ed Clarke

Joseph Sifakis

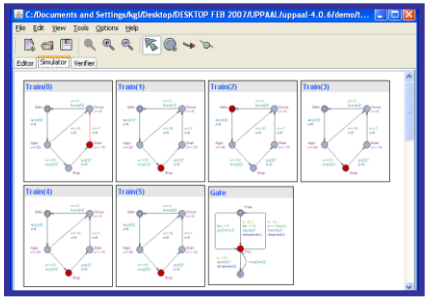
Alen Emerson





Turing Award Winners 2007

QUANTITATIVE Model Checking

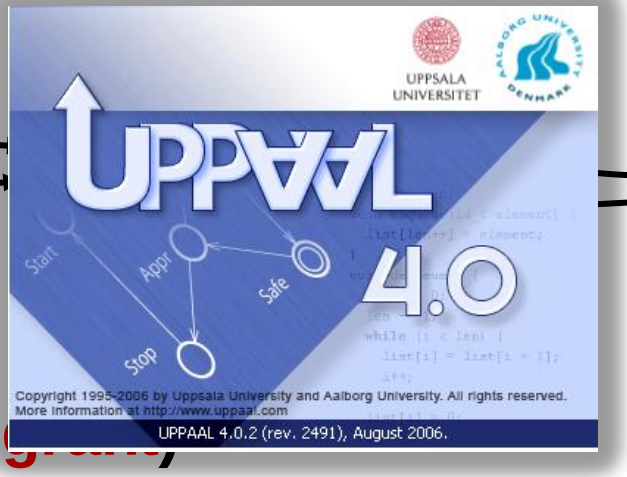


System Description

Time

Cost

Probability



No!
Debugging Information

Yes
Prototypes
Executable Code
Test sequences

Requirement

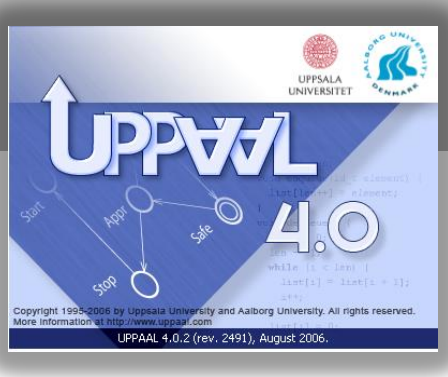
$A\Box(\ req \Rightarrow A\Diamond\ grant)$

$A\Box(\ req \Rightarrow A\Diamond_{t<30s}\ grant)$

$A\Box(\ req \Rightarrow A\Diamond_{t<30s,c<5\$}\ grant)$

$A\Box(\ req \Rightarrow A\Diamond_{t<30s\ ,\ p>0.90}\ grant)$

Overview of Talk



- Finite State Model Checking (review)
  - Kripke Structures, CTL
  - Fixpoint Characterization
- Timed Model Checking
  - Timed Automata
  - TCTL
- Weighted Model Checking
  - Weighted Timed Automata
  - WCTL
- Probabilistic Model Checking
  - Markov Chains, MDPs
  - PCTL
  - Probabilistic Timed Automata
  - Stochastic Timed Automata
- Timed Controller Synthesis
  - Timed Game Automata
- Conclusion

Classic

Cora

SMC

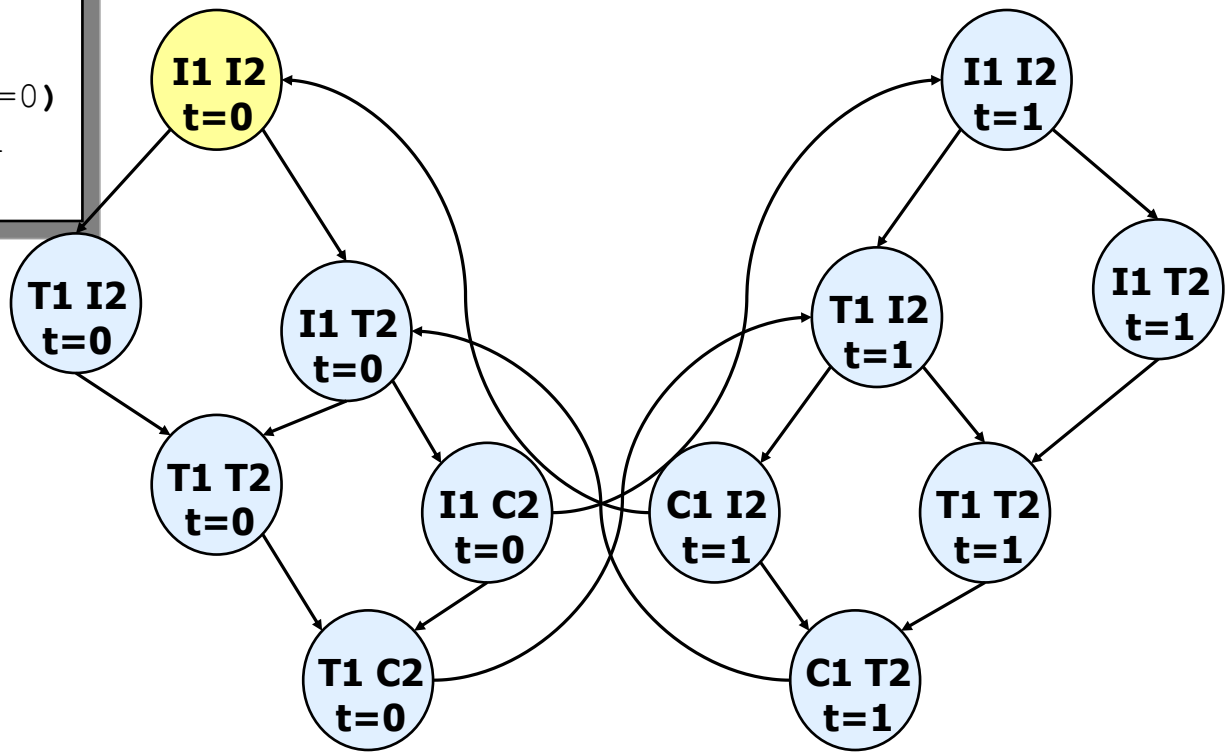
Tiga

Model Checking

Finite State

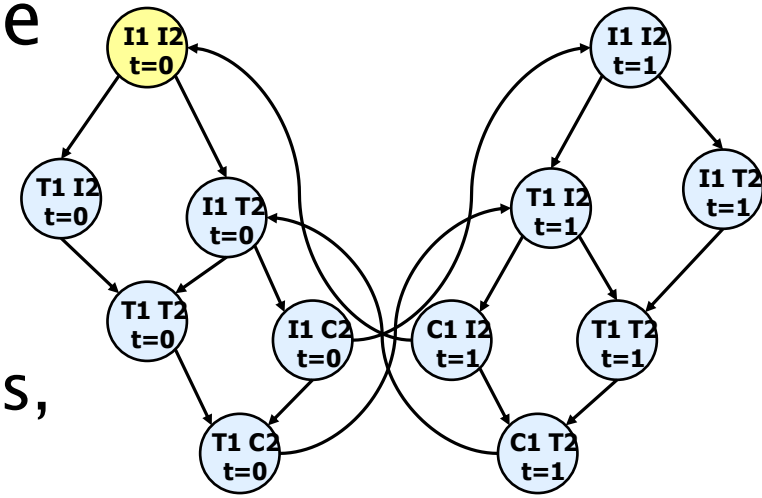
From Programs to Kripke Structures

```
P1 :: while True do
      T1 : wait(turn=1)
         C1 : turn:=0
           endwhile
          ||
P2 :: while True do
      T2 : wait(turn=0)
         C2 : turn:=1
           endwhile
```



Models = Kripke Structure

- A Kripke structure is a triple
  $M = (S, R, Lab)$
  where
  - S is a non-empty set of states,
  - $R \subseteq S \times S$ is a total relation on S,
  - $Lab : S \rightarrow 2^{AP}$, assigns to each state $s \in S$ the atomic propositions $Lab(s)$ that are valid in s.



Computation Tree Logic

Clarke, Emerson 1980

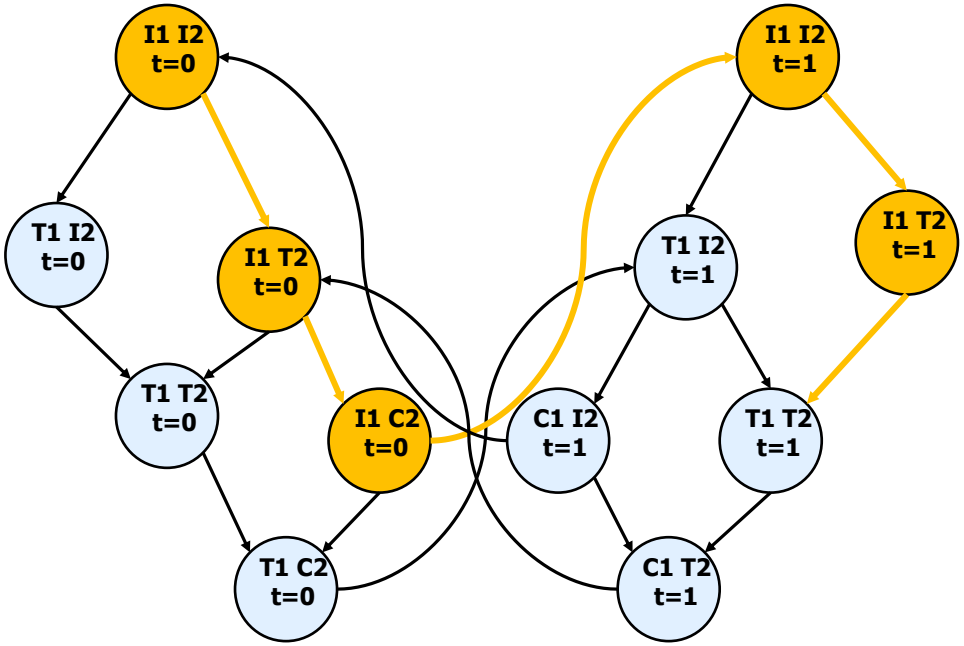
$$\phi ::= p \mid \neg \phi \mid \phi \wedge \phi \mid$$

$$\mathbf{EX}\,\phi \mid \mathbf{A}[\,\phi\ \mathbf{U}\ \phi] \mid \mathbf{E}[\,\phi\ \mathbf{U}\ \phi\,]$$

- **E** (for some path)
- **A** (for all path)
- **X** (next state)
- **U** (until)

Path

- A path is a sequence σ of states
  $s_0, s_1, s_2, ..$
  such that
  $(s_i, s_{i+1}) \in R$
  for all $i \geq 0$.



$P_M(s)$ denotes the set of path out of s.

Formal Semantics

(satisfaction relation $\models$)

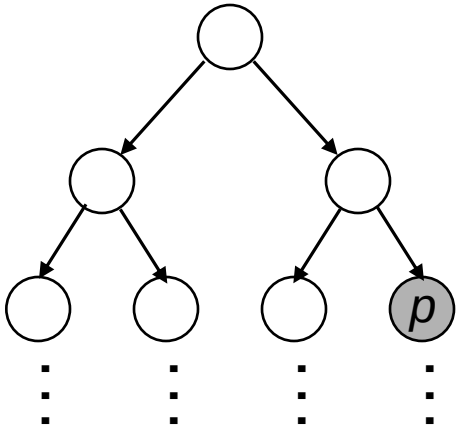
$s \models p$	iff	$p \in Lab(s)$
$s \models \neg\phi$	iff	$\neg (s \models \phi)$
$s \models \phi \wedge \psi$	iff	$(s \models \phi)$ and $(s \models \psi)$
$s \models EX\phi$	iff	$\exists \sigma \in P_M(s).\ \sigma[1] \models \phi$
$s \models E[\phi\ U\ \psi]$	iff	$\exists \sigma \in P_M(s).\ (\exists\ j \geq 0.\ \sigma[j] \models \psi \wedge (\forall\ k<j.\ \sigma[k] \models \phi))$
$s \models A[\phi\ U\ \psi]$	iff	$\forall \sigma \in P_M(s).\ (\exists\ j \geq 0.\ \sigma[j] \models \psi \wedge (\forall\ k<j.\ \sigma[k] \models \phi))$

Derived Operators

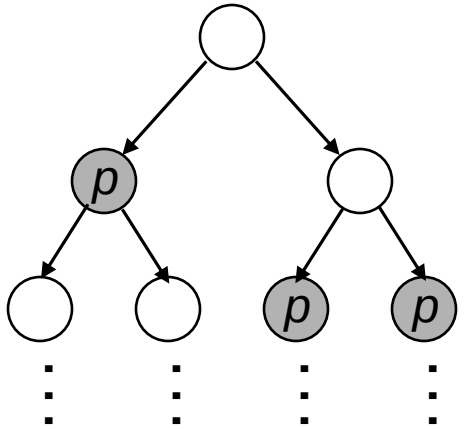
$$\mathsf{EF}\,\phi \equiv \mathsf{E}\,[\mathrm{true}\,\mathsf{U}\,\phi] \qquad \textit{possible}$$

$$\mathsf{AF}\,\phi \equiv \mathsf{A}\,[\mathrm{true}\,\mathsf{U}\,\phi]. \qquad \textit{inevitable}$$

EF p



AF p



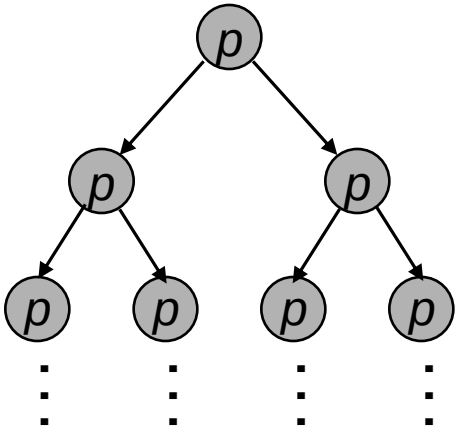
Derived Operators (cnt).

$$\mathsf{EG}\,\phi \equiv \neg\,\mathsf{AF}\,\neg\,\phi$$ *potentially always*

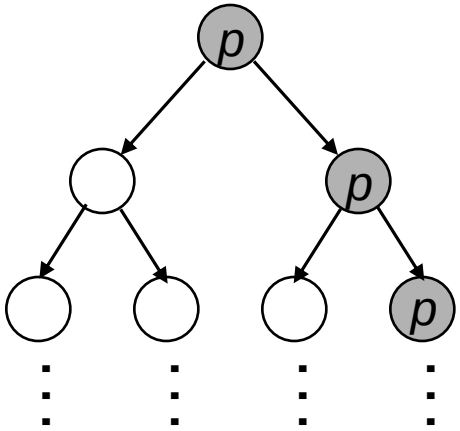
$$\mathsf{AG}\,\phi \equiv \neg\,\mathsf{EF}\,\neg\,\phi$$ *always*

$$\mathsf{AX}\,\phi \equiv \neg\,\mathsf{EX}\,\neg\,\phi.$$

AG p



EG p



Theorem

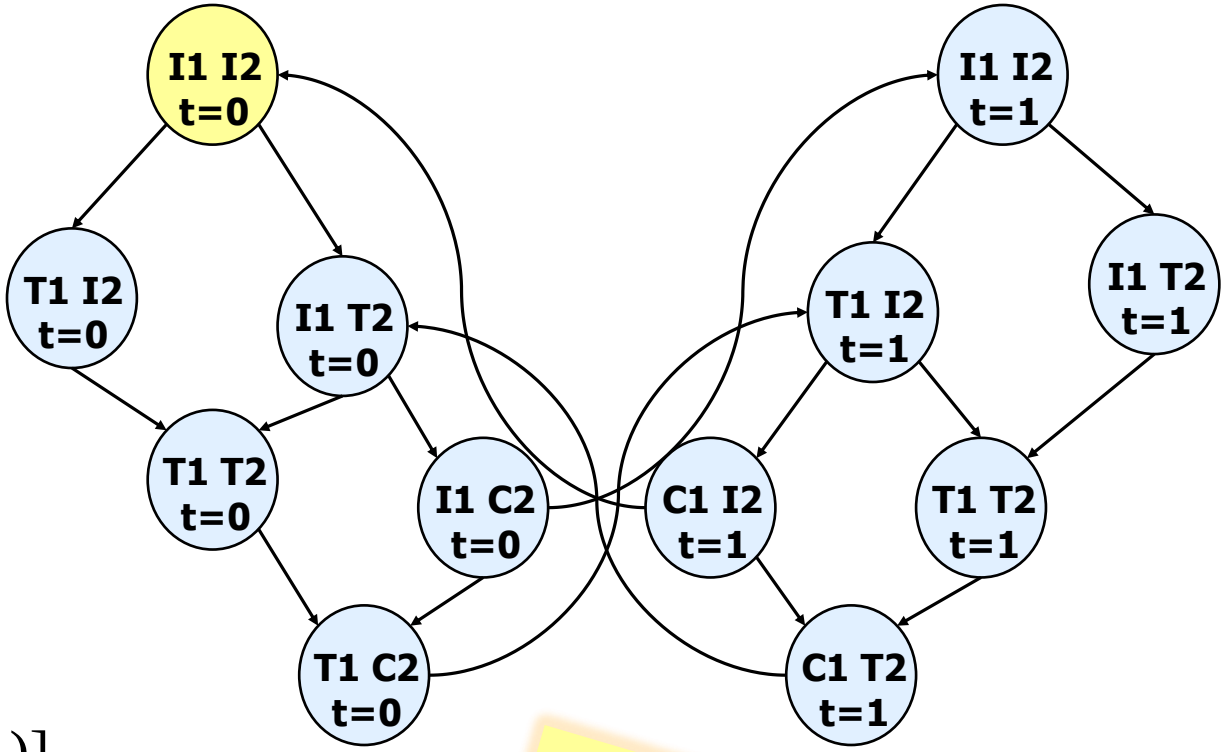
All operators are derivable from

EX f
EG f
E[f U g]

and boolean connectives

$$A[f\ U\ g] \equiv \neg E[\neg g U (\neg f \wedge \neg g)] \wedge \neg EG \neg g$$

Properties of MUTEX



$AG\ \neg(C_1 \wedge C_2)$

$AG[\ T_1 \Rightarrow AF(C_1)]$

$EG[\neg C_1]$

$AG[C_1 \Rightarrow A[C_1\ U\ (\neg C_1 \wedge A[\neg C_1\ U\ C_2])]]$

HOW to DECIDE IN GENERAL

Fixpoint Characterizations

$$\mathsf{EF}\, p \quad \equiv \quad p \vee \mathsf{EXEF}\, p$$

or let A **be the set of states satisfying** EF p **then**

$$\mathsf{A} \quad \equiv \quad p \vee \mathsf{EXA}$$

in fact A **is the smallest such set (the least fixpoint)**

Tarski's Fixpoint Theory

- Let τ be a function $2^S \rightarrow 2^S$
- Say τ is *monotonic* when

$$x \subseteq y \text{ implies } \tau(x) \subseteq \tau(y)$$

- Fixed point of τ is y such that

$$\tau(y) = y$$

- If τ monotonic, then it has
  - least fixed point $\mu y.\ \tau(y)$
  - greatest fixed point $\nu y.\ \tau(y)$

Iterative Fixpoint Computation

- Suppose S is finite
  - The least fixed point $\mu Y.\ \tau(Y)$ is the limit of

$$\text{false} \subseteq \tau(\text{false}) \subseteq \tau(\tau(\text{false})) \subseteq \cdots$$

  - The greatest fixed point $\nu Y.\ \tau(Y)$ is the limit of

$$\text{true} \supseteq \tau(\text{true}) \supseteq \tau(\tau(\text{true})) \supseteq \cdots$$

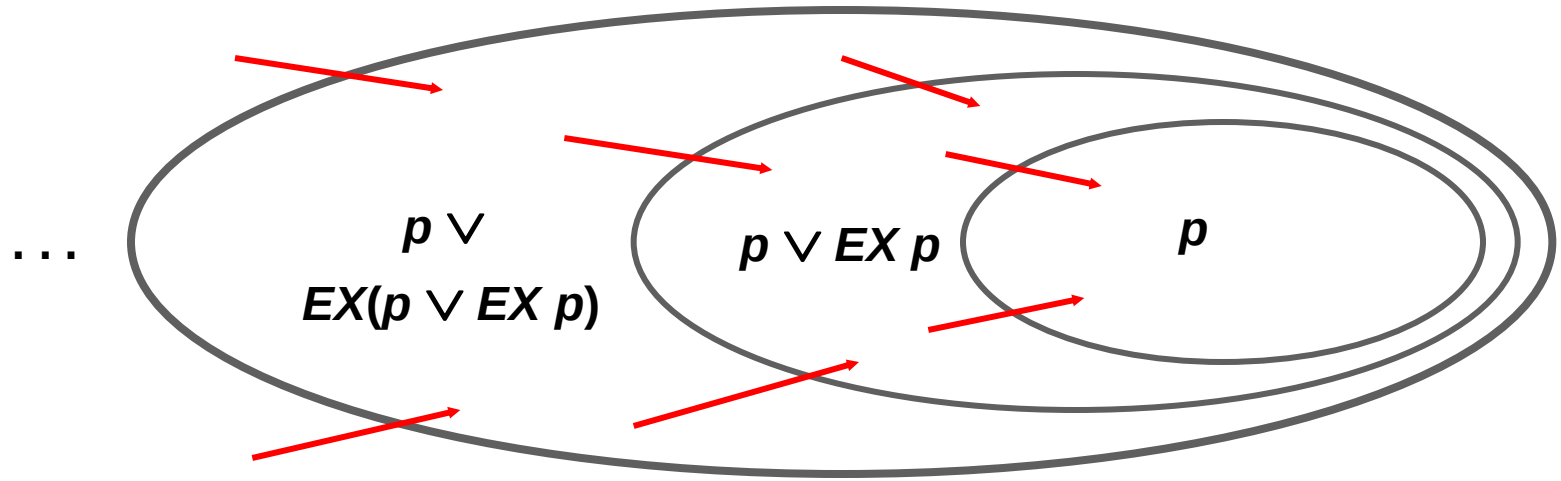
Note, since S is finite, convergence is finite

Example: *EF p*

- *EF p* is characterized by

$$EF\ p = \mu y.(p \vee EX\ y)$$

- Thus, it is the limit of the increasing series...

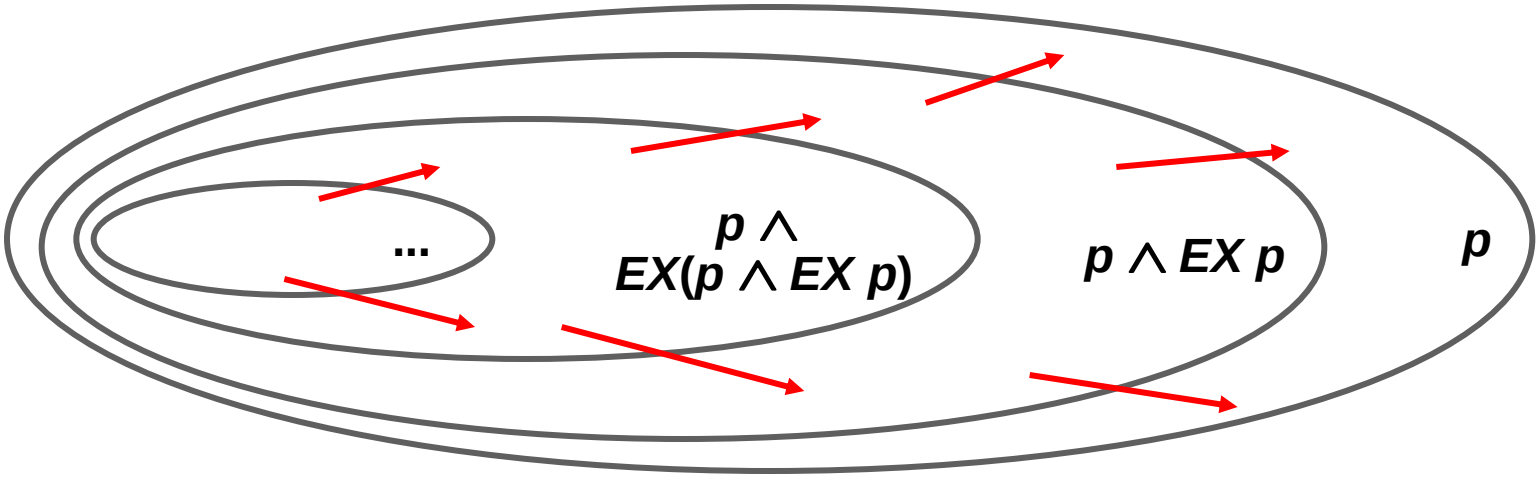


Example: *EG p*

- *EG p* is characterized by

$$EG\ p = \nu y.\ (p \wedge EX\ y)$$

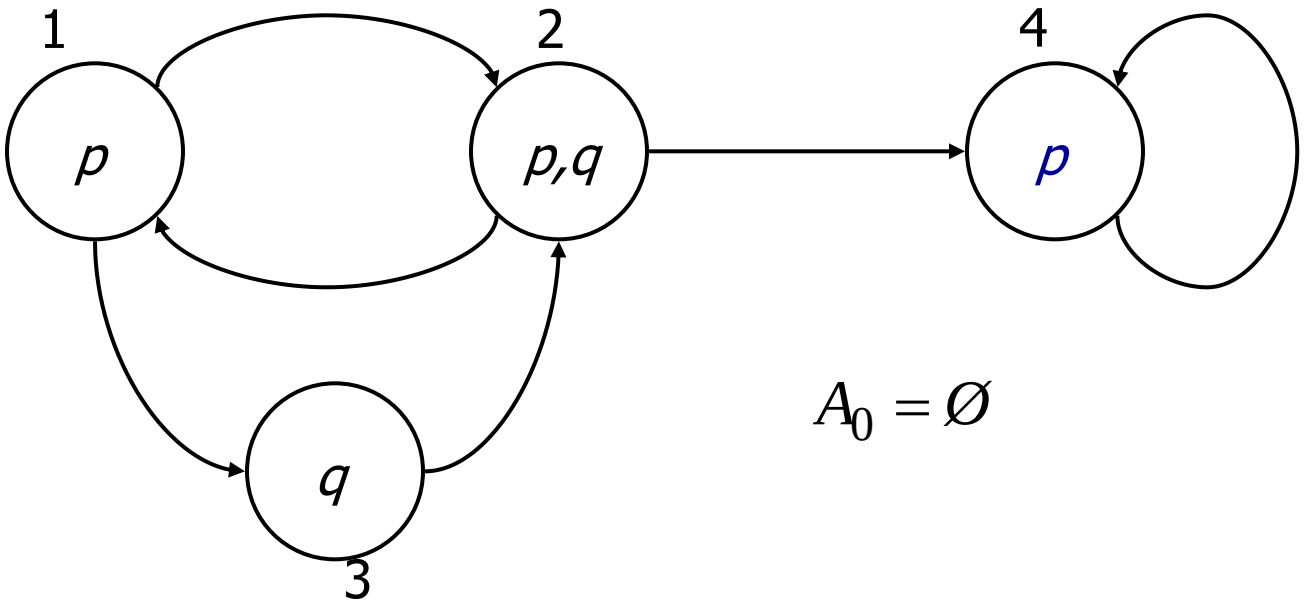
- Thus, it is the limit of the decreasing series...



Example, cnt.

EF *q*

$$EF\, q = \mu y.(q \vee EX\, y)$$

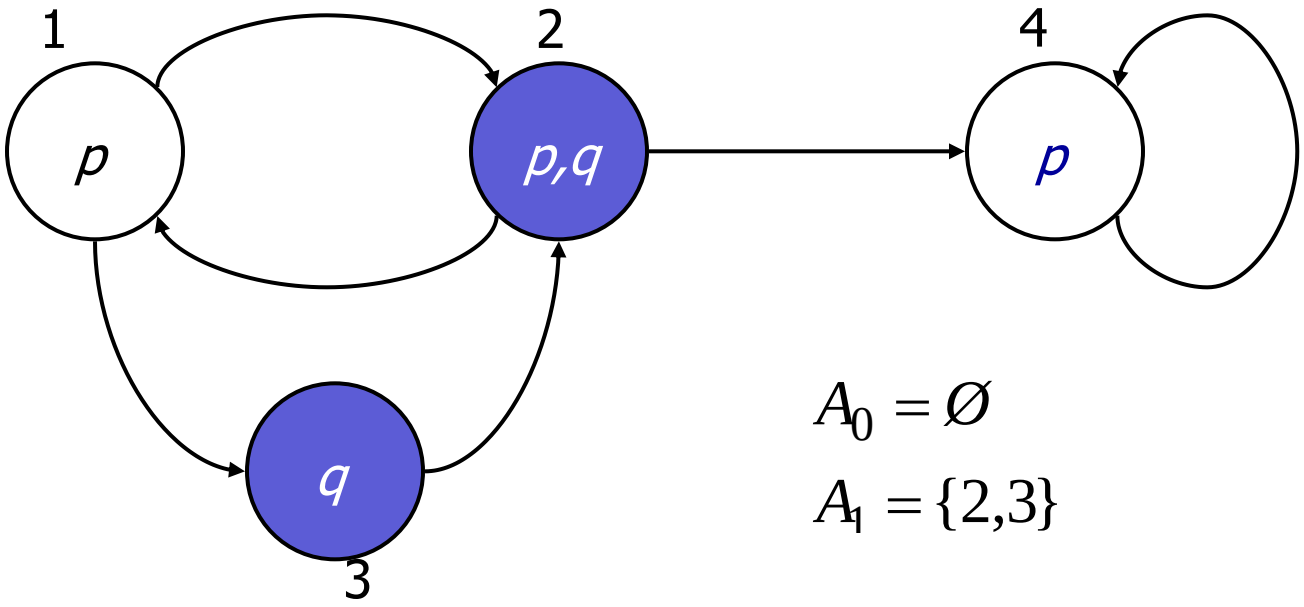


$$A_0 = \emptyset$$

Example, cnt.

EF *q*

$$EF\, q = \mu y.(q \vee EX\, y)$$



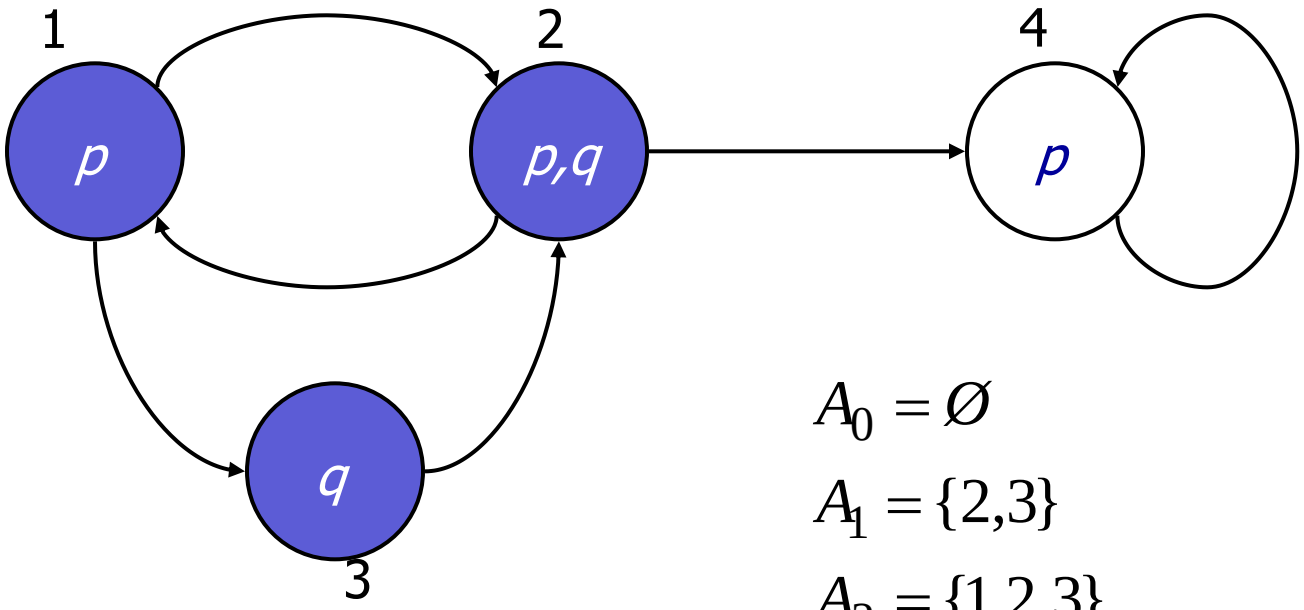
$$A_0 = \emptyset$$

$$A_1 = \{2,3\}$$

Example, cnt.

EF *q*

$$EF\, q = \mu y.(q \vee EX\, y)$$



$$A_0 = \emptyset$$

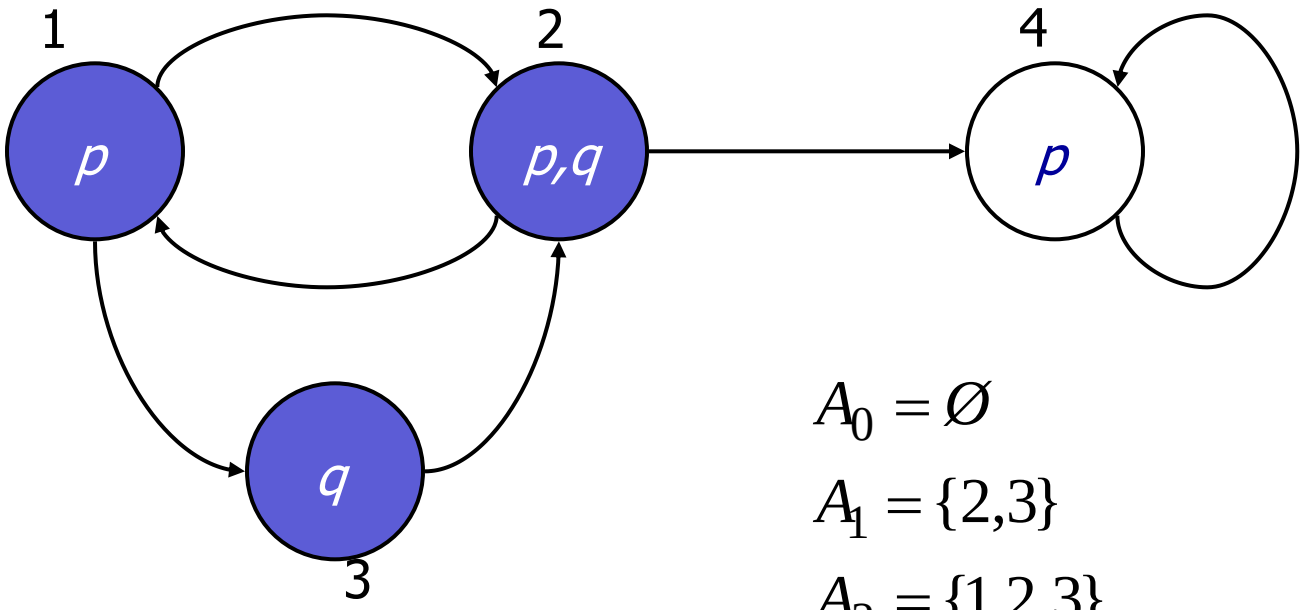
$$A_1 = \{2,3\}$$

$$A_2 = \{1,2,3\}$$

Example, cnt.

EF *q*

$$EF\, q = \mu y.(q \vee EX\, y)$$



$$A_0 = \emptyset$$

$$A_1 = \{2,3\}$$

$$A_2 = \{1,2,3\}$$

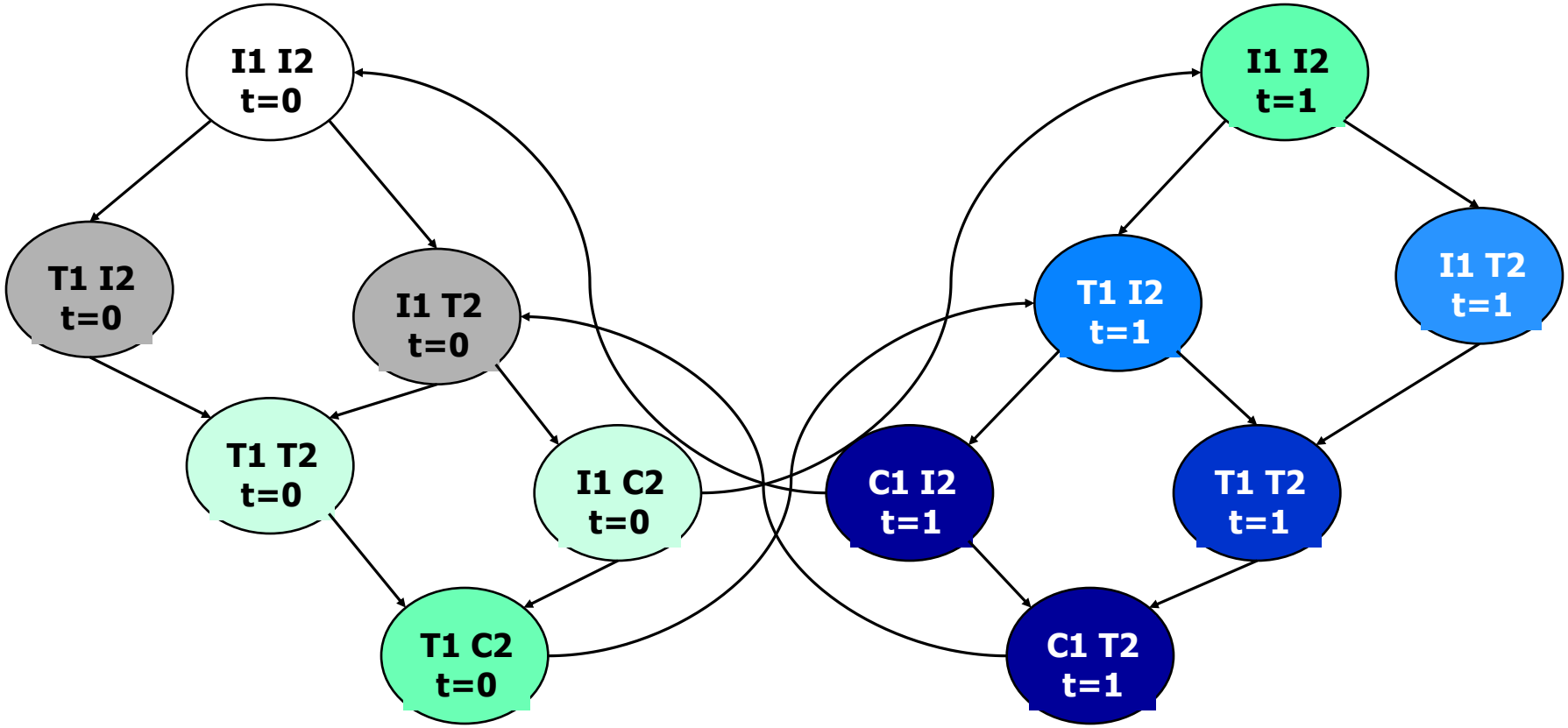
$$A_3 = \{1,2,3\}$$

Remaining Operators

$$
\begin{aligned}
AF\ p &= \mu y.(p \vee AX\ y) \\
AG\ p &= \nu y.(p \wedge AX\ y) \\
E(p\,U\,q) &= \mu y.(q \vee (p \wedge EX\ y)) \\
A(p\,U\,q) &= \mu y.(q \vee (p \wedge AX\ y))
\end{aligned}
$$

Properties of MUTEX

$$AF(C1) = \mu y.\ (\ C1 \vee AX\ y\ )$$



Complexity

The worst-case time complexity of checking whether a system-model *sys* satisfies the CTL-formula ϕ is $O(|S_{sys}|^2 \times |\phi|)$

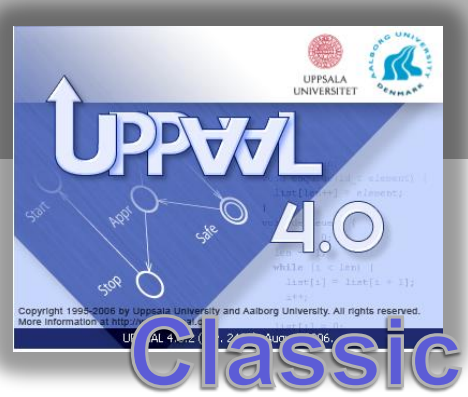
However S_{sys} may be EXPONENTIAL in number of parallel components!

--

FIXPOINT COMPUTATIONS may be carried out using ROBDD's (Reduced Ordered Binary Decision Diagrams)

Bryant, 86

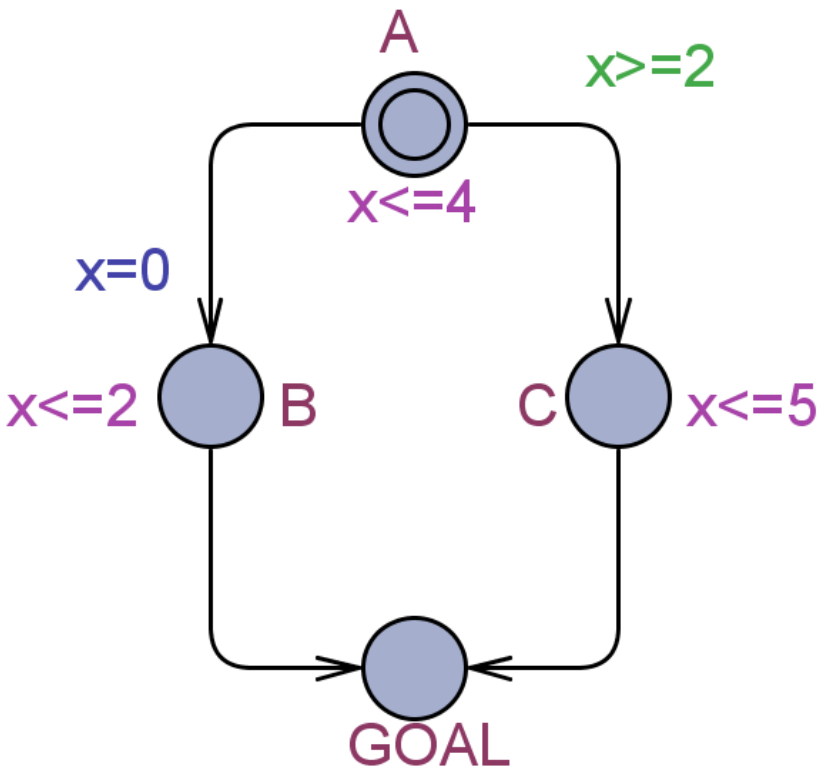
Timed CTL



AALBORG UNIVERSITY DENMARK

Timed Automata

Clocks **x**
Guards
Resets
Invariant



States = (location,clock-values)
e.g. (A , x=3.5)

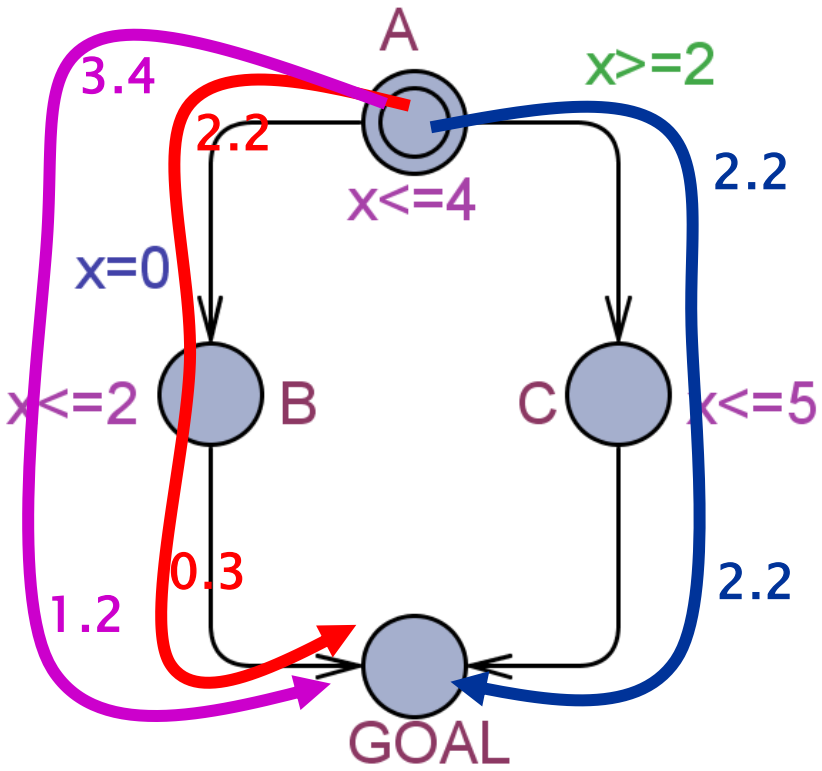
Transitions:

Delay
(A , x=3.5) $\rightarrow_{0.2}$ (A , x=3.7)

Discrete
(A , x=3.7) → (B , x=0)

Timed Automata

Clocks **x**
Guards
Resets
Invariant



States = (location,clock-values)
e.g. (A , x=3.5)

Transitions:

Delay
$(A , x=3.5) \rightarrow_{0.2} (A , x=3.7)$

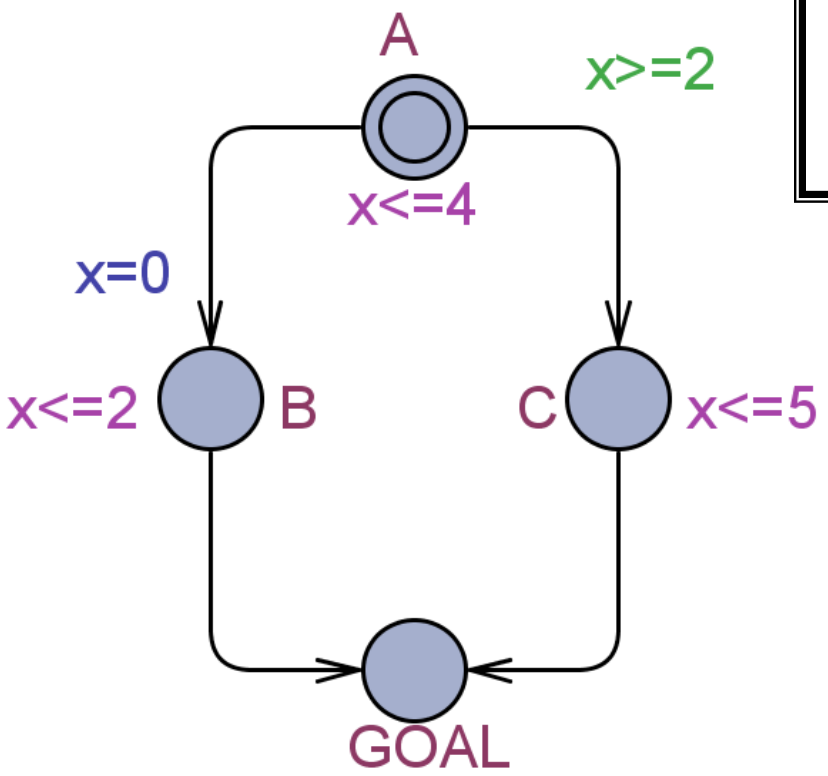
Discrete
$(A , x=3.7) \rightarrow (B , x=0)$

$A[\ true\ \ U_{t \leq 10}\ \ GOAL]$

$AF_{t \leq 10}\ GOAL$

TCTL

Clocks **x**
Guards
Resets
Invariant



$$\phi ::= p \mid \neg \phi \mid \phi \wedge \phi \mid$$

$$EX\ \phi \mid A[\ \phi\ U_{t \leq T}\ \phi] \mid$$

$$E[\ \phi\ U_{t \leq T}\ \phi\ ]$$

$$A[\ true\ U_{t \leq 10}\ GOAL]$$

$$AF_{t \leq 10}\ GOAL$$

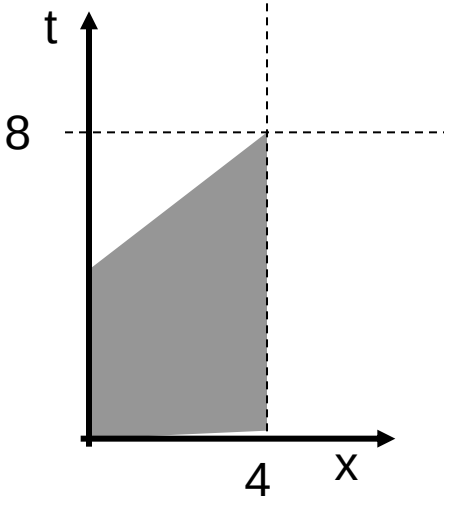
TCTL Model Checking

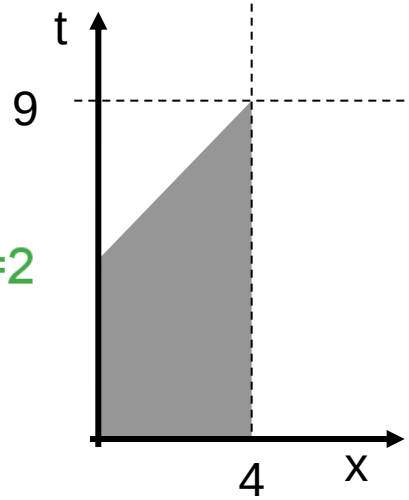
Fixpoint Computation

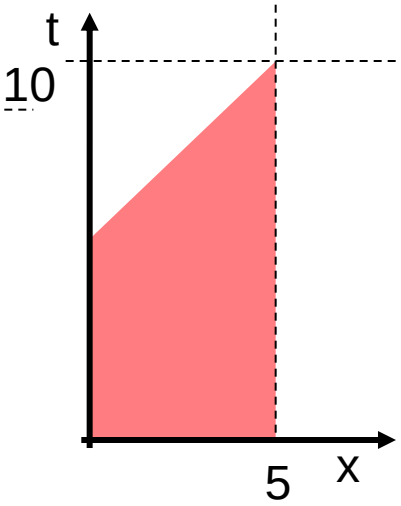
$AF_{t\leq 10}$ GOAL

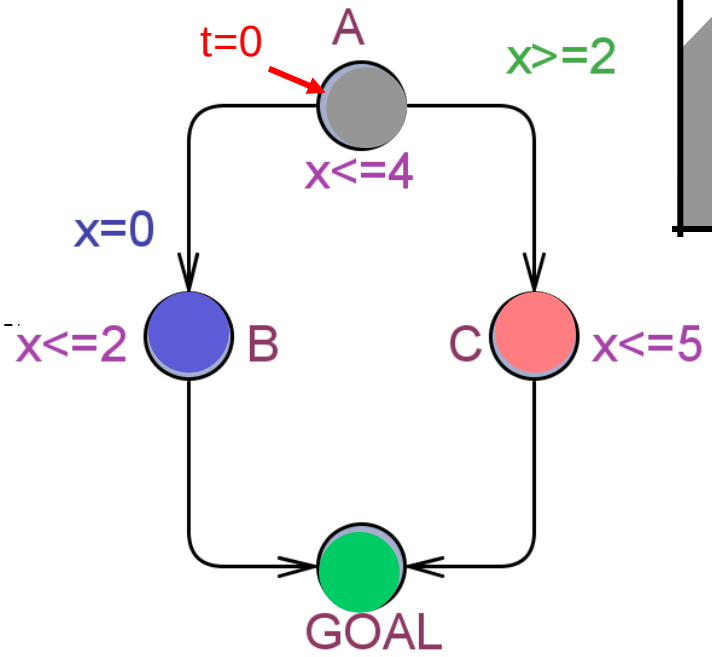
W=AF($t\leq 10 \wedge$ GOAL)

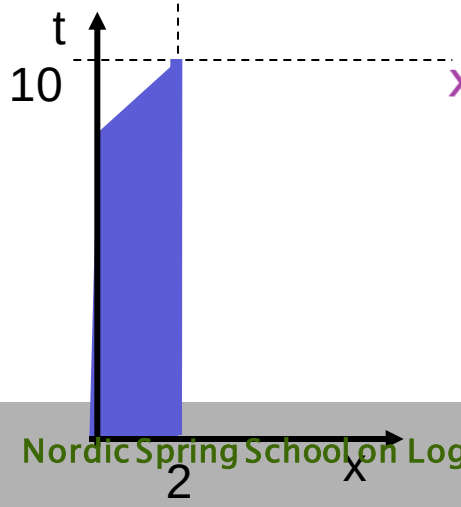
W = ($t\leq 10 \wedge$ GOAL) $\vee$ (AX W)

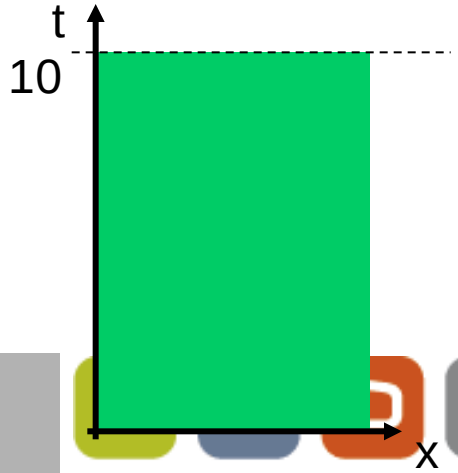












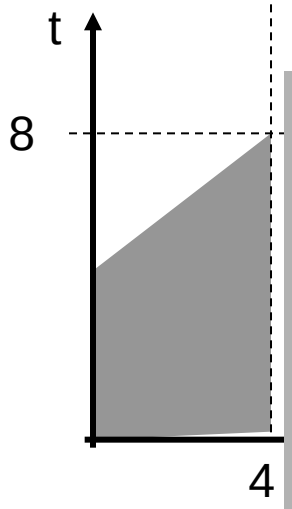
TCTL Model Checking

Fixpoint Computation

$AF_{t\leq 10}$ GOAL

$W = AF(\ t \leq 10 \wedge \text{GOAL}\ )$

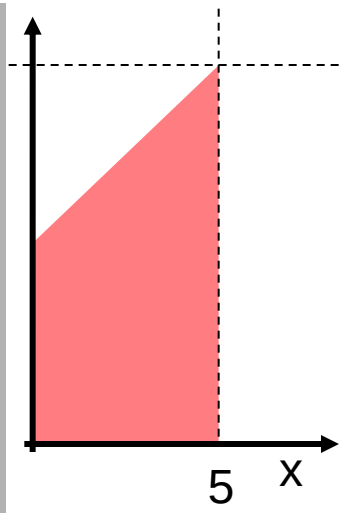
$W = (\ t \leq 10 \wedge \text{GOAL}\ ) \vee (AX\ W \wedge \forall\ W)$

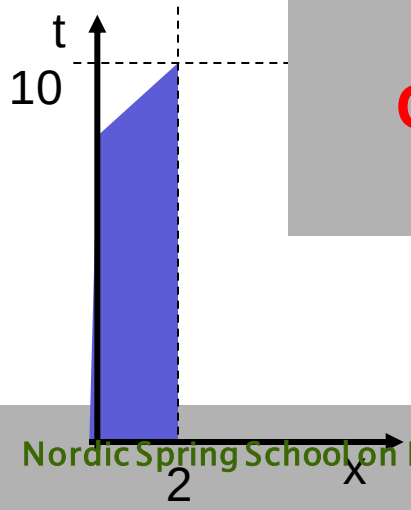


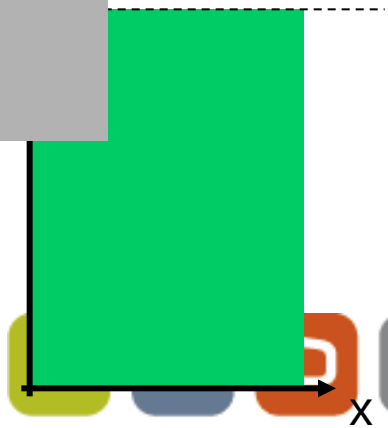
ALUR & DILL'94:
TCTL model checking for TA is decidable and **PSPACE-complete.**

Symbolic representation of state-sets using difference-constraints on clocks.

On-the-fly forward algorithms yield efficiency in practice

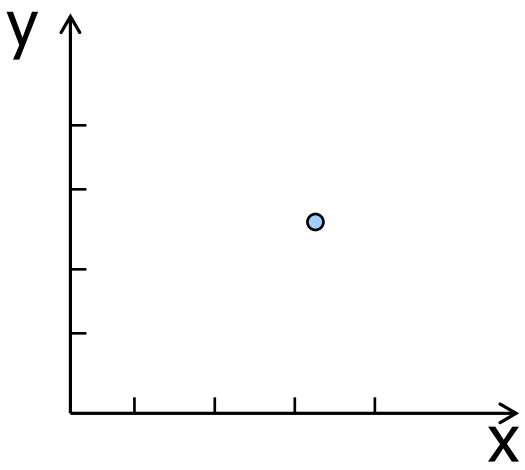






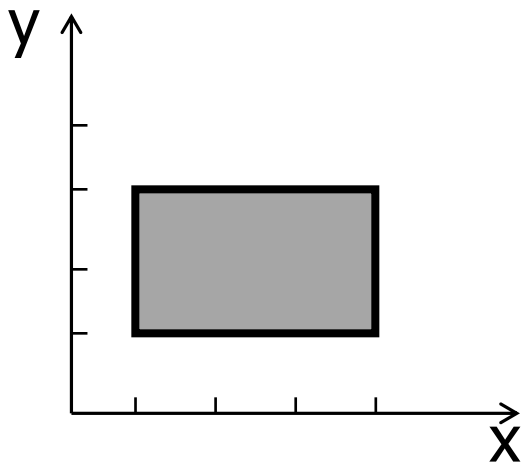
Zones – From infinite to finite

State
(n, x=3.2, y=2.5)

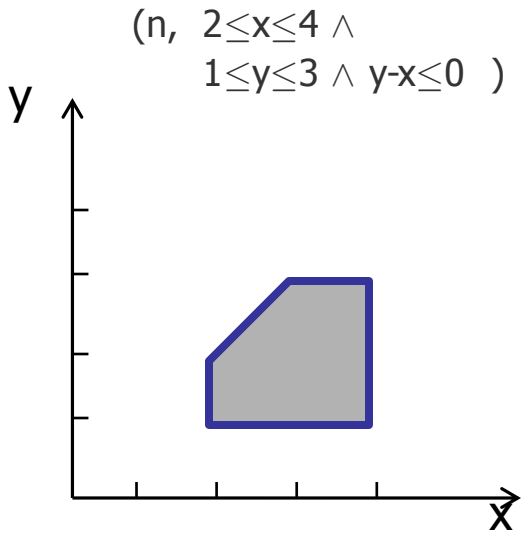


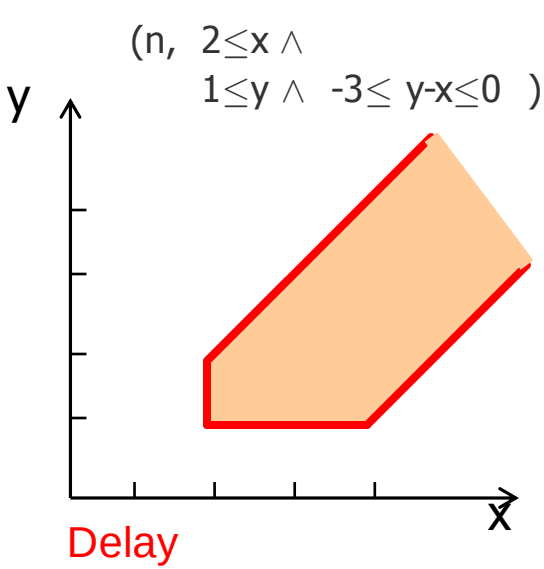
Symbolic state (set)
(n, $1 \leq x \leq 4$, $1 \leq y \leq 3$)

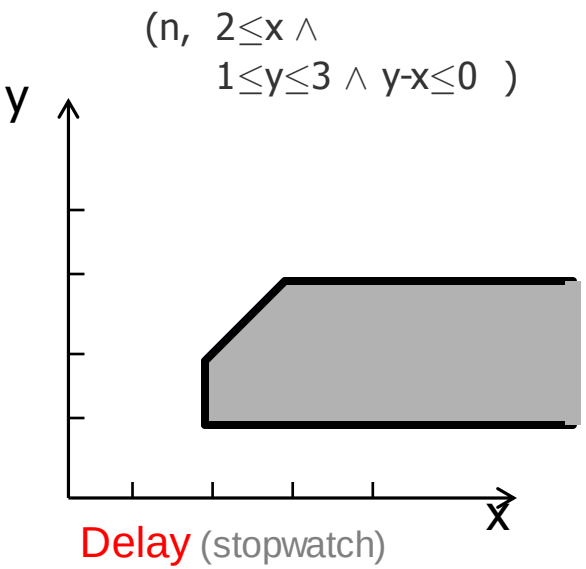
Zone:
conjunction of
x-y<=n, x<=>n

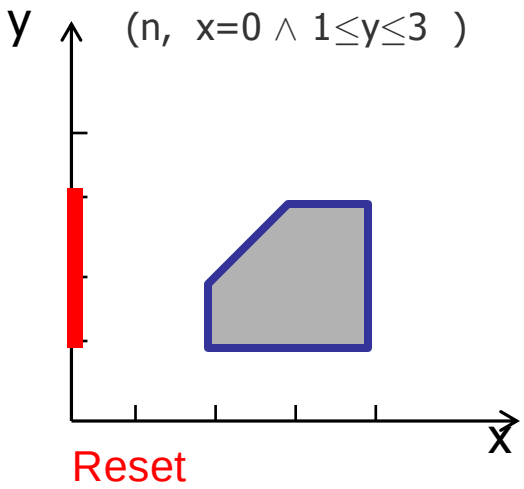


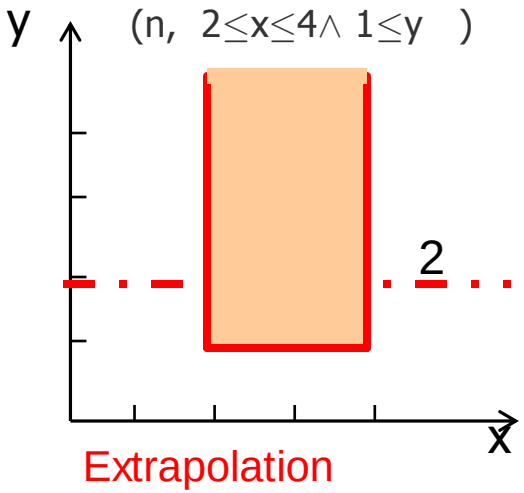
Zones – Operations

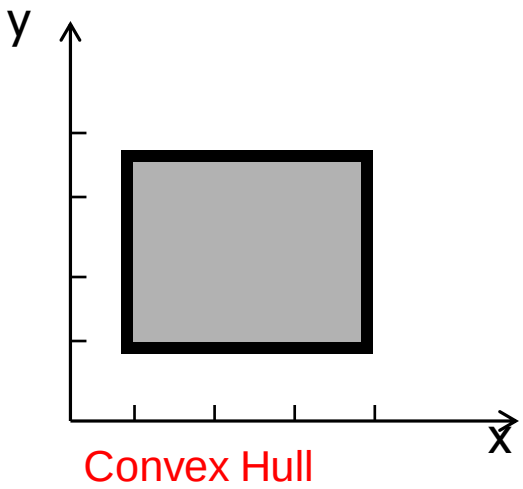






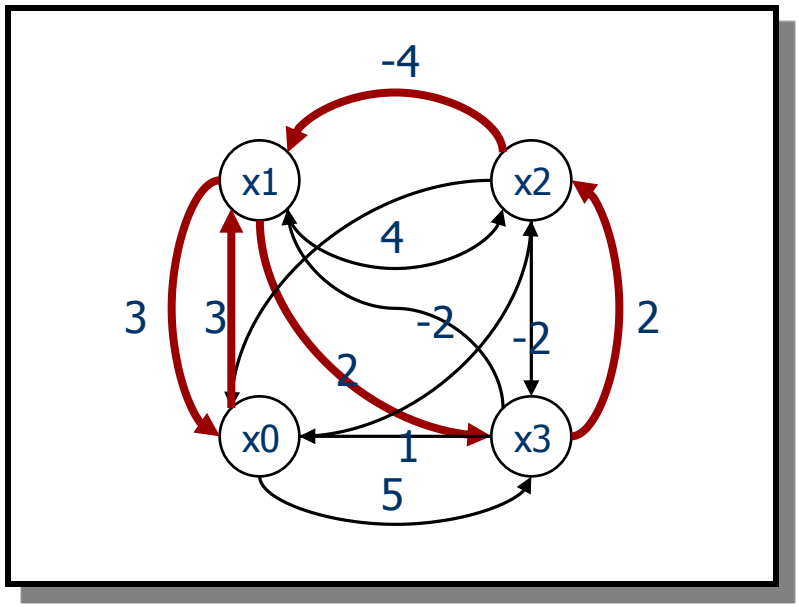


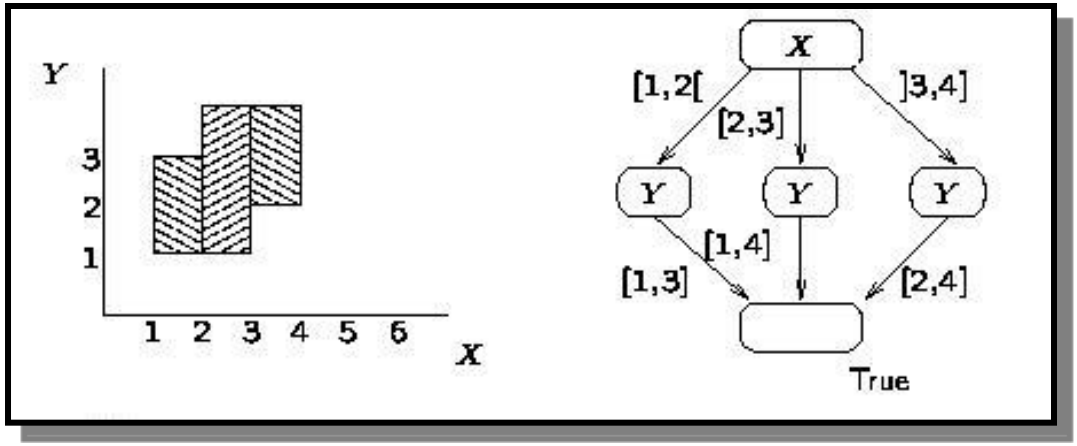




Datastructures for Zones

- Difference Bounded Matrices (DBMs)
- Minimal Constraint Form
  [RTSS97]
- Clock Difference Diagrams
  [CAV99]
- PW List
  [SPIN03]





Train Crossing