Decidability of Reachability Problems for Classes of Two Counters Automata

Alain Finkel, Grégoire Sutre

LSV, ENS Cachan & CNRS UMR 8643, France.

{finkel, sutre}@lsv.ens-cachan.fr

Plan

1 – Introduction

2 – Extended VASS

3 – Symbolic Reachability Tree algorithm

4 – Conclusion

1– Introduction

Verification of infinite state systems

- Infinite state systems:

  $\Longrightarrow$ finite control

  $\Longrightarrow$ variables ranging over infinite data domains

- Model-checking safety properties:

$$\begin{array}{lcll} \mathit{Bad} & \cap & \mathrm{post}^*(\mathit{Init}) & \stackrel{?}{=} \emptyset \\ \mathit{Init} & \cap & \mathrm{pre}^*(\mathit{Bad}) & \stackrel{?}{=} \emptyset \end{array}$$

$\Longrightarrow$ exact computation of $\mathrm{post}^*/\mathrm{pre}^*$

Computation of $\mathrm{post}^*/\mathrm{pre}^*$

- finitely describable infinite subsets of data values

- good properties:

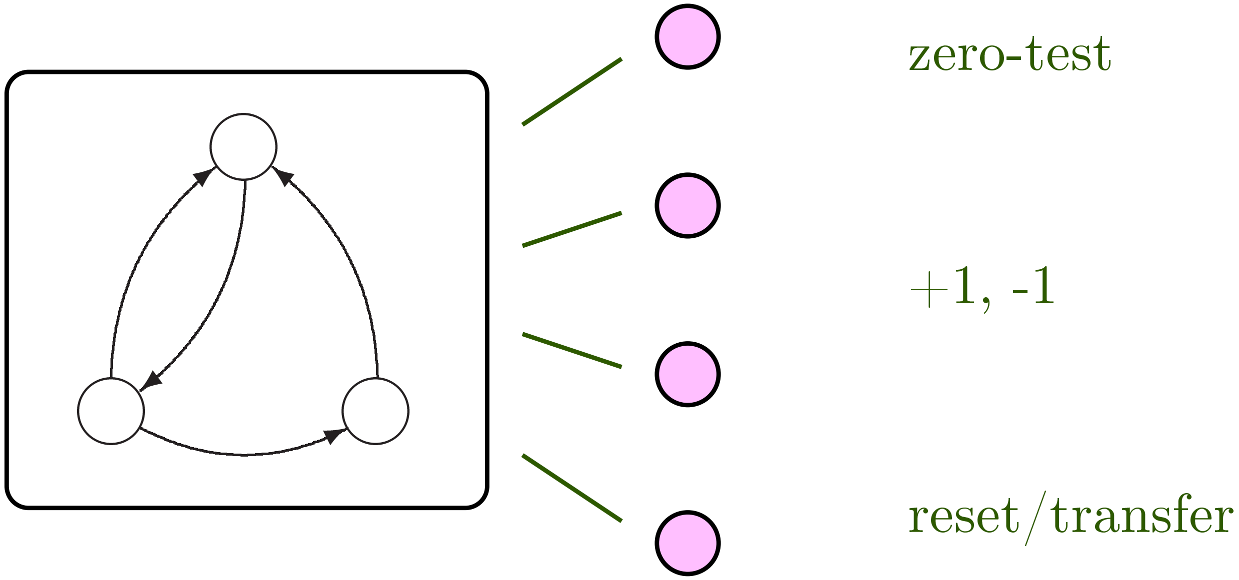
  $\Longrightarrow$ effectively closed under post/pre, $\cup$ and $\cap$

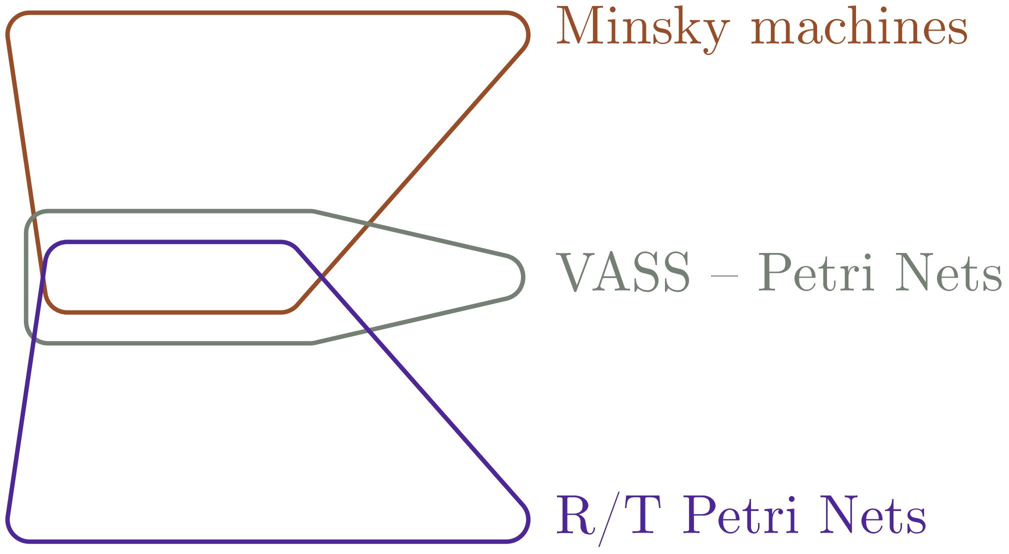
  $\Longrightarrow$ membership and inclusion are decidable

- acceleration techniques

Extended VASS

- variables — counters — range over $\mathbb{N}$





post* and pre* are not in general recursive for Extended VASS

Semilinear sets

$$\bigcup_{i \in I} (x_i + P_i{}^*) \quad \text{where} \quad \begin{cases} x_i \in \mathbb{N}^n \\ P_i = \{p_1, p_2, \cdots, p_k\} \subseteq \mathbb{N}^n \\ P_i{}^* = \{\Sigma_{i=1}^k \alpha_i p_i \ / \ \alpha_i \in \mathbb{N}\} \end{cases}$$

- finitely describable infinite subsets of $\mathbb{N}^n$

- good properties:

  $\Longrightarrow$ effectively closed under post/pre, $\cup$ and $\cap$

  $\Longrightarrow$ membership and inclusion are decidable

Our goal

- Find classes of Extended VASS for which post*/pre* is effectively semilinear

- Develop semi-algorithms computing post*/pre* for general Extended VASS

Related work

- Reset/Transfer Petri Nets [DFS98 (ICALP'98), DJS99 (ICALP'99)]

  $\Longrightarrow$ Reachability U for 3-dim E-VASS with $reset(1) + reset(2)$

  $\Longrightarrow$ Boundedness U for 3-dim E-VASS with resets

  $\Longrightarrow$ Boundedness D for n-dim E-VASS with $reset(1) + reset(2)$

- 2-dim VASS [HP79 (TCS)]

  $\Longrightarrow$ post* effectively semilinear for 2-dim VASS

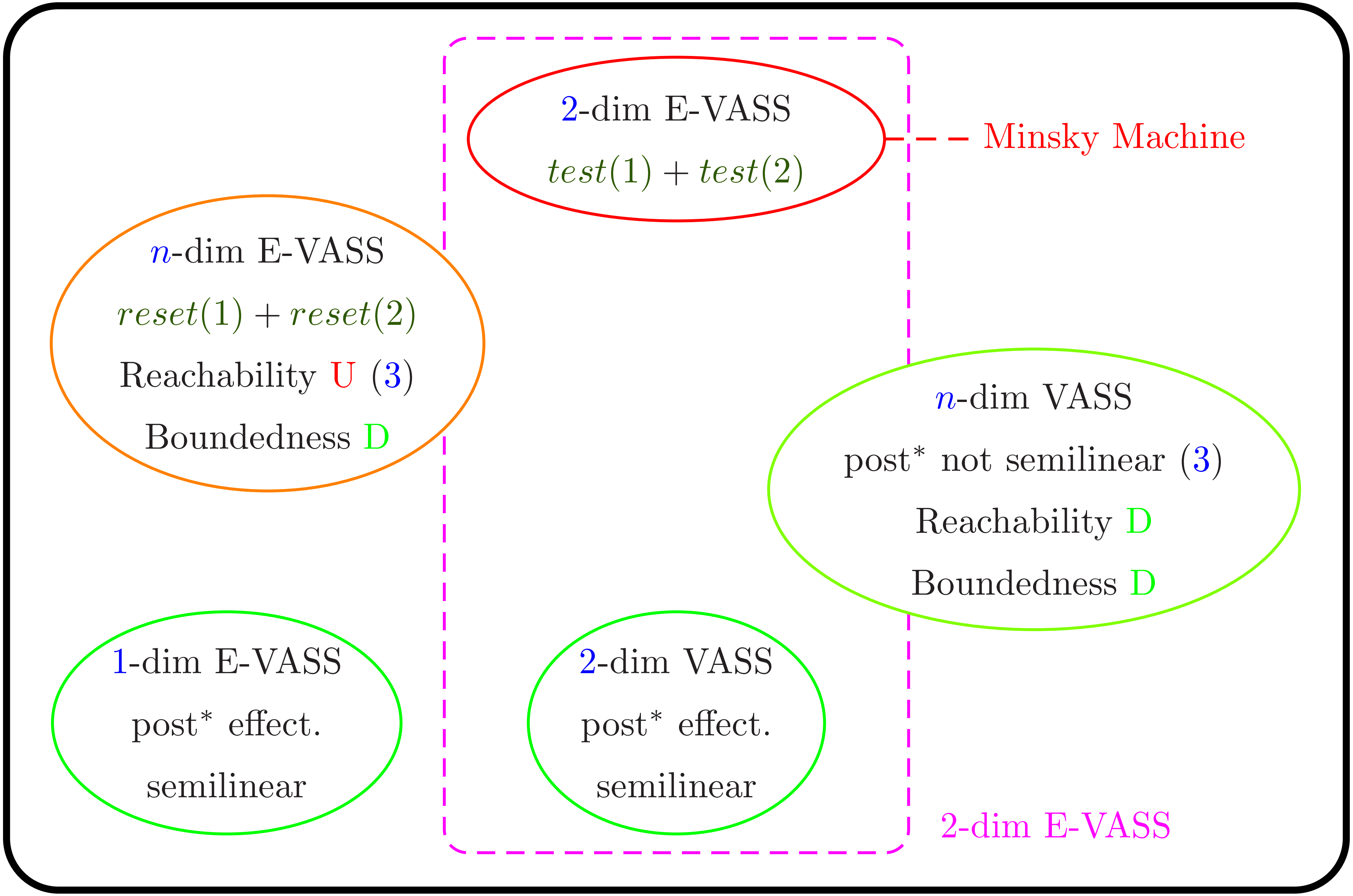
  $\Longrightarrow$ post* not semilinear for 3-dim VASS

- Lossy VASS [BM99 (STACS'99)]

  $\Longrightarrow$ pre* effectively semilinear for Lossy E-VASS with zero-tests

  $\Longrightarrow$ post* effectively semilinear for Lossy E-VASS

  $\Longrightarrow$ post* not constructible for Lossy E-VASS with zero-tests


2-dim E-VASS
$test(1) + test(2)$
Minsky Machine
n-dim E-VASS
$reset(1) + reset(2)$
Reachability U (3)
Boundedness D
n-dim VASS
post* not semilinear (3)
Reachability D
Boundedness D
1-dim E-VASS
post* effect.
semilinear
2-dim VASS
post* effect.
semilinear
2-dim E-VASS

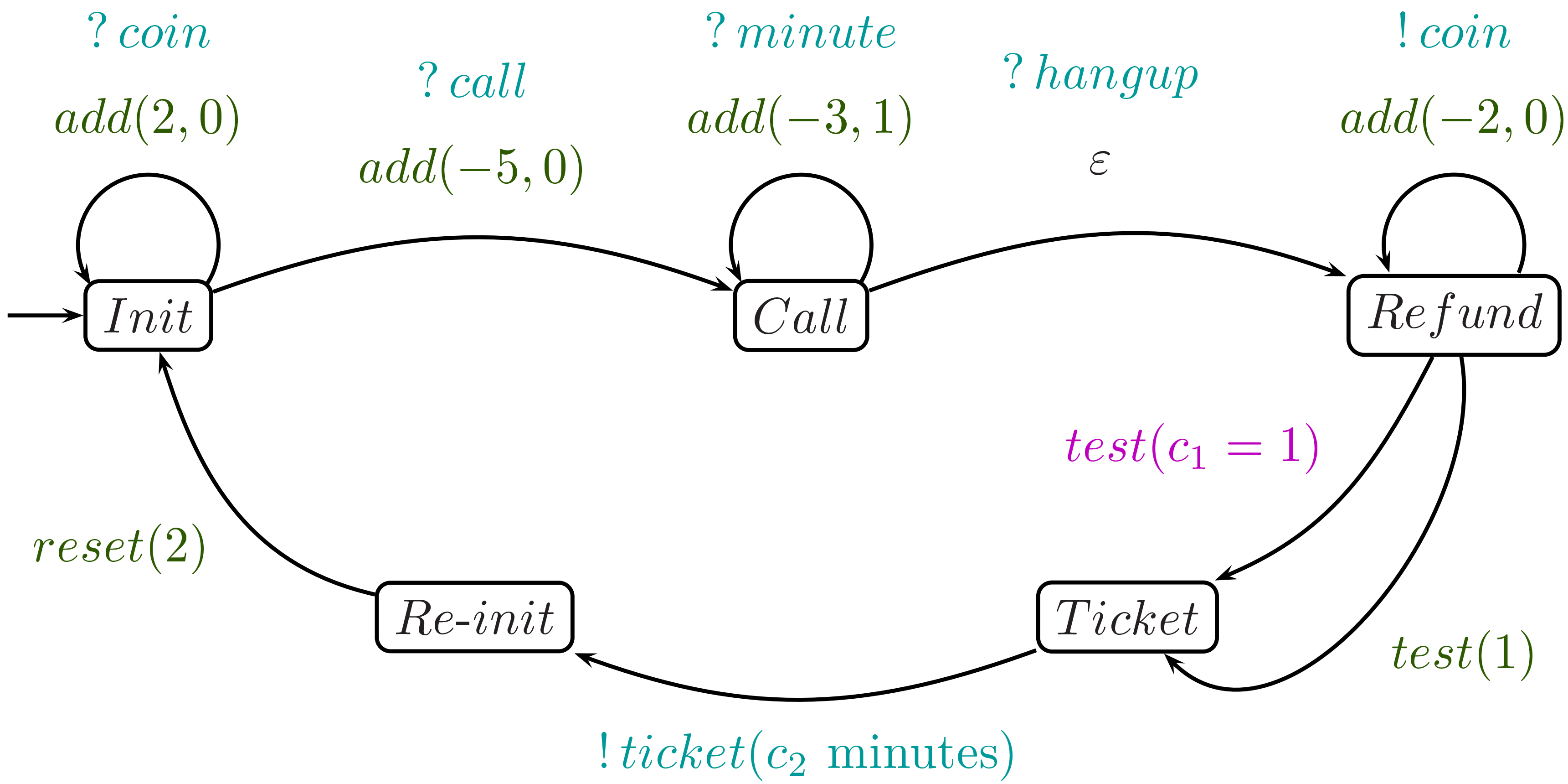
Our goal (revisited)

- Find classes of 2-dim Extended VASS for which post*/pre* is effectively semilinear

- Develop semi-algorithms computing post*/pre* for general Extended VASS

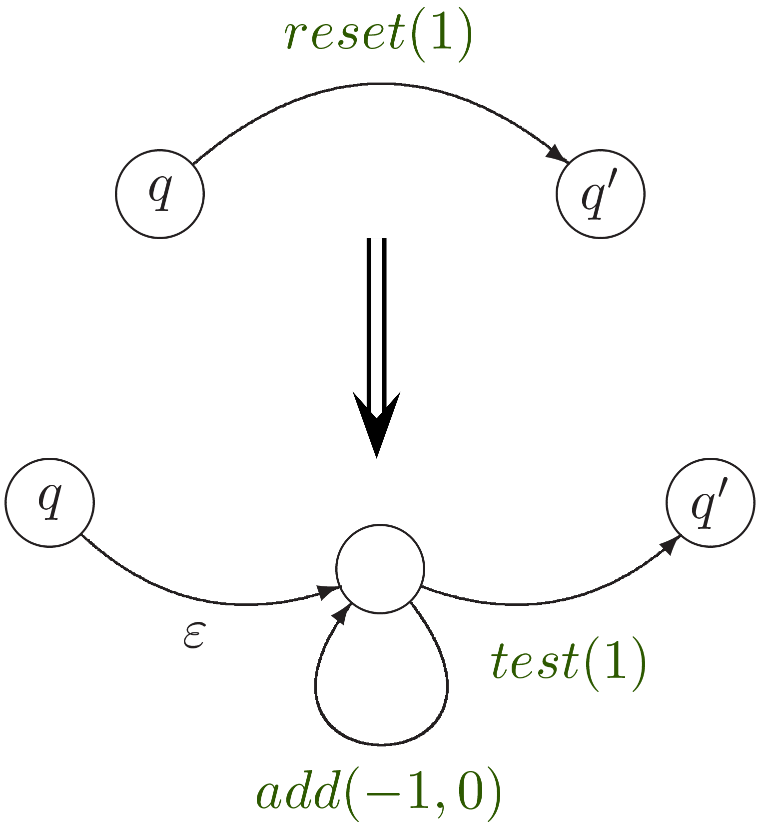
2– Extended VASS

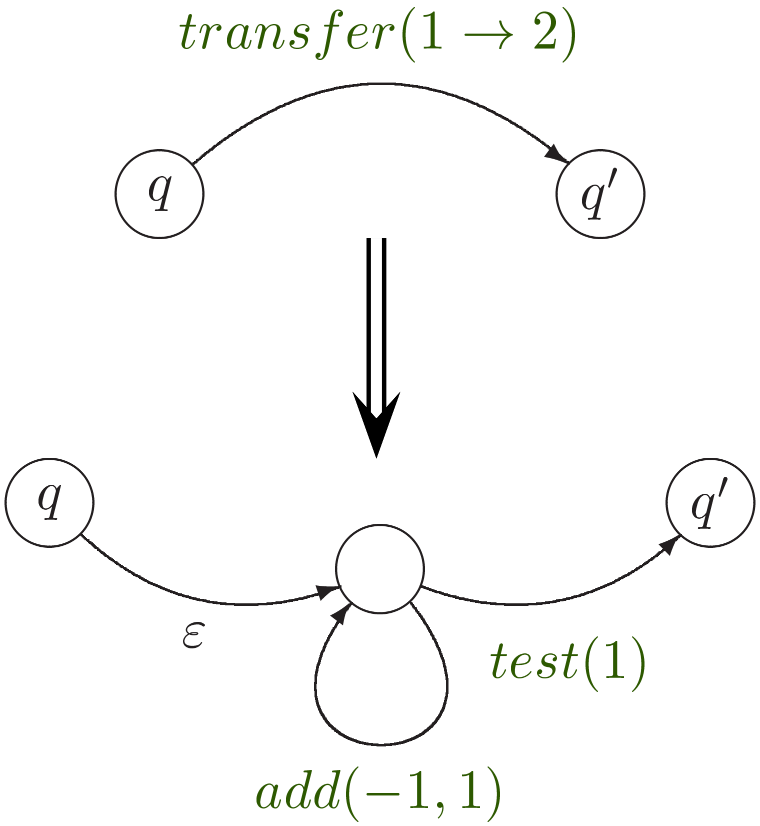
Extended VASS example: a phone box



c_1 : coins $\qquad$ c_2 : minutes

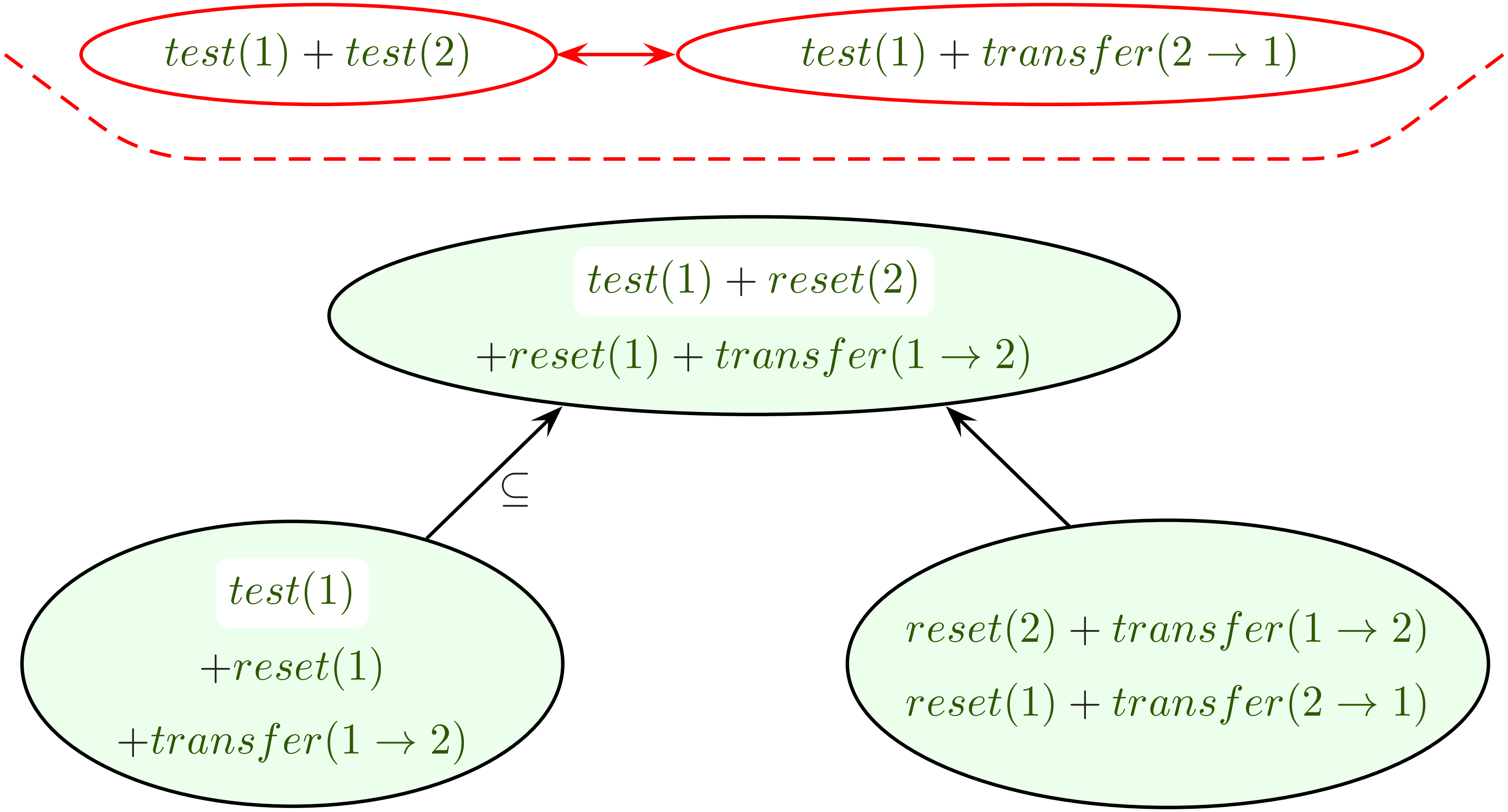
Simulations





these simulations preserve post* and pre*

Hierarchy of 2-dim E-VASS



3– Symbolic Reachability Tree algorithm

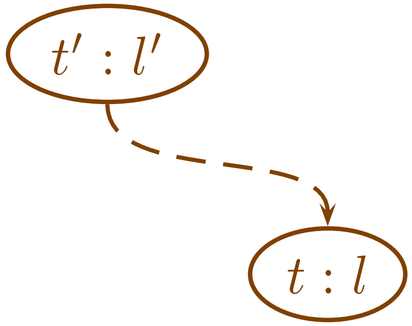
`SymbolicTree` algorithm

- semi-algorithm for general Extended VASS

- nodes labelled by 3-tuples (q, x, P), with $[\![q, x, P]\!] = \{q\} \times (x + P^*)$

- use of an `Acceleration` function

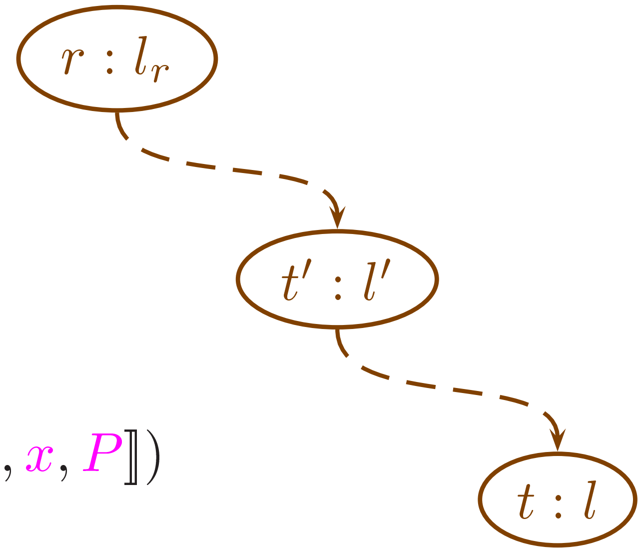
- stopping criteria:



with $[\![l]\!] \subseteq [\![l']\!]$

$\Longrightarrow$ terminates for some classes of 2-dim Extended VASS

Acceleration function



$\texttt{Acceleration} : Nodes \to 2^{\mathbb{N}^2}$

$W = \texttt{Acceleration}(t : (q, x, P))$

$(stability) \qquad [\![q, x, P]\!] \subseteq [\![q, x, W]\!] \subseteq post^*([\![q, x, P]\!])$

- stability ensures correctness of the `SymbolicTree` algorithm:

$$\mathrm{post}^*([\![l_{init}]\!]) = \bigcup_{t:l \in Nodes} [\![l]\!]$$

Goal: given a class, find a stable `Acceleration` function ensuring termination

`Acceleration` function for 2-dim VASS

- inspired from Hopcroft-Pansiot's algorithm: $\texttt{Acceleration}_{HP}$

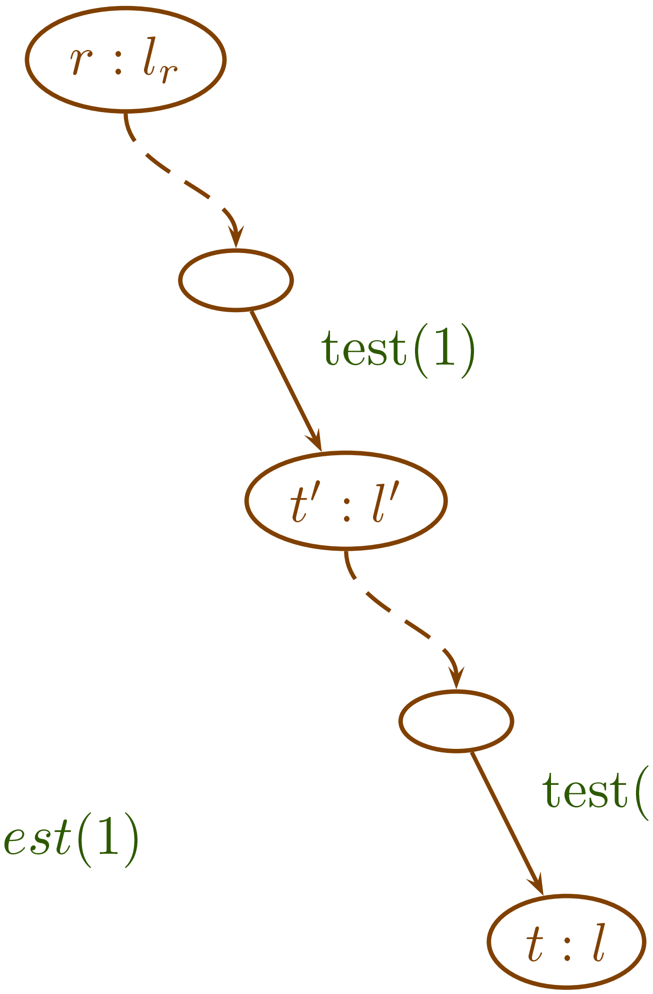
$\Longrightarrow$ stable

$\Longrightarrow$ ensures termination for 2-dim VASS

$\Longrightarrow$ $post^*/pre^*$ is effectively semilinear for 2-dim VASS [HP79]

Acceleration function for $test(1)$

$l = (q, (0, a), P)$ and $l' = (q', (0, a'), P')$



> **if** $(q = q')$ and $(P' \subseteq P)$ and $(a > a')$ **then**
> **return** $P \cup \{(0, a - a')\}$

$\Longrightarrow$ stable

$\Longrightarrow$ ensures termination for 2-dim E-VASS with $test(1)$

$\Longrightarrow$ post*/pre* is effectively semilinear

`Acceleration` **function for** $test(1) + reset(2)$ — pre*

- $weaktest$ transitions: $(q, x) \xrightarrow{weaktest(i)} (q', x')$ iff $(q', x') \xrightarrow{reset(i)} (q, x)$

- design of a stable `Acceleration` function ensuring termination for 2-dim E-VASS with $test(1) + weaktest(2)$

$\Longrightarrow$ post* is effectively semilinear for 2-dim E-VASS with $test(1) + weaktest(2)$

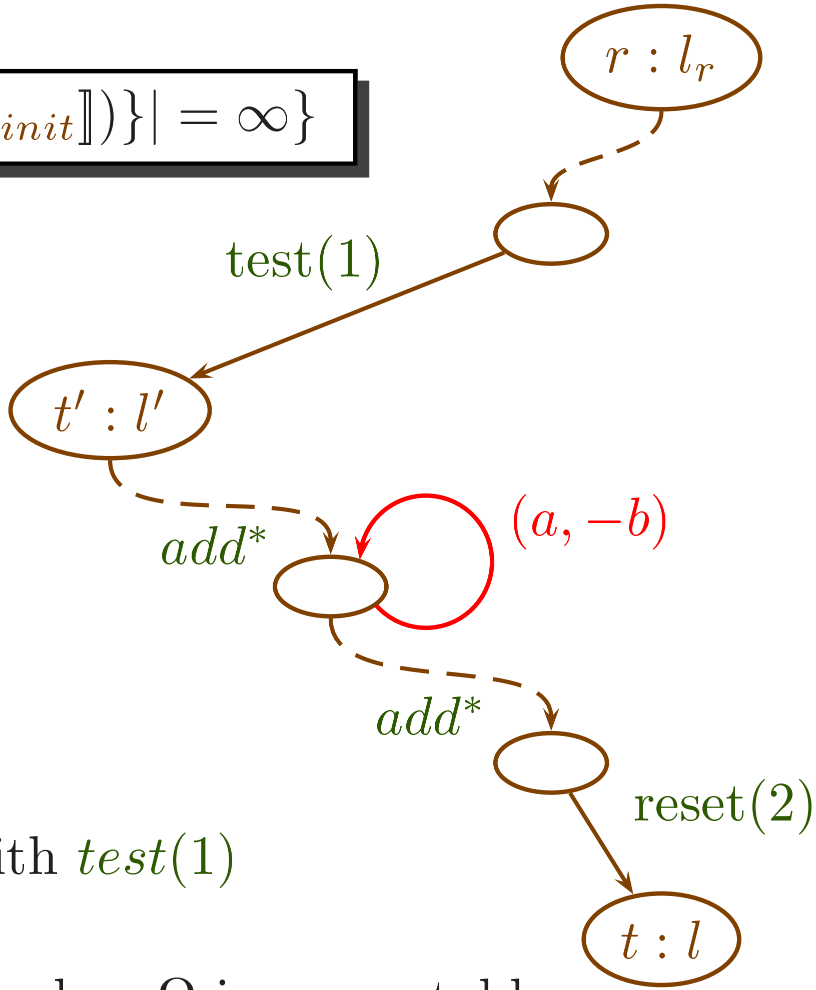
$\Longrightarrow$ pre* is effectively semilinear for 2-dim E-VASS with $test(1) + reset(2)$

Acceleration function for $test(1) + reset(2)$ — post*

$$\Omega = \{q \in Q \ / \ |\{x \in \mathbb{N} \ / \ (q, (0, x)) \in \text{post}^*([\![l_{init}]\!])\}| = \infty\}$$

$l' = (q', (0, a'), P')$

if $(q' \in \Omega)$ and $a, b > 0$ **then**
return $P \cup \{(a, 0)\}$



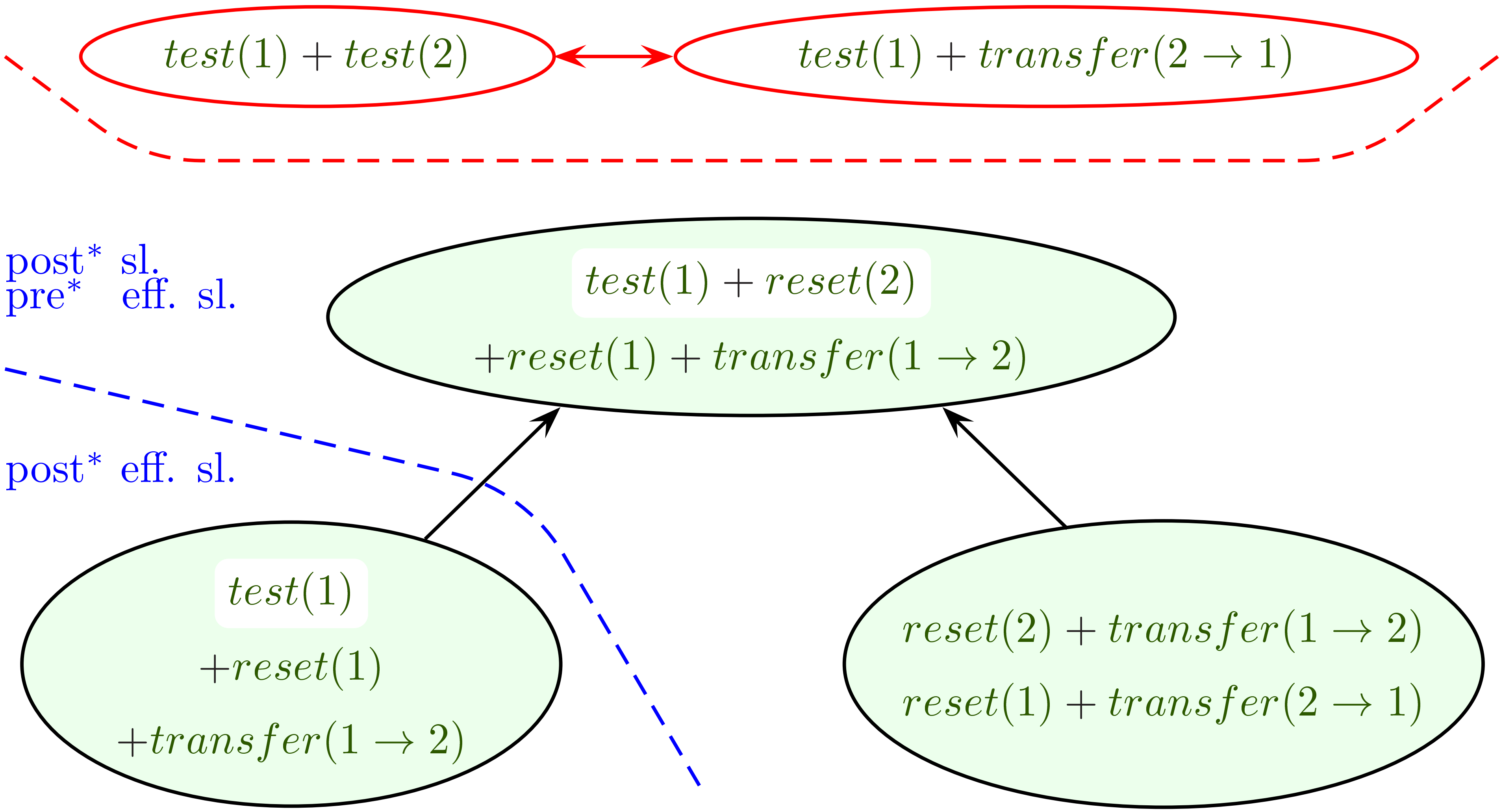
$\Longrightarrow$ stable

$\Longrightarrow$ ensures termination for 2-dim E-VASS with $test(1)$

$\Longrightarrow$ post* is semilinear and it is constructible when Ω is computable

4– Conclusion

Computation of post*/pre* for 2-dim E-VASS



Applications

- computation of pre^* : model-checking with the {EF, EX} fragment of CTL

- semi-algorithm computing $\text{post}^*/\text{pre}^*$ for n-dim E-VASS

  $\Longrightarrow$ give an under-approximation when stopped

- over-approximation of $\text{post}^*/\text{pre}^*$ for n-dim E-VASS

  $\Longrightarrow$ 2 counters with zero-tests: replace 1 zero-test by a reset or ignore it

  $\Longrightarrow$ keep only 2 counters, compute $\text{post}^*/\text{pre}^*$ (over-approximated) and then intersect the results for each pair

Open problems

- computation of post* for the class of 2-dim E-VASS with $test(1) + reset(2)$

- model-checking with LTL (on infinite traces)