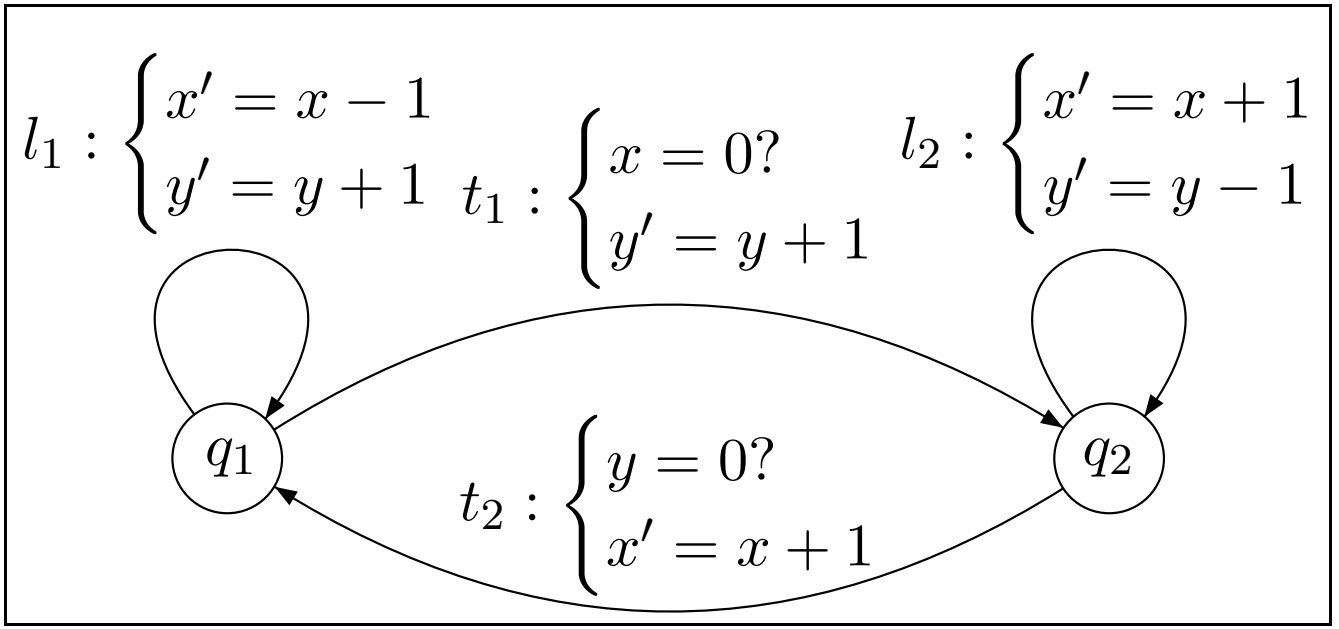
Flat counter automata almost everywhere !

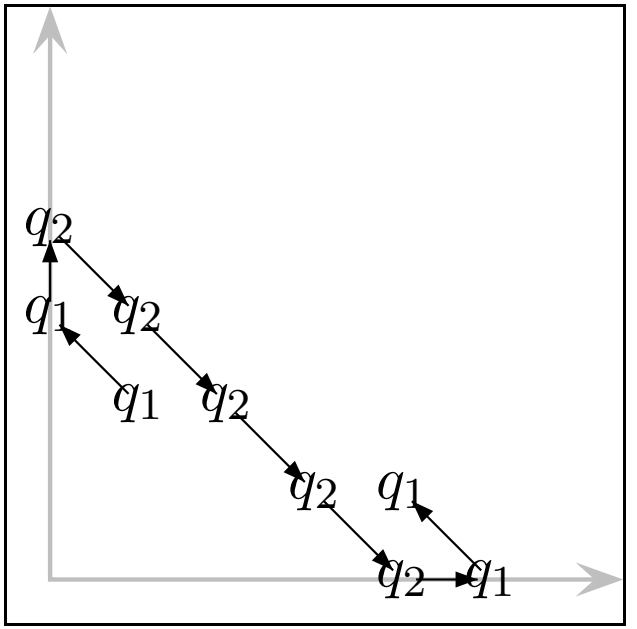
Jérôme Leroux and Grégoire Sutre

Projet Vertecs, IRISA / INRIA Rennes, FRANCE
Équipe MVTsi, CNRS / LABRI, FRANCE

Counter-automata verification

A simple counter-automata:

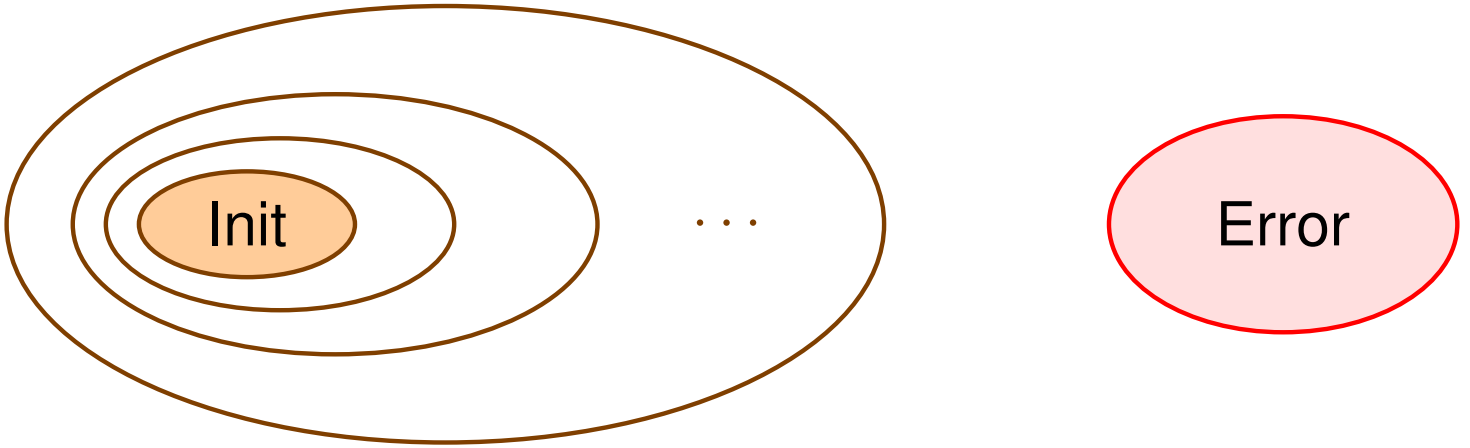




Counter-automata verification naturally appears in practice:
$\longrightarrow$ Parametrized systems, system abstractions, communication protocols, and so on.

Counter-automata reachability

Verification can often be reduced to the reachability problem.



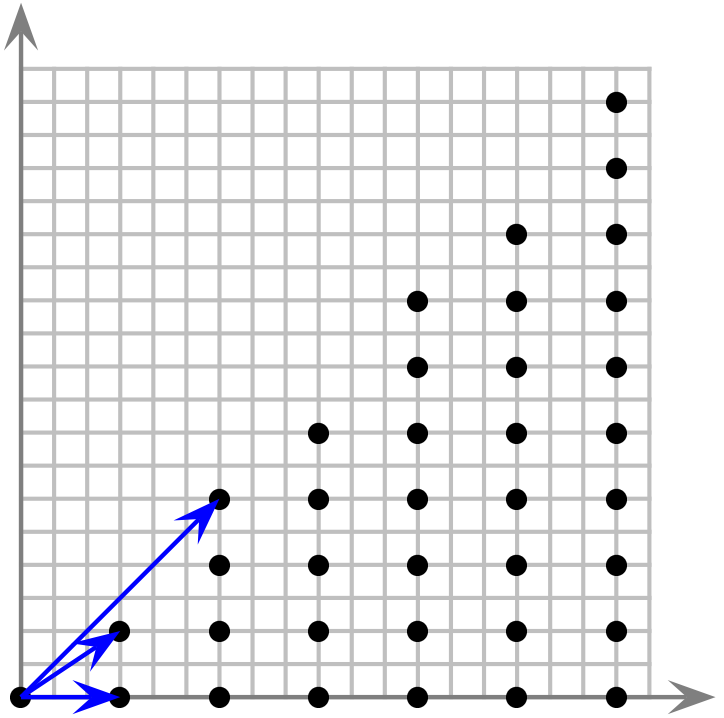
An agorithm in general ?
NO ! Because reachability is undecidable even for 2-counter automata.

However, there exist algorithms for subclasses of counter-automata.

Some of these algorithm use semilinear sets to symbolically represent and manipulate infinite subsets of $\mathbb{Z}^n$.

Semilinear sets

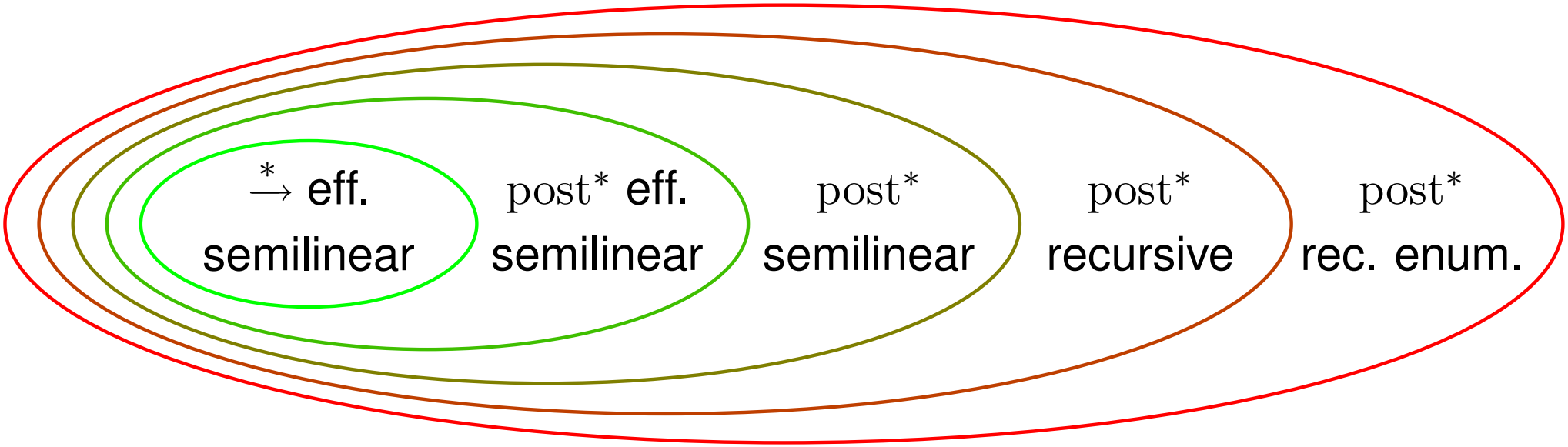
A semilinear set $X \subseteq \mathbb{Z}^m$ is a finite union of linear sets $b + \{p_1, \ldots, p_n\}^*$.



$$\{(0,0)\} + \{(3,0),(3,2),(6,6)\}^*$$

Recall that semilinear sets can be manipulated with:
$\longrightarrow$ Finite sets of basis and periods, Presburger formulas, digit vector automata.

Subclasses of counter automata



Reversible Petri nets
Conflict-free Petri nets
BPP nets

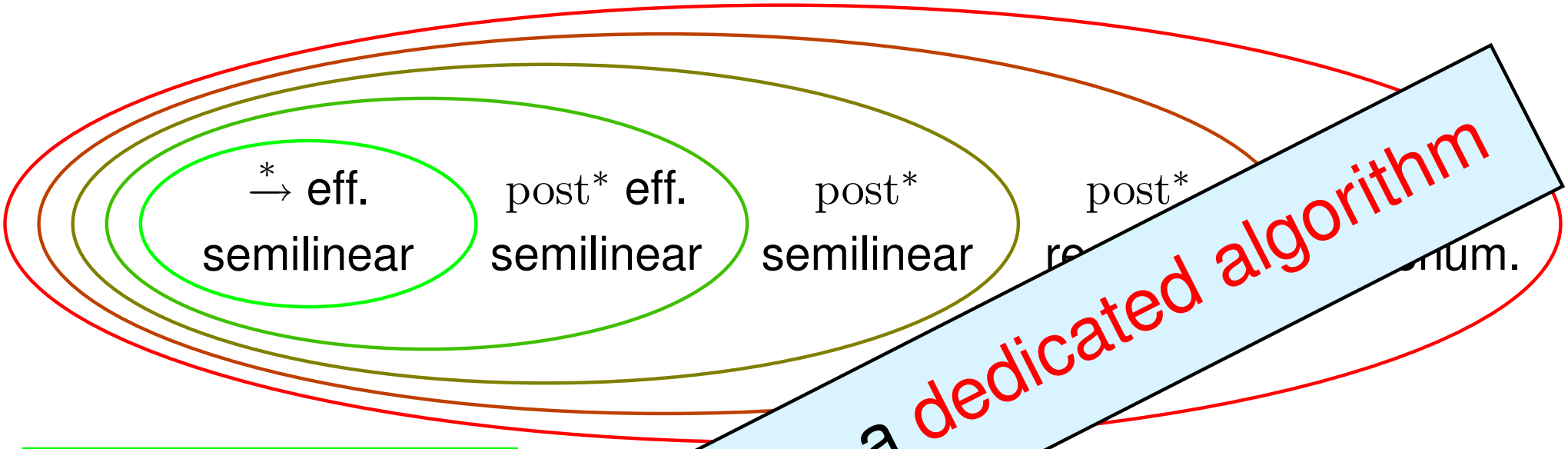
Cyclic Petri nets
Persistant Petri nets
Regular Petri nets
Test-free 2-counter machines, ...
Lossy test-free counter machines

Lossy counter machines

Petri nets

2-counter machines

Subclasses of counter automata



Reversible Petri nets
Conflict-free Petri nets
BPP nets

machines

tri nets
rsistant Petri nets
Regular Petri nets
Test-free 2-counter machines, . . .
Lossy test-free counter machines

Each (decidable) class has a dedicated algorithm

Petri nets

2-counter machines

A generic accelerated algorithm

In practice counter automata are not exactly in a known subclass.
$\longrightarrow$ we are interested in semi-algorithms for general classes.

Input: A counter automaton $\mathcal{S}$.

Output: The global reachability relation $\xrightarrow{*}$.

`let` $R \leftarrow Id$ and `repeat forever`

`select` one of the following tasks:

- `if` $\xrightarrow{T} \cdot R \subseteq R$ `return` R
- `select` $\pi \in T^*$ and $R', R'' \subseteq R$
  `let` $R \leftarrow R \cup (R' \cdot \xrightarrow{\pi^*} \cdot R'')$
- `select` $t \in T$ and $R', R'' \subseteq R$
  `let` $R \leftarrow R \cup (R' \cdot \xrightarrow{t} \cdot R'')$

Input: An initialized counter automaton $(\mathcal{S}, I)$.

Output: The reachability set $\mathrm{post}^*(I)$.

`let` $X \leftarrow I$ and `repeat forever`

`select` one of the following tasks:

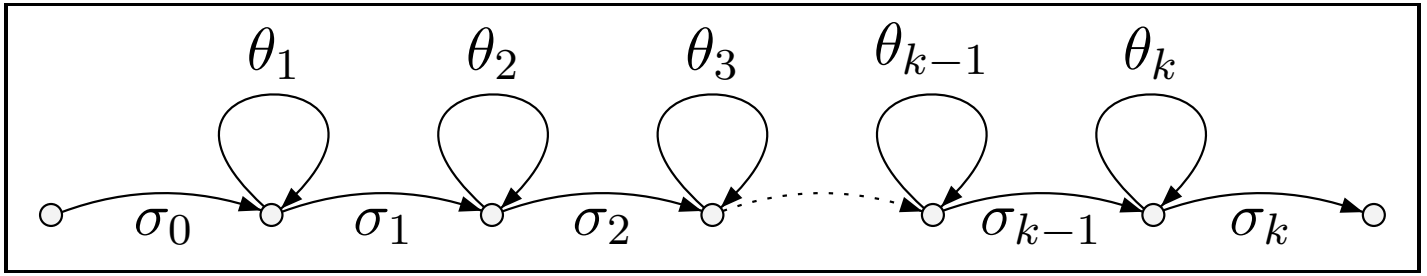
- `if` $\mathrm{post}(T, X) \subseteq X$ `return` X
- `select` $\pi \in T^*$ and $X' \subseteq X$
  `let` $X \leftarrow X \cup \mathrm{post}(\pi^*, X')$
- `select` $t \in T$ and $X' \subseteq X$
  `let` $X \leftarrow X \cup \mathrm{post}(t, X')$

Implemented in tools: FAST, LASH, TReX.
$\longrightarrow$ Accelerated symbolic verification works well in practice.

Flatness

$\mathcal{S}$ is flat if $\mathcal{S}$ is equivalent (w.r.t. reachability) to $\mathcal{S}'$ where:

- $\mathcal{S}'$ is "extracted" from $\mathcal{S}$ (some loops of $\mathcal{S}$ may be unrolled)
- $\mathcal{S}'$ contains no nested loops



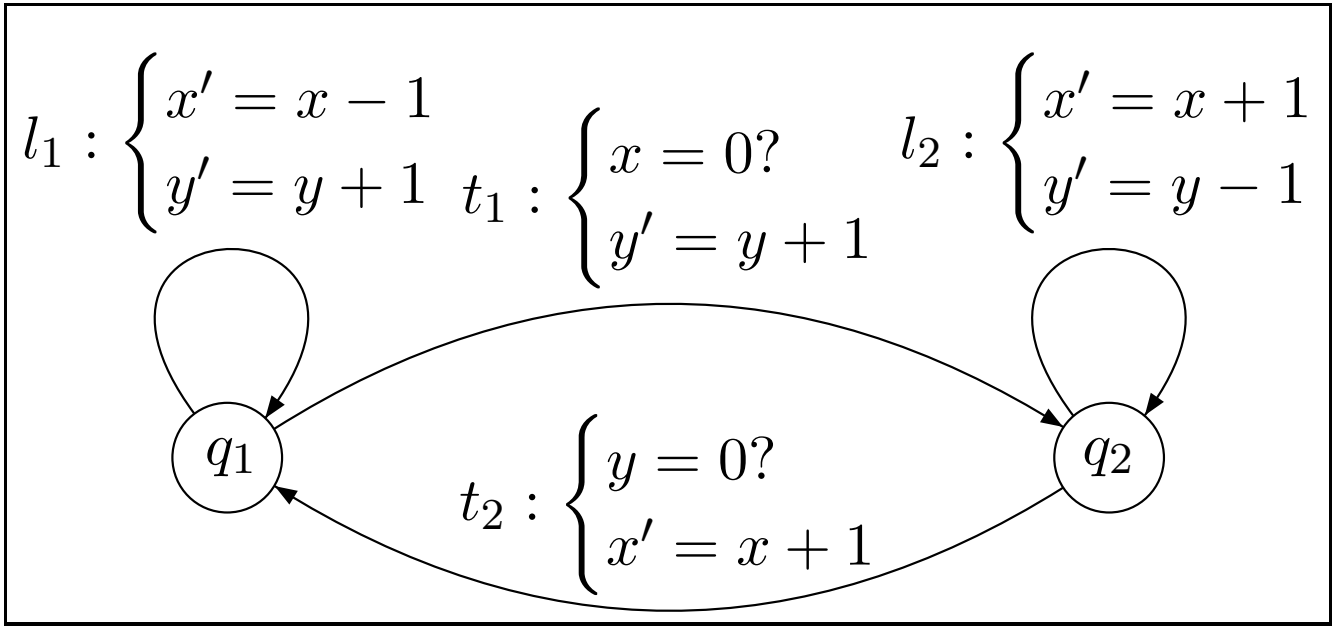
Accelerated $\xrightarrow{*}/\mathrm{post}^*$ computation terminates iff $\mathcal{S}$ is flat.

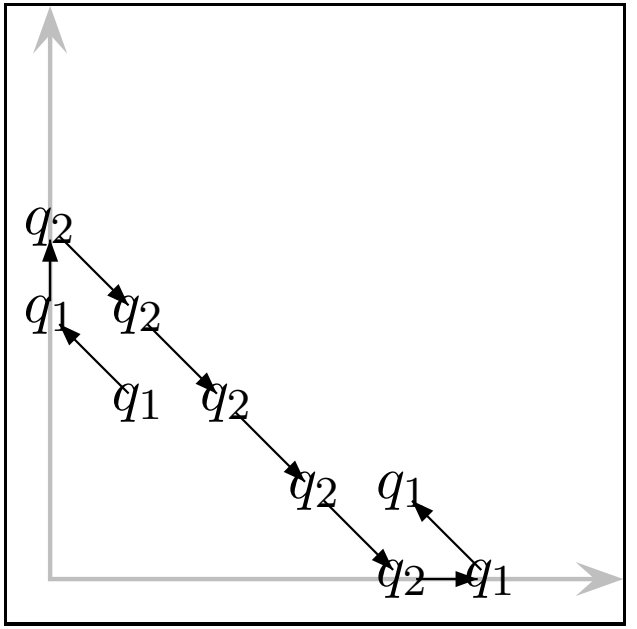
Hence $\xrightarrow{*}/\mathrm{post}^*$ is effectively semilinear for flat counter automata where:

- all loops can be effectively accelerated
- other natural effectivity conditions hold

Improving the acceleration algorithm

Flatness implies effective semilinearity of post^*. The converse is not true:





To obtain $\mathrm{post}^*(1,0)$, we "need" the path: $(l_1)^1 t_1 (l_2)^2 t_2 (l_1)^3 t_1 (l_2)^4 \cdots$

Is-it possible to improve acceleration techniques with parts of the known dedicated algorithms for semilinear subclasses ?
$\longrightarrow$ characterize the effective semilinear subclasses that are not flat ?

Outline

Flat counter automata almost everywhere !

- Introduction.

$\Rightarrow$ Counter machines and acceleration.

- Flat counter machines.
  - Reversal bounded counter machines.
  - Lossy/inserting counter machines.
  - Test-free 2-dim counter machines.
- Flat Petri nets.
  - Cyclic and reversible Petri nets.
  - Regular Petri nets.
  - BPP-nets.
- Conclusion.

Counter machines

An n-dim counter machine $\mathcal{S}$ is a tuple $\mathcal{S} = (Q, T, (\xrightarrow{t})_{t \in T}, \alpha, \beta, \#, \mu, \delta)$:

- Q is a non-empty finite set of locations.
- T is a set of transitions.
- Relation $\xrightarrow{t}$ is defined over the set of configurations $Q \times \mathbb{N}^n$ by $(q, x) \xrightarrow{t} (q', x')$ if and only if $q = \alpha(t)$, $q' = \beta(t)$, $x \,\#(t)\, \mu(t)$ and $x' = x + \delta(t)$, where:
  - $\alpha, \beta : T \to Q$ are the source and target mappings,
  - $\# : T \to \{=, \geq\}^n$, and
  - $\mu : T \to \mathbb{N}^n$ and $\delta : T \to \mathbb{Z}^n$ are such that $\mu(t) + \delta(t) \geq 0$.

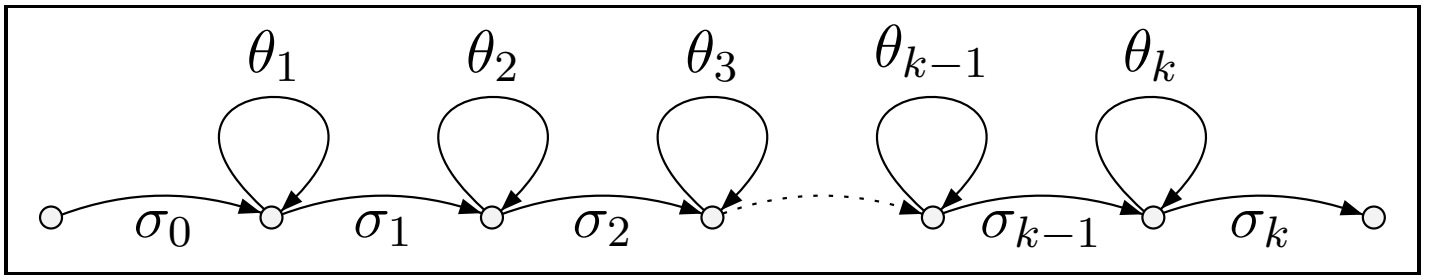
An initialized n-dim counter machine is a pair $(\mathcal{S}, I)$ where $\mathcal{S}$ is an n-dim counter machine and $I \subseteq Q \times \mathbb{N}^n$.

$\xrightarrow{\pi}$ and $\mathrm{post}(\pi, I)$ are naturally defined for any path $\pi \in T^*$.

Global reachability relation $\xrightarrow{*}$ is $\xrightarrow{T^*}$.
Reachability set $\mathrm{post}^*(I)$ is $\mathrm{post}(T^*, I)$.

Acceleration for counter machines

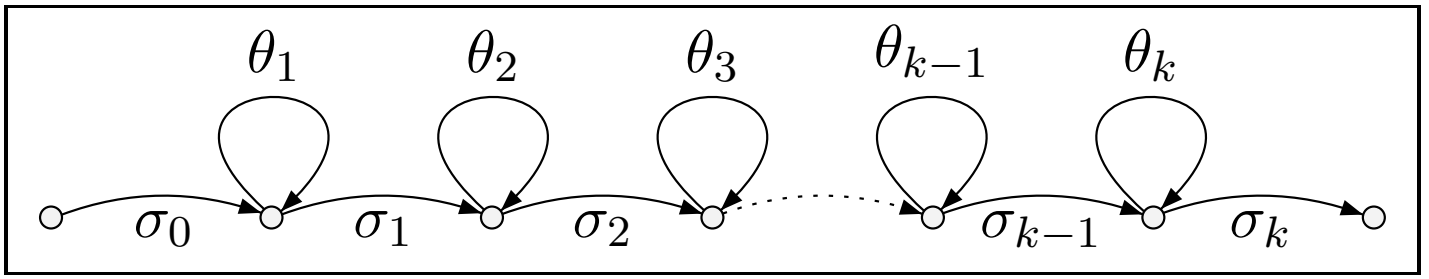
A semilinear path scheme $\rho \subseteq T^*$ is a finite union of linear path schemes $\sigma_0\theta_1^*\sigma_1 \cdots \theta_k^*\sigma_k$.



Thm[Finkel&Leroux'02, . . .]: For any SLPS ρ in a counter machine $\mathcal{S}$, the reachability subrelation $\xrightarrow{\rho}$ is effectively semilinear.

Flatness for counter machines (1/2)

A semilinear path scheme $\rho \subseteq T^*$ is a finite union of linear path schemes $\sigma_0\theta_1^*\sigma_1 \cdots \theta_k^*\sigma_k$.



A counter machine $\mathcal{S}$ is globally flat if $\xrightarrow{*} = \xrightarrow{\rho}$ for some SLPS ρ.
An initialized counter machine $(\mathcal{S}, I)$ is flat if $\mathrm{post}^*(I) = \mathrm{post}(\rho, I)$ for some SLPS ρ.

$\longrightarrow$ Global flatness implies flatness for any I. Converse is not true.

- $\xrightarrow{*}$ is effectively semilinear for any globally flat $\mathcal{S}$
- $\mathrm{post}^*(I)$ is effectively semilinear for any flat $(\mathcal{S}, I)$

Flatness for counter machines (2/2)

Input: A counter automaton $\mathcal{S}$.

Output: The global reachability relation $\xrightarrow{*}$.

`let` $R \leftarrow Id$ and `repeat forever`

`select` one of the following tasks:

- `if` $\xrightarrow{T} \cdot R \subseteq R$ `return` R
- `select` $\pi \in T^*$ and $R', R'' \subseteq R$
  `let` $R \leftarrow R \cup (R' \cdot \xrightarrow{\pi^*} \cdot R'')$
- `select` $t \in T$ and $R', R'' \subseteq R$
  `let` $R \leftarrow R \cup (R' \cdot \xrightarrow{t} \cdot R'')$

Input: An initialized counter automaton $(\mathcal{S}, I)$.

Output: The reachability set $\mathrm{post}^*(I)$.

`let` $X \leftarrow I$ and `repeat forever`

`select` one of the following tasks:

- `if` $\mathrm{post}(T, X) \subseteq X$ `return` X
- `select` $\pi \in T^*$ and $X' \subseteq X$
  `let` $X \leftarrow X \cup \mathrm{post}(\pi^*, X')$
- `select` $t \in T$ and $X' \subseteq X$
  `let` $X \leftarrow X \cup \mathrm{post}(t, X')$

Thm: These semi-algorithms are correct, and they admit a terminating execution iff the counter machine is (globally) flat.

$\longrightarrow$ The exploration strategy should be “fair” to ensure termination

Outline

Flat counter automata almost everywhere !

- Introduction.
- Counter machines and acceleration.

$\Rightarrow$ Flat counter machines.

  - Reversal bounded counter machines.
  - Lossy/inserting counter machines.
  - Test-free 2-dim counter machines.

- Flat Petri nets.
  - Cyclic and reversible Petri nets.
  - Regular Petri nets.
  - BPP-nets.
- Conclusion.

Reversal-bounded (1/2)

Recall: T set of transitions, $\delta : T \to \mathbb{Z}^n$ displacement labeling.
Let $\varphi_i^\delta : T^* \to \{+,-\}^*$ be the morphism defined by:

$$\varphi_i^\delta(t) = \begin{cases} + & \text{if } \delta(t)[i] > 0 \\ \varepsilon & \text{if } \delta(t)[i] = 0 \\ - & \text{if } \delta(t)[i] < 0 \end{cases}$$

Example: $T = \{t_1, t_2, t_3\}$, $\delta(t_1) = 3$, $\delta(t_2) = 0$, and $\delta(t_3) = -1$. Then $\varphi_1^\delta(t_1 t_2 t_3 t_3) = + - -$.

An initialized counter machine $(\mathcal{S}, I)$ is called reversal-bounded if there exists $r \in \mathbb{N}$ such that for any $\pi \in T^*$:

$$\mathrm{post}(\pi, I) \neq \emptyset \implies \varphi_i^\delta(\pi) \in (\{+\}^* \cup \{-\}^*)^{\leq r}$$

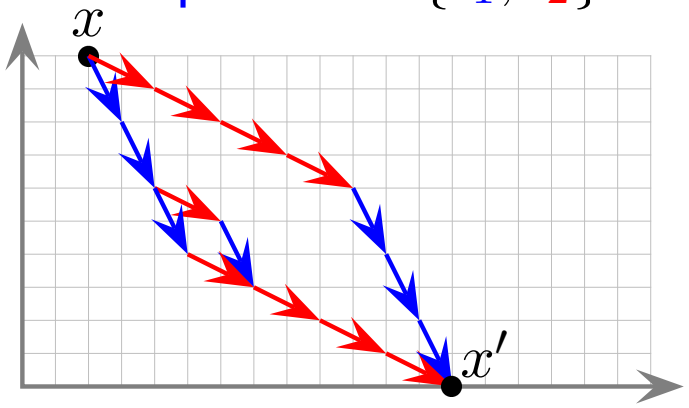
An counter machine $\mathcal{S}$ is called globally reversal-bounded if $(\mathcal{S}, Q \times \mathbb{N}^n)$ is reversal-bounded.

Reversal-bounded (2/2)

Thm: Every initialized reversal-bounded counter machine is flat. Every globally reversal-bounded counter machine is globally flat.

Key ideas:

- Reduce to the case $\mathrm{post}(\pi, I) \neq \emptyset$ implies $\varphi_i^{\delta}(\pi) \in \{+\}^* \cup \{-\}^*$.
- Remove the intermediate guards along π.
  Example: $T = \{t_1, t_2\}$ with $\delta(t_1) = (1, -2)$ and $\delta(t_2) = (2, -1)$.



- Extract from the regular langage $\mathcal{L}$ defined by the control graph, an SLPS $\rho \subseteq \mathcal{L}$ such that $\delta(\mathcal{L}) = \delta(\rho)$ with a variant of Parikh's theorem.

Lossy/inserting counter machines

A counter machine $\mathcal{S}$ is called lossy (resp. inserting) when there are loss loops (resp. insertion loops) on each location and for each counter.

Thm: Every initialized lossy test-free counter machine is flat.

Key ideas:

- Karp&Miller's algorithm can be seen as a (deterministic) "refinement" of the generic accelerated post^* computation.
- This accelerated post^* semi-algorithm has a terminating execution iff the initialized counter machine is flat.

Thm: Every initialized inserting counter machine is flat.

Key ideas:

- As $\mathrm{Min}(\mathrm{post}^*(I))$ is finite, we have $\mathrm{post}(\rho_m, I) = \mathrm{Min}(\mathrm{post}^*(I))$ for some finite SLPS ρ_m.
- Append insertion loops to ρ_m.

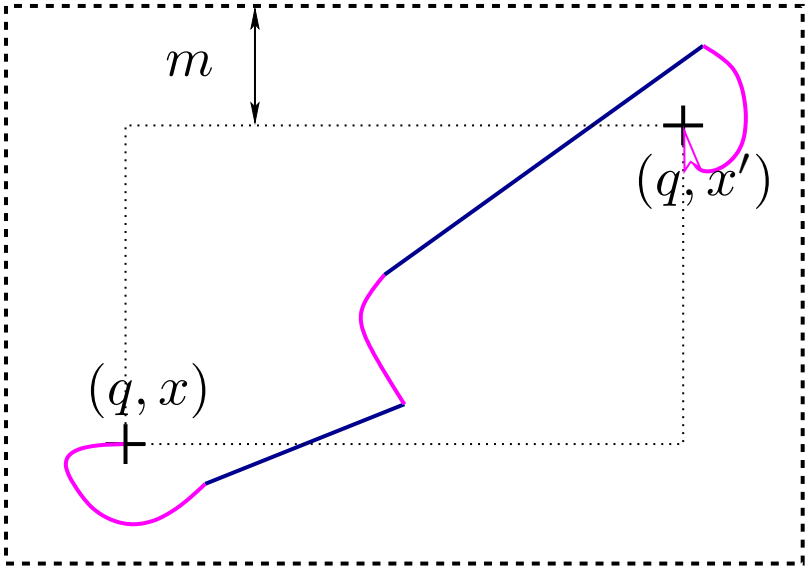
Test-free 2-dim counter machines

A counter machine $\mathcal{S}$ is called test-free when $\mu : T \to \{\geq\}^n$.

Thm: Every test-free 2-dim counter machine is globally flat.

Key ideas:

- Every path $\pi \in T^*$ can be re-ordered into a zigzag-free path:



- For large counter values, we obtain some kind of reversal-bounded counter machine.
- Split $\mathbb{N}^n$ into four zones: $\{[0, c], [c, \infty]\}^2$ and show flatness for each.

Outline

Flat counter automata almost everywhere !

- Introduction.
- Counter machines and acceleration.
- Flat counter machines.
  - Reversal bounded counter machines.
  - Lossy/inserting counter machines.
  - Test-free 2-dim counter machines.

$\Rightarrow$ Flat Petri nets.

  - Cyclic and reversible Petri nets.
  - Regular Petri nets.
  - BPP-nets.

- Conclusion.

Cyclic Petri nets

A Petri net is a test-free counter machine "without control location", i.e. such that $Q = \{q_0\}$.

An initialized Petri net $(\mathcal{S}, I)$ is called cyclic if $I \subseteq \mathrm{post}^*(X)$ for every $X \subseteq \mathrm{post}^*(I)$.

Thm: Every cyclic initialized Petri net is flat.

Key idea:

- $\mathrm{post}^*(I) = \mathrm{post}^*(\{x_0\})$, where $x_0 \in I$.
- $\mathrm{post}^*(I) = \mathrm{Min}(\mathrm{post}^*(\{x_0\})) + (\mathrm{Min}((\mathrm{post}^*(x_0) - x_0) \cap \mathbb{N}^n))^*$.

Reversible Petri nets

A Petri net $\mathcal{S}$ is called globally cyclic if $\xrightarrow{*}$ is symmetric.

Thm: Every globally cyclic Petri net is globally flat.

Key idea:

- $\xrightarrow{*}$ is a congruence on $\mathbb{N}^n$ and hence it is semilinear.
- Consider $(x, x') + \{(p_1, p_1'), \ldots, (p_k, p_k')\} \subseteq \xrightarrow{*}$.
- $x \xrightarrow{\pi_0} x'$ and $x + p_i \xrightarrow{\pi_i} x' + p_i' \xrightarrow{\overline{\pi_i}} x + p_i$.
- Take $\rho = (\pi_1 \overline{\pi_0})^* \ldots (\pi_k \overline{\pi_0})^* \cdot \pi_0$.

A Petri net $\mathcal{S}$ is called reversible if for every $t \in T$, there is $t' \in T$ with $\xrightarrow{t'} = (\xrightarrow{t})^{-1}$.

Thm: Every reversible Petri net is globally flat.

Persistent and conflict-free Petri nets

An initialized Petri net $(\mathcal{S}, I)$ is called persistent if for any $x \in \mathrm{post}^*(I)$:

$$x \xrightarrow{t_1} \text{ and } x \xrightarrow{t_2} \implies x \xrightarrow{t_1 t_2}$$

Thm: Every semilinearly-initialized persistent Petri net is flat.

Key idea:

- Use the proof in [Landweber&Robertson'78] showing semilinearity of post^* for persistent Petri nets.

A Petri net $\mathcal{S}$ is called conflict-free if $(\mathcal{S}, Q \times \mathbb{N}^n)$ is persistent.

Thm: Every conflict-free Petri net is globally flat.

Key idea:

- Duplicate counters: the new counters remain unchanged (not used).
- Use the semilinear set $I = \{(x, x') \in \mathbb{N}^{2n} \mid x = x'\}$.

Regular Petri nets

A singly initialized Petri net $(\mathcal{S}, \{x_0\})$ is said regular if the following trace language $\mathcal{L}$ is regular:

$$\mathcal{L} = \{\pi \in T^* \mid \mathrm{post}(\pi, \{x_0\}) \neq \emptyset\}$$

Thm: Every regular singly initialized Petri net is flat.

Key idea:

- Extract from $\mathcal{L}$ an SLPS $\rho \subseteq \mathcal{L}$ such that $\delta(\mathcal{L}) = \delta(\rho)$ with a variant of Parikh's theorem.

BPP-nets

A Petri net $\mathcal{S}$ is called a BPP-net if for any $t \in T$, we have:

$$\mu(t) = (0, \ldots, 0, 1, 0, \ldots, 0)$$

Thm[Fribourg&Olsen'97]: Every BPP-net is globally flat.

Key idea: Let R be defined by $t_1 R t_2$ iff $\mu(t_1) + \delta(t_1) \geq \mu(t_2)$.

- $\mathrm{post}(t_1, \{x\}) \neq \emptyset$ implies $\mathrm{post}(t_1 t_2, \{x\}) \neq \emptyset$ for any $t_1 R t_2$.
- Moreover if $\theta = t_1 \ldots t_n$ with $t_1 R \cdots R t_n R t_1$, then $\delta(\theta) \geq 0$.
- Build an SLPS $\rho = \theta_1^* \ldots \theta_k^*$ where $\theta_i \in T$ or $\theta_i = t_1 \ldots t_n$ with t_1, ..., t_n 2 by 2 distincts and $t_1 R \cdots R t_n R t_1$.

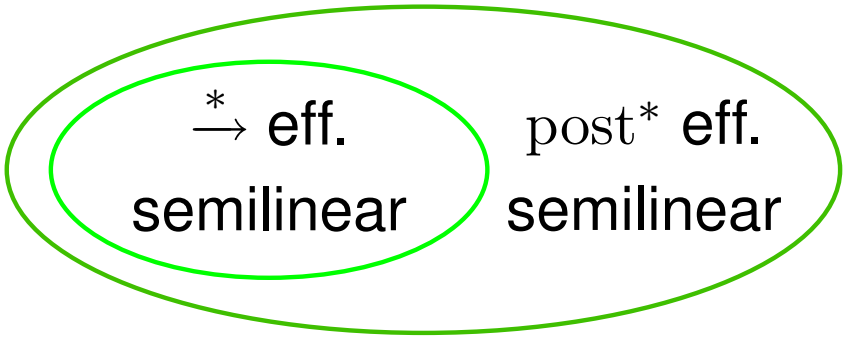
Outline

Flat counter automata almost everywhere !

- Introduction.
- Counter machines and acceleration.
- Flat counter machines.
  - Reversal bounded counter machines.
  - Lossy/inserting counter machines.
  - Test-free 2-dim counter machines.
- Flat Petri nets.
  - Cyclic and reversible Petri nets.
  - Regular Petri nets.
  - BPP-nets.

$\Rightarrow$ Conclusion.

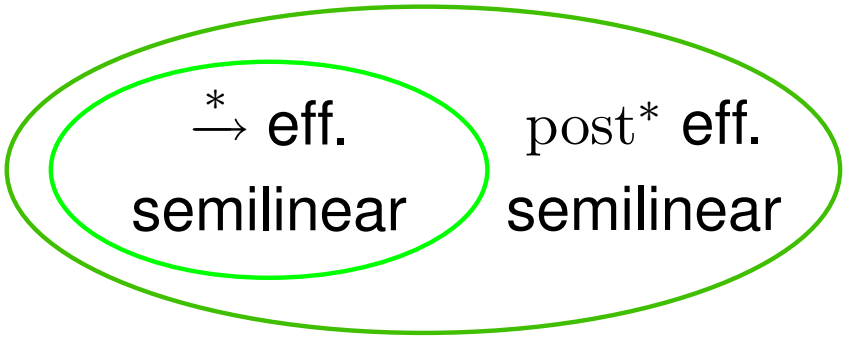
Summary



Reversible Petri nets
BPP nets

Cyclic Petri nets
Persistant Petri nets
Conflict-free Petri nets
Regular Petri nets
Reversal-bounded counter machines
Test-free 2-counter machines
Lossy test-free counter machines

Summary



Reversible Petri nets
Conflict-free Petri nets
BPP nets
Test-free 2-counter machines

Cyclic Petri nets
Persistant Petri nets
Regular Petri nets
Reversal-bounded counter machines
Lossy test-free counter machines

globally flat

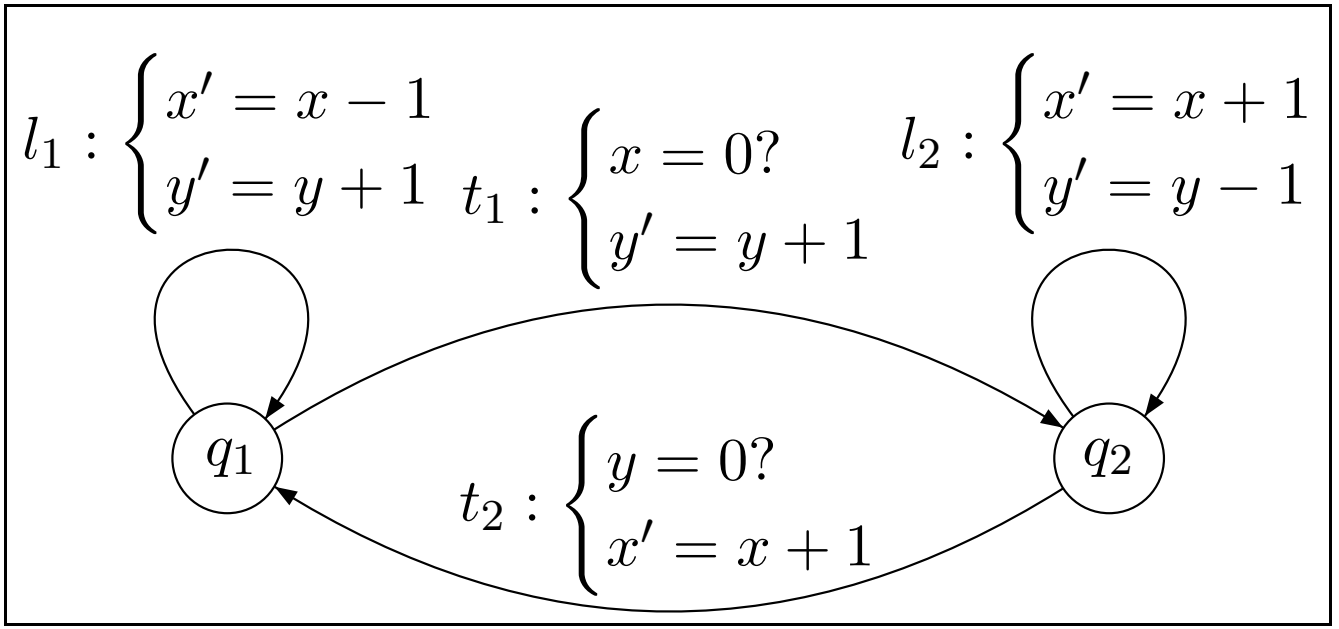
flat

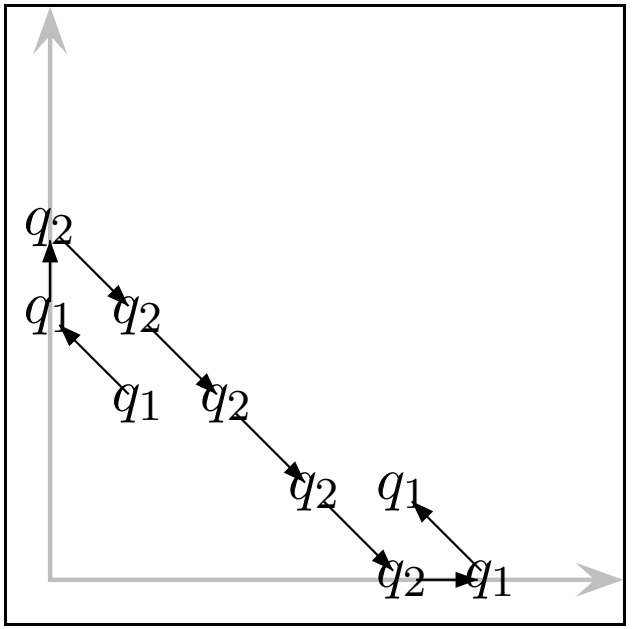
Future work

$\longrightarrow$ Flatness for subclasses of 2-counter machines

Remark: post^* and pre^* are effectively semilinear of lossy 2-counter machines, but these counter machines are not flat in general.

$\longrightarrow$ Extend acceleration techniques to compute post^* for:





$\longrightarrow$ Is flatness decidable for Petri nets ?

$\longrightarrow$ Is flatness equivalent to semilinearity of post^* for Petri nets ?