

Flat counter automata almost everywhere !

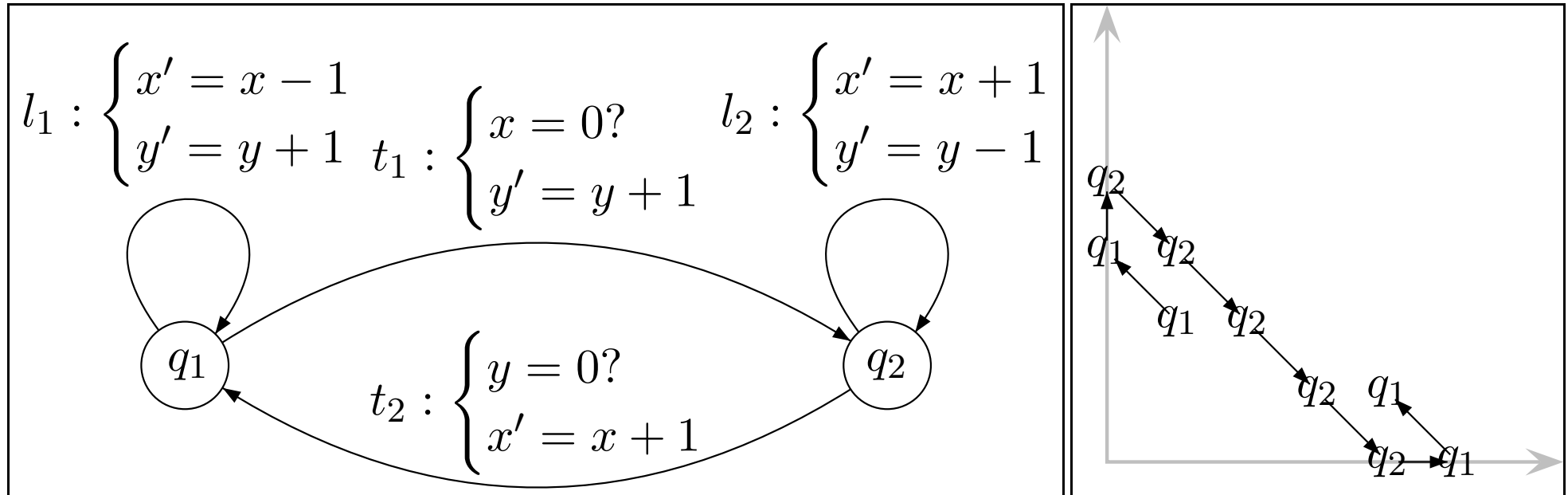
Jérôme Leroux and Grégoire Sutre

Vertecs Project, IRISA / INRIA, Rennes, FRANCE

MVTsi Team, CNRS / LaBRI, Bordeaux, FRANCE

Counter-automata verification

A simple counter-automata:

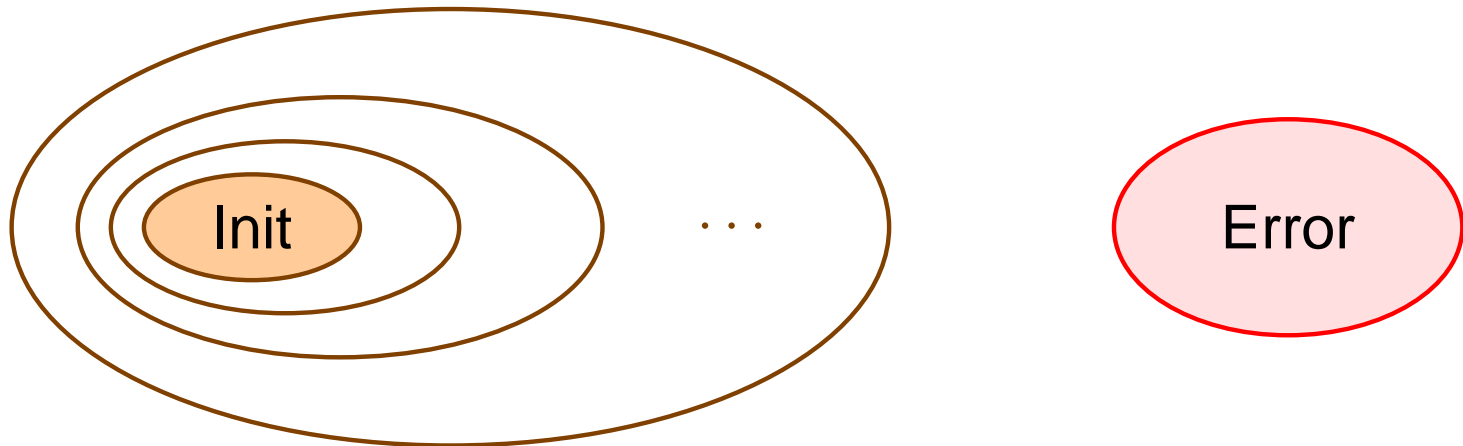


Counter-automata verification naturally appears in practice:

→ Parametrized systems, system abstractions, communication protocols, and so on.

Counter-automata reachability

Verification can often be reduced to the reachability problem.



An algorithm in general ?

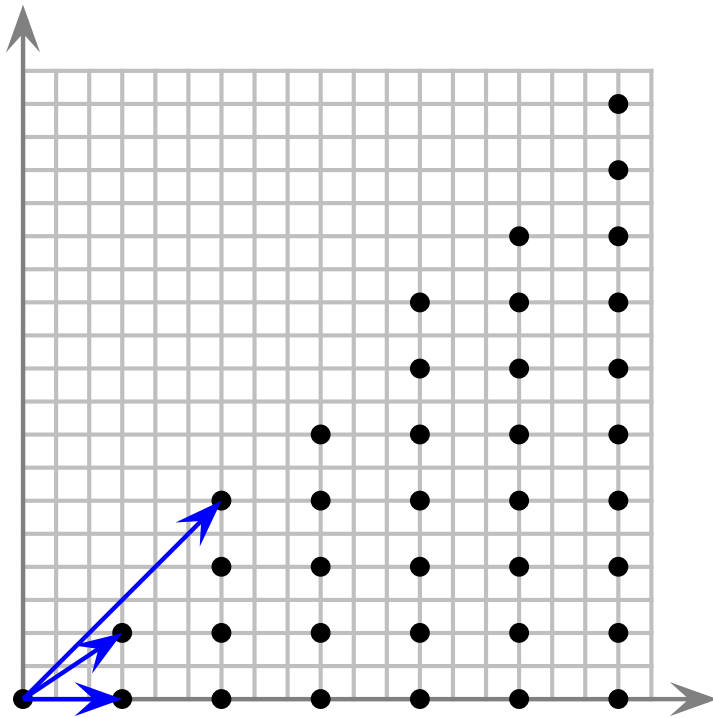
NO ! Because reachability is **undecidable** even for 2-counter automata.

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

Some of these algorithms 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** $b + \{p_1, \dots, p_n\}^*$.

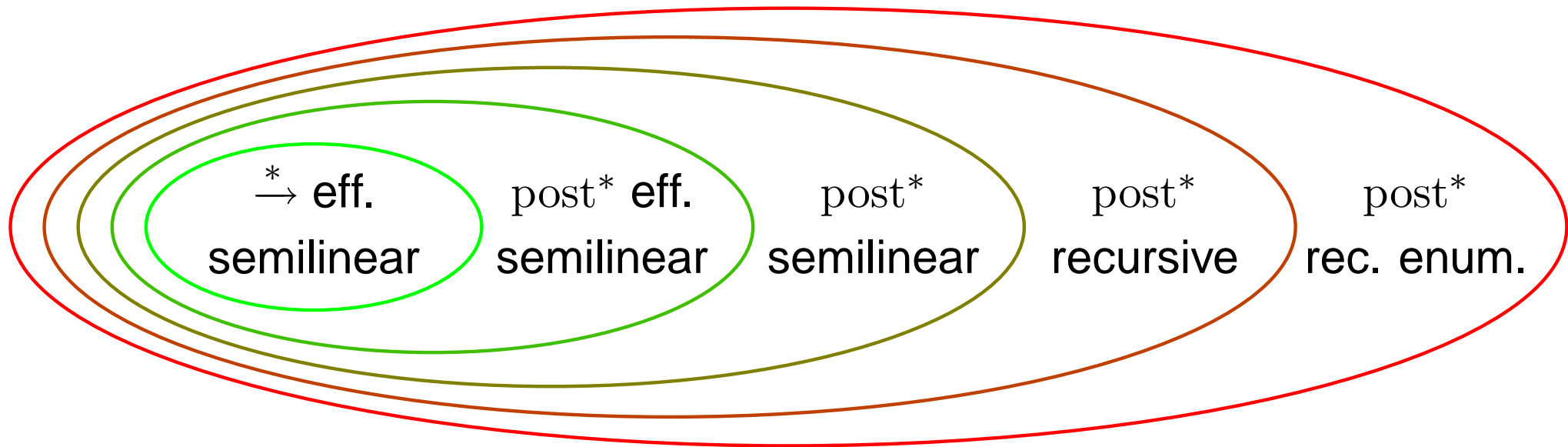


$$\{(0, 0)\} + \{(3, 0), (3, 2), (6, 6)\}^*$$

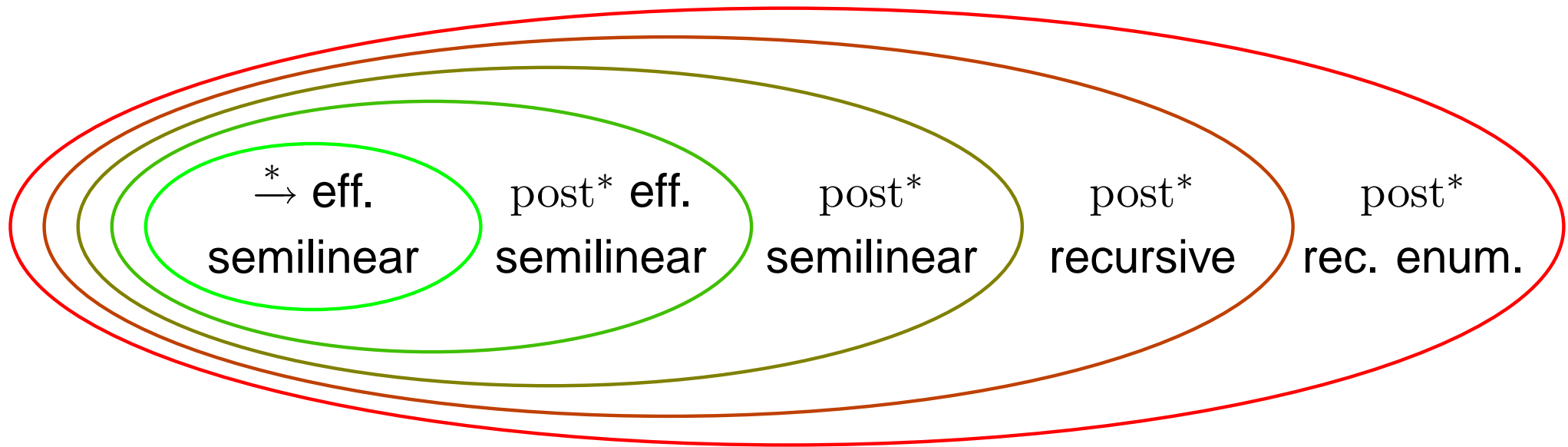
Recall that semilinear sets can be **manipulated** with:

→ Finite sets of basis and periods, Presburger formulas, digit vector automata.

Subclasses of counter automata

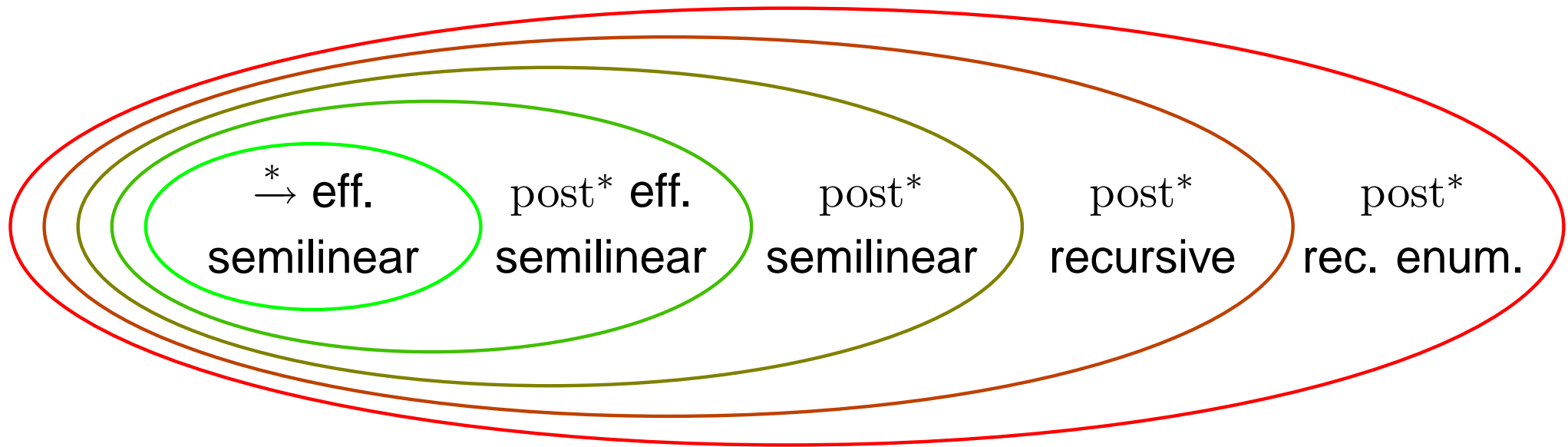


Subclasses of counter automata



2-counter machines

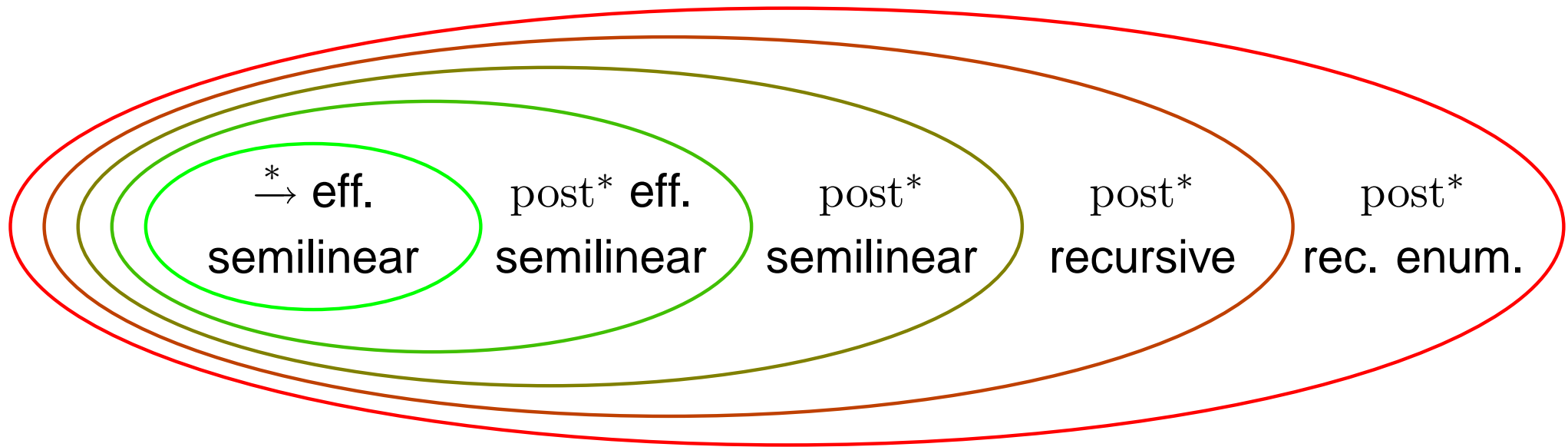
Subclasses of counter automata



Petri nets

2-counter machines

Subclasses of counter automata

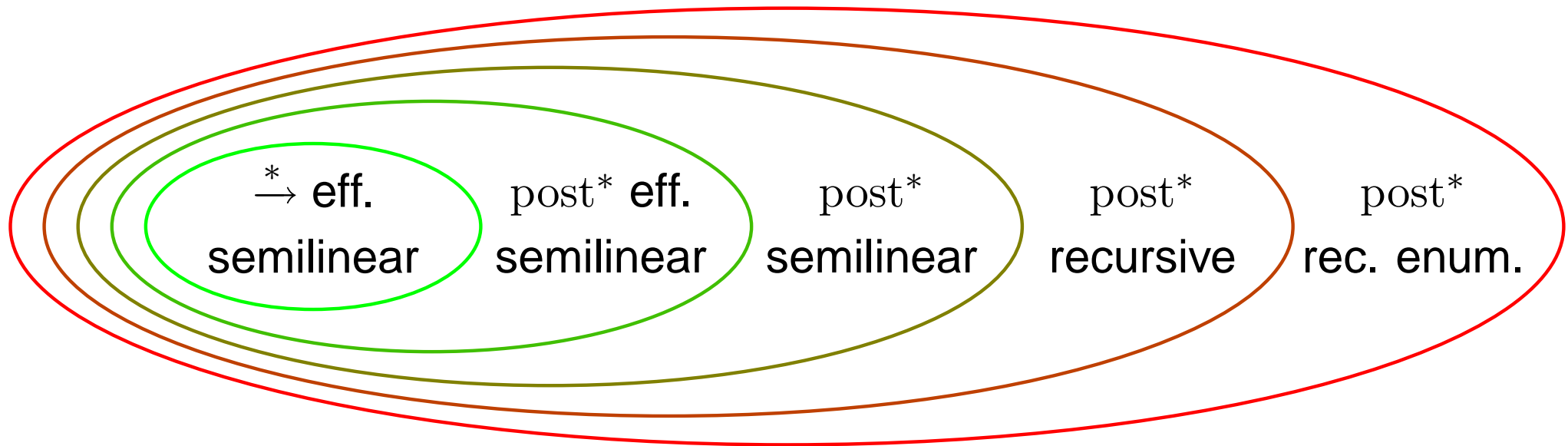


Lossy counter machines

Petri nets

2-counter machines

Subclasses of counter automata



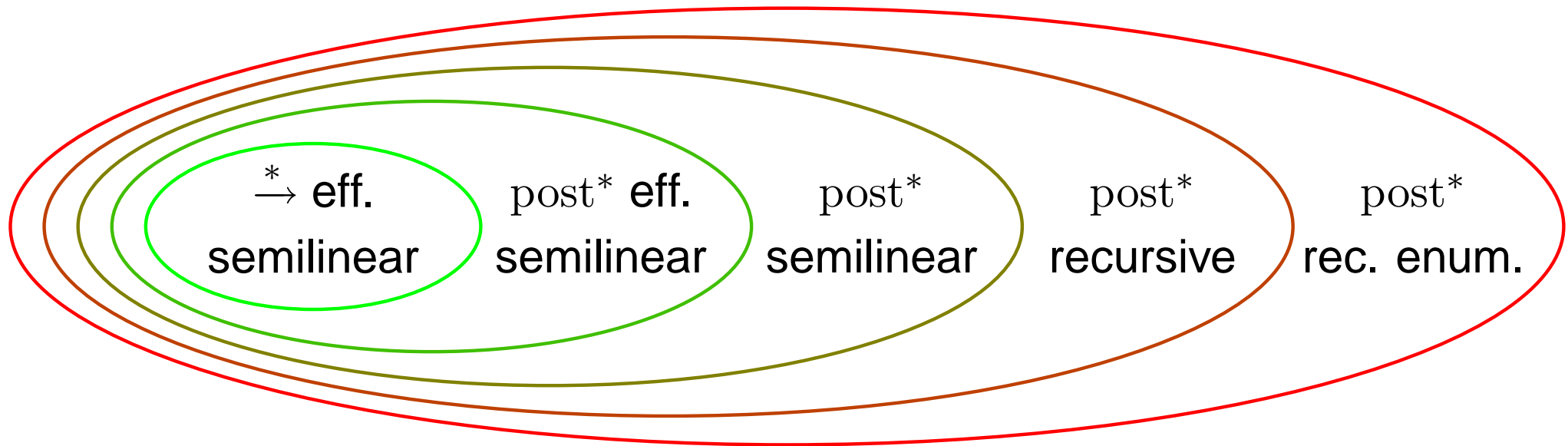
Lossy counter machines

Cyclic Petri nets
Persistent/Conflict-free Petri nets
Regular Petri nets
Restricted 2-counter machines
Lossy test-free counter machines

Petri nets

2-counter machines

Subclasses of counter automata



Reversible Petri nets
BPP nets
2-dim VASS

Lossy counter machines

Cyclic Petri nets
Persistent/Conflict-free Petri nets
Regular Petri nets
Restricted 2-counter machines
Lossy test-free counter machines

Petri nets

2-counter machines

Subclasses of counter automata



Reversible Petri nets
BPP nets
2-dim VASS

Petri nets
Persistent/Conflict-free Petri nets
Regular Petri nets
Restricted 2-counter machines
Lossy test-free counter machines

2-counter machines

Petri nets

2-counter machines

Each (decidable) class has a dedicated algorithm

A generic accelerated algorithm

In **practice** counter automata are not exactly in a known subclass.
→ 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 $\text{post}^*(I)$.

let $X \leftarrow I$ and repeat forever

select one of the following tasks:

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

Implemented in tools: FAST, LASH, TReX.

→ Accelerated symbolic verification works well in **practice**.

Completeness?

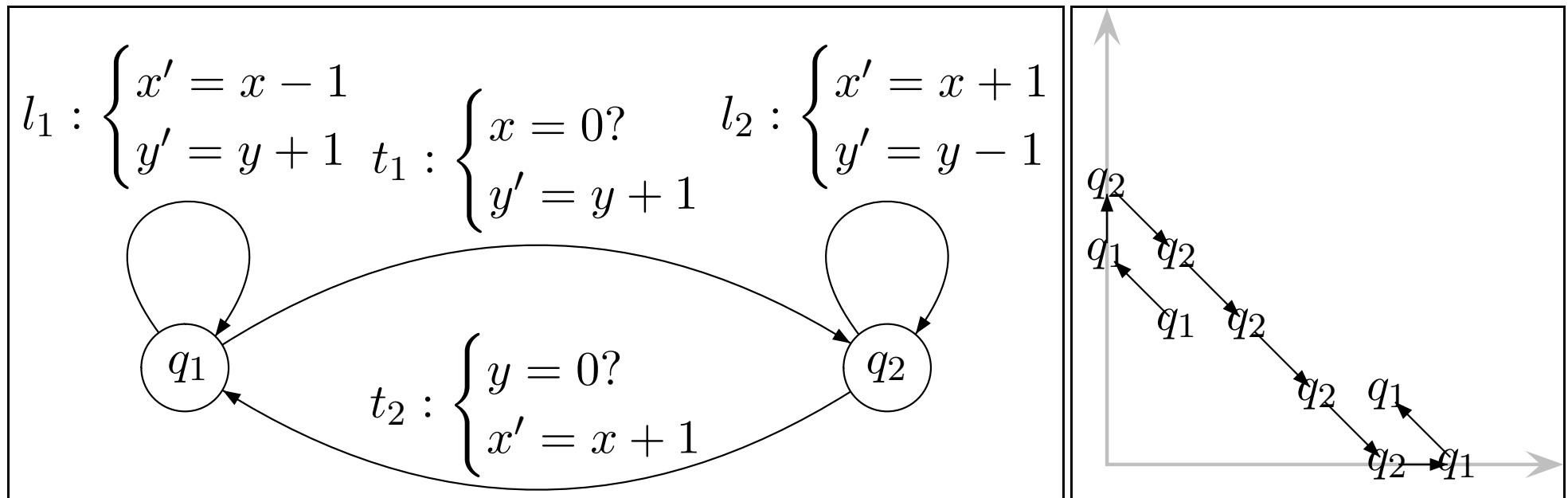
Algorithm does not terminate for non-semilinear counter automata

Does the accelerated algorithm terminate for every semilinear counter automaton?

Completeness?

Algorithm does not terminate for non-semilinear counter automata

Does the accelerated algorithm terminate for every semilinear counter automaton? **No!**

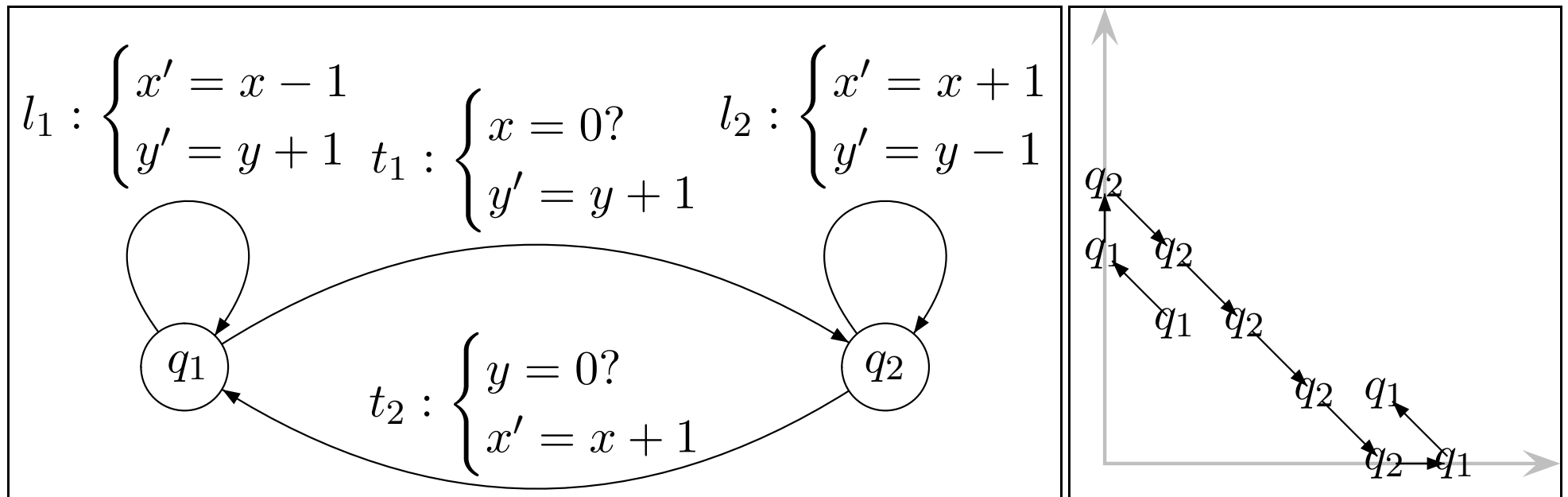


To obtain $\text{post}^*(1, 0)$, we “need” the path: $(l_1)^1 t_1 (l_2)^2 t_2 (l_1)^3 t_1 (l_2)^4 \dots$

Completeness?

Algorithm does not terminate for non-semilinear counter automata

Does the accelerated algorithm terminate for every semilinear counter automaton? **No!**

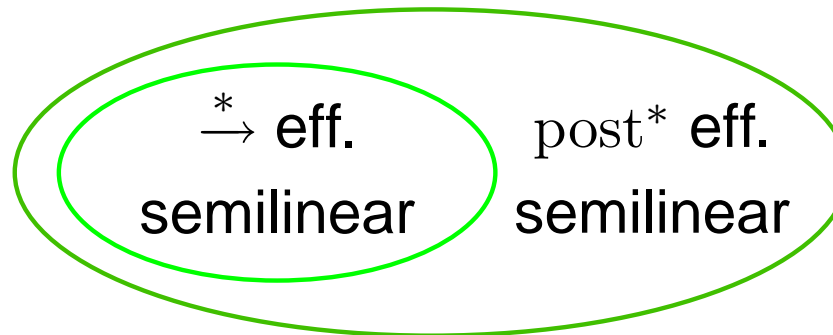


To obtain $\text{post}^*(1, 0)$, we “need” the path: $(l_1)^1 t_1 (l_2)^2 t_2 (l_1)^3 t_1 (l_2)^4 \dots$

Does the accelerated algorithm terminate for every known semilinear class of counter automata?

Partial Completeness?

Does the accelerated algorithm terminate for every known semilinear class of counter automata?



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

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

Outline

Flat counter automata almost everywhere !

- Introduction.
- ⇒ Counter machines, acceleration, flatness.
- 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.
 - Persistent and Conflict-free Petri nets.
 - Regular Petri nets.
 - BPP-nets.
- Conclusion.

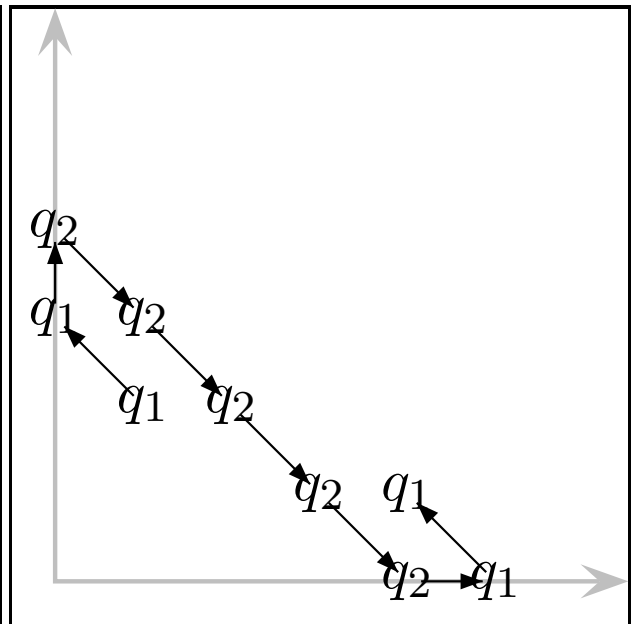
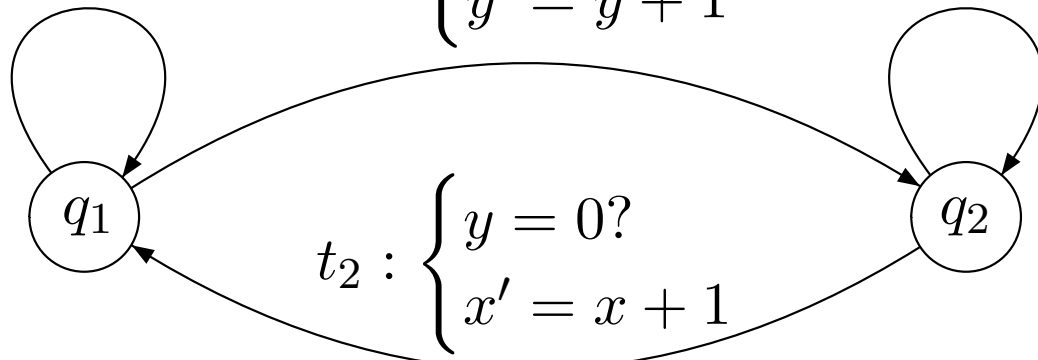
Counter machines

- n counters (ranging over \mathbb{N})
- Finite set of **locations**
- Finite set of **transitions**, labeled with **guarded assignments**
 - guards of the form $\bigwedge x_i \succeq_i \mu_i$, with $\succeq_i \in \{=, \geq\}$
 - assignments of the form $\bigwedge x'_i = x_i + \delta_i$, with $\delta_i \in \mathbb{Z}$

Counter machines

- n counters (ranging over \mathbb{N})
- Finite set of **locations**
- Finite set of **transitions**, labeled with **guarded assignments**
 - guards of the form $\bigwedge x_i \succeq_i \mu_i$, with $\succeq_i \in \{=, \geq\}$
 - assignments of the form $\bigwedge x'_i = x_i + \delta_i$, with $\delta_i \in \mathbb{Z}$

$$l_1 : \begin{cases} x' = x - 1 \\ y' = y + 1 \end{cases} \quad t_1 : \begin{cases} x = 0? \\ y' = y + 1 \end{cases} \quad l_2 : \begin{cases} x' = x + 1 \\ y' = y - 1 \end{cases}$$



Counter machines

- n counters (ranging over \mathbb{N})
- Finite set of **locations**
- Finite set of **transitions**, labeled with **guarded assignments**
 - guards of the form $\bigwedge x_i \succeq_i \mu_i$, with $\succeq_i \in \{=, \geq\}$
 - assignments of the form $\bigwedge x'_i = x_i + \delta_i$, with $\delta_i \in \mathbb{Z}$

- **Configurations**: $(q, x) \in Q \times \mathbb{N}^n$
- $(q, x) \xrightarrow{t} (q', x')$ when
 - control flows from q to q' along transition t
 - counters evolve according to the guarded assignment of transition t

Counter machines

- n counters (ranging over \mathbb{N})
- Finite set of **locations**
- Finite set of **transitions**, labeled with **guarded assignments**
 - guards of the form $\bigwedge x_i \succeq_i \mu_i$, with $\succeq_i \in \{=, \geq\}$
 - assignments of the form $\bigwedge x'_i = x_i + \delta_i$, with $\delta_i \in \mathbb{Z}$

- **Configurations**: $(q, x) \in Q \times \mathbb{N}^n$
- $(q, x) \xrightarrow{t} (q', x')$ when
 - control flows from q to q' along transition t
 - counters evolve according to the guarded assignment of transition t

Initialized n -dim counter machine: (\mathcal{S}, I)

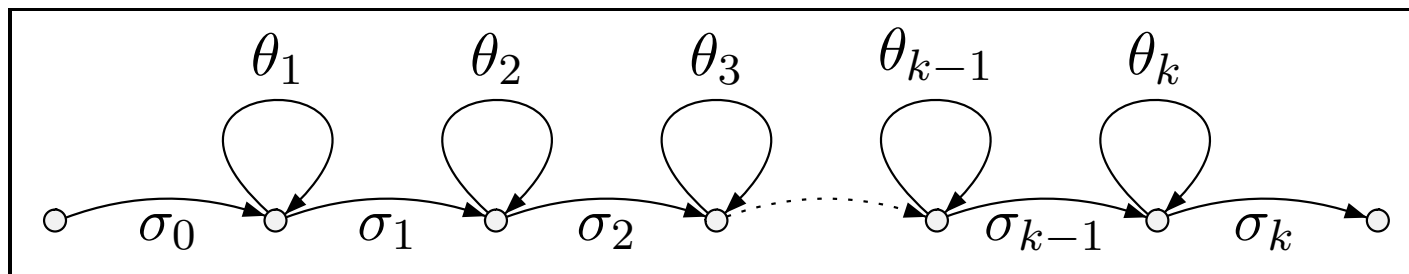
$\xrightarrow{\pi}$ and $\text{post}(\pi, I)$ are naturally defined for any **path** $\pi \in T^*$.

Global reachability relation $\xrightarrow{*}$ is $\xrightarrow{T^*}$.

Reachability set $\text{post}^*(I)$ is $\text{post}(T^*, I)$.

Acceleration for counter machines

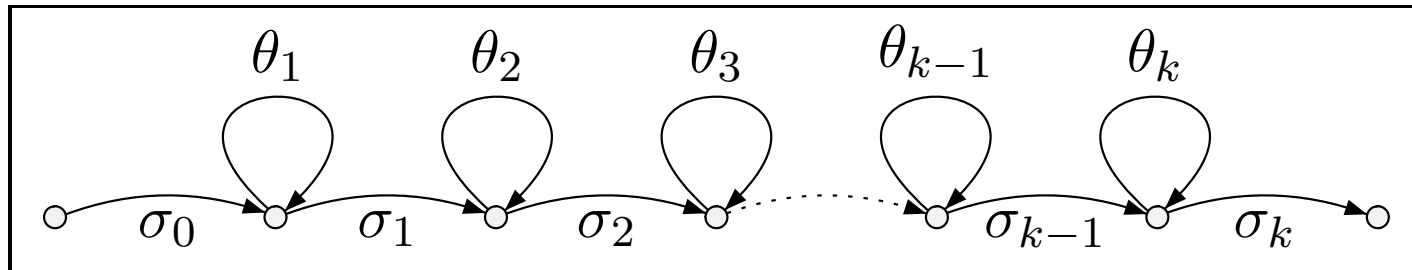
A **semilinear path scheme** $\rho \subseteq T^*$ is a finite union of **linear path schemes** $\sigma_0\theta_1^*\sigma_1 \cdots \theta_k^*\sigma_k$.



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 **forward flat** if $\text{post}^*(I) = \text{post}(\rho, I)$ for some SLPS ρ .

→ Global flatness implies forward flatness for any I . Converse false.

- $\xrightarrow{*}$ is **effectively semilinear** for any **globally flat** counter machine \mathcal{S}
- $\text{post}^*(I)$ is **effectively semilinear** for any **semilinearly initialized forward flat** counter machine (\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 $\text{post}^*(I)$.

let $X \leftarrow I$ and repeat forever

select one of the following tasks:

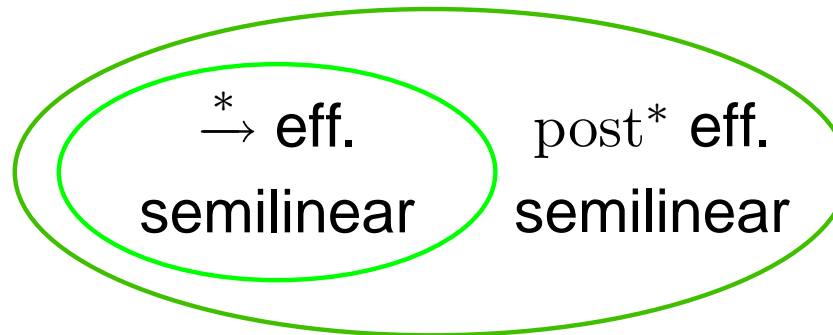
- if $\text{post}(T, X) \subseteq X$ return X
- select $\pi \in T^*$ and $X' \subseteq X$
let $X \leftarrow X \cup \text{post}(\pi^*, X')$
- select $t \in T$ and $X' \subseteq X$
let $X \leftarrow X \cup \text{post}(t, X')$

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

→ The exploration strategy should be “fair” to ensure termination

Partial Completeness?

Are these known semilinear classes of counter automata **flat**?



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

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

Outline

Flat counter automata almost everywhere !

- Introduction.
- Counter machines, acceleration, flatness.
- ⇒ **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.
 - Persistent and Conflict-free Petri nets.
 - Regular Petri nets.
 - BPP-nets.
- Conclusion.

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 forward 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 forward flat.

Thm: Every initialized inserting counter machine is forward flat.

Key ideas:

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

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.
 - Persistent and Conflict-free 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** (S, I) is called **cyclic** if $I \subseteq \text{post}^*(X)$ for every $X \subseteq \text{post}^*(I)$.

Thm: Every cyclic initialized Petri net is forward flat.

Key idea:

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

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 $(x, x') + \{(p_1, p'_1), \dots, (p_k, p'_k)\} \subseteq \xrightarrow{*}$.
- $x \xrightarrow{\pi_0} x'$ and $x + p_i \xrightarrow{\pi_i} x' + p'_i \xrightarrow{\overline{\pi_i}} x + p_i$.
- Take $\rho = (\pi_1 \overline{\pi_0})^* \dots (\pi_k \overline{\pi_0})^* \cdot \pi_0$.

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 $x \in \text{post}^*(I)$:

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

Thm: Every semilinearly-initialized persistent Petri net is forward 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 $I = \{(x, x') \in \mathbb{N}^{2n} \mid x = x'\}$.

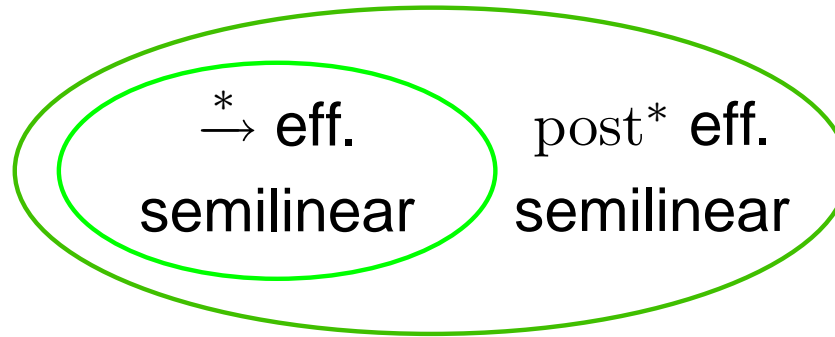
Outline

Flat counter automata almost everywhere !

- Introduction.
- Counter machines, acceleration, flatness.
- 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.
 - Persistent and Conflict-free Petri nets.
 - Regular Petri nets.
 - BPP-nets.

⇒ **Conclusion.**

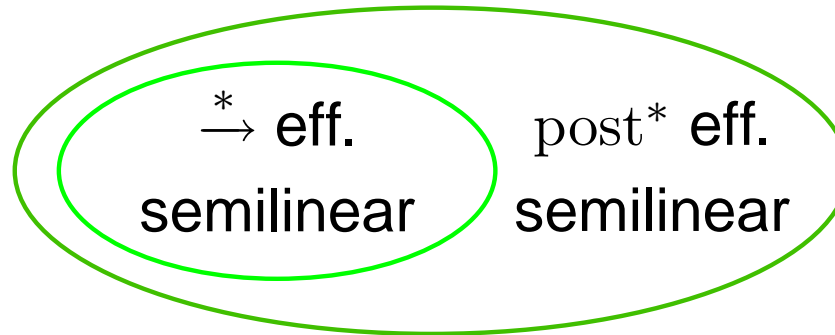
Summary



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

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

Summary



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

globally flat

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

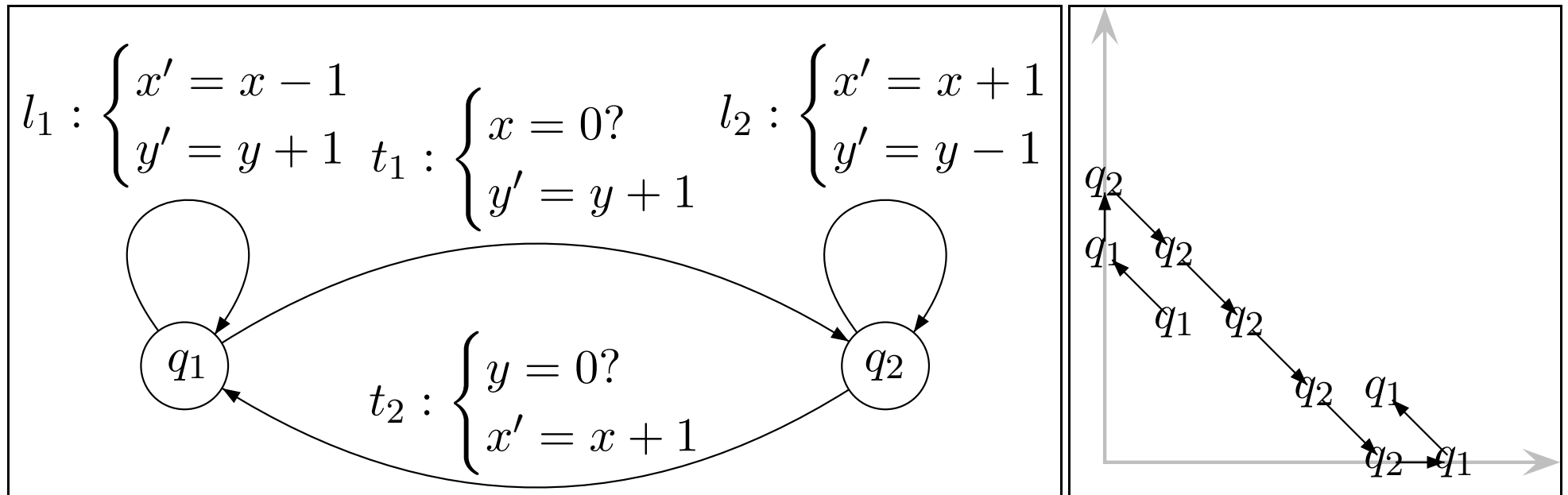
forward flat

Future work

→ 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.

→ **Extend acceleration techniques** to compute post^* for:



→ Is **global/forward flatness decidable** for Petri nets ?

→ Is **flatness equivalent to semilinearity** of $\rightarrow^* / \text{post}^*$ for Petri nets ?

Appendix

Reversal-bounded (1/2)

Recall: T set of transitions, $\delta : T \rightarrow \mathbb{Z}^n$ displacement labeling.

Let $\varphi_i^\delta : T^* \rightarrow \{+, -\}^*$ 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 $\pi \in T^*$:

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

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

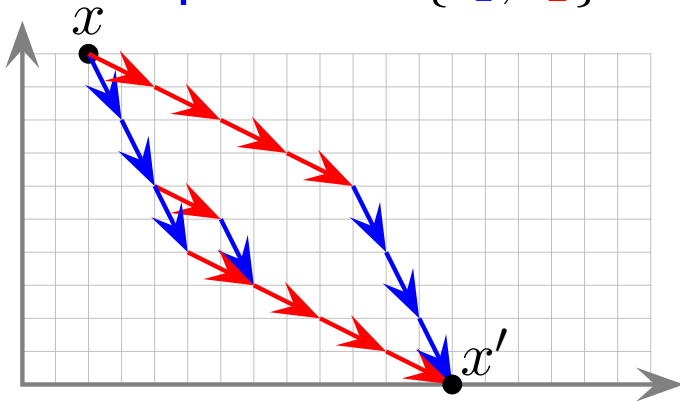
Reversal-bounded (2/2)

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

Key ideas:

- Reduce to the case $\text{post}(\pi, I) \neq \emptyset$ 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 language \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.

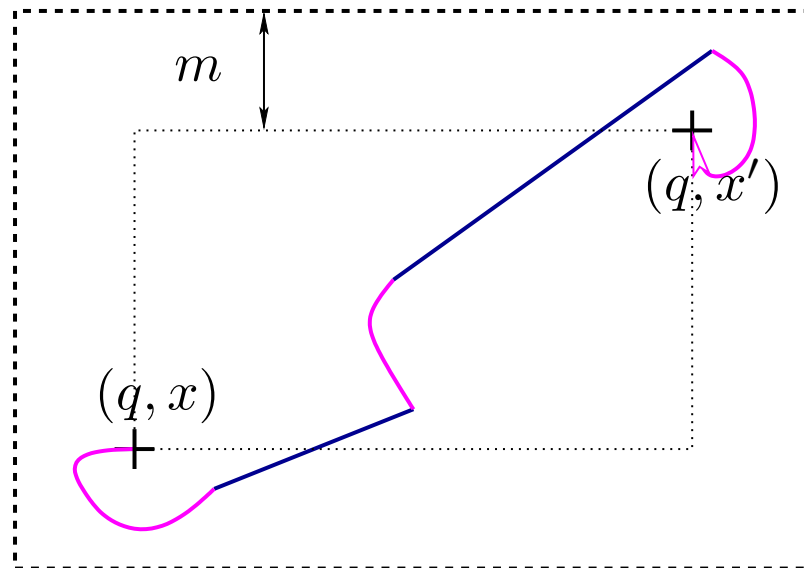
Test-free 2-dim counter machines

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

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.

Regular Petri nets

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

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

Thm: Every regular singly initialized Petri net is forward 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:

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

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

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

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