A Generalised Twinning Property for Minimisation of Cost Register Automata

Laure Daviaud
LIP, ENS Lyon

Joint work with P-A.Reynier and J-M.Talbot
LIF, Aix-Marseille Université

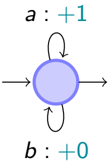
Journées ALGA, 11-12/04/16

A first example

Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.

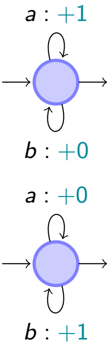
A first example

Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.



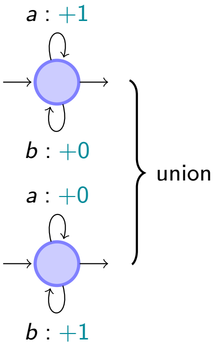
A first example

Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.



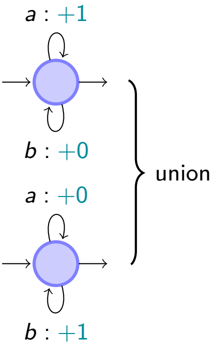
A first example

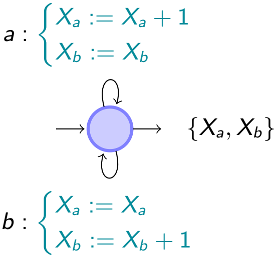
Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.



A first example

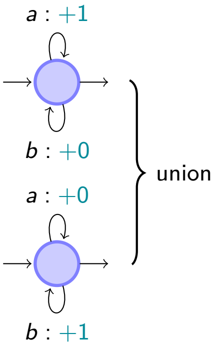
Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.

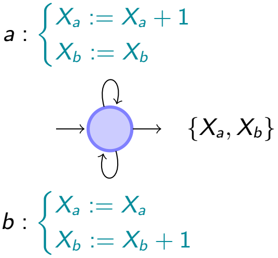




A first example

Given a word $w \in \{a, b\}^*$, compute $\{|w|_a, |w|_b\}$.





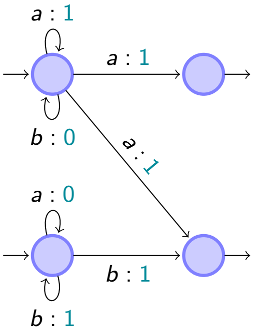
Question: How many values do we need to keep in memory?

Weighted automata (in a restricted case)

Non deterministic finite automaton whose transitions are weighted by elements of a semiring $\longrightarrow$ here $(\mathcal{P}_f(G), \cup, \cdot)$, with $(G, \cdot)$ a group

Weight of a run ρ: $\omega(\rho) =$ product of the weights of the transitions

Function: $w \mapsto \{\omega(\rho) \quad | \rho \text{ accepting run labelled by } w \quad \}$



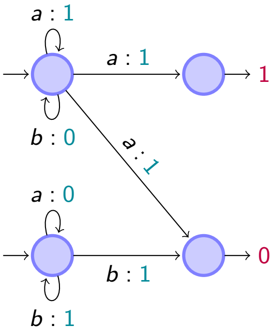
$(\mathbb{Z}, +)$

Weighted automata (in a restricted case)

Non deterministic finite automaton whose transitions are weighted by elements of a semiring $\longrightarrow$ here $(\mathcal{P}_f(G), \cup, \cdot)$, with $(G, \cdot)$ a group
$+$ an function t from the final states to G.

Weight of a run ρ: $\omega(\rho) =$ product of the weights of the transitions

Function: $w \mapsto \{\omega(\rho)t(q) \mid \rho$ accepting run labelled by w ending in $q\}$



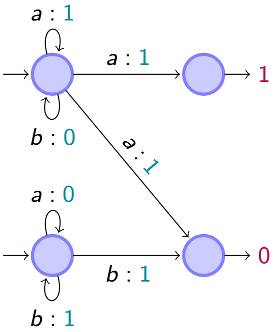
$(\mathbb{Z}, +)$

Weighted automata (in a restricted case)

Non deterministic finite automaton whose transitions are weighted by elements of a semiring $\longrightarrow$ here $(\mathcal{P}_f(G), \cup, \cdot)$, with $(G, \cdot)$ a group
+ an function t from the final states to G.

Weight of a run ρ: $\omega(\rho) =$ product of the weights of the transitions

Function: $w \mapsto \{\omega(\rho)t(q) \mid \rho$ accepting run labelled by w ending in $q\}$



$(\mathbb{Z}, +)$

$[\![\mathcal{W}]\!](w) = \{|w|_a, |w|_a + 1\}$
if w ends with an a

$[\![\mathcal{W}]\!](w) = \{|w|_b\}$
if w ends with a b

Cost register automata (in a restricted case too)

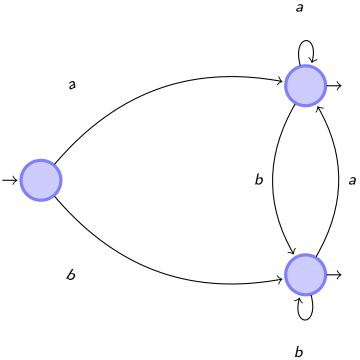
Deterministic finite state machine with registers + an output function

Register updates: $X := Y\alpha$ with $\alpha \in G$.

Cost register automata (in a restricted case too)

Deterministic finite state machine with registers + an output function

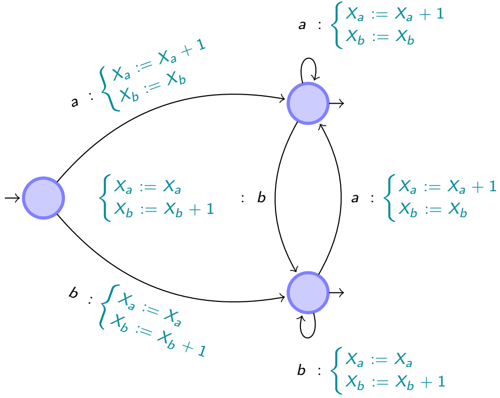
Register updates: $X := Y\alpha$ with $\alpha \in G$.



Cost register automata (in a restricted case too)

Deterministic finite state machine with registers + an output function

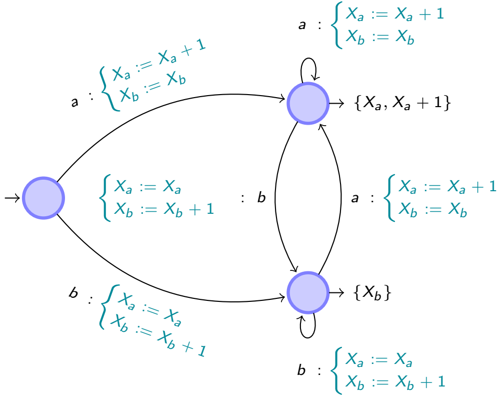
Register updates: $X := Y\alpha$ with $\alpha \in G$.



Cost register automata (in a restricted case too)

Deterministic finite state machine with registers + an output function

Register updates: $X := Y\alpha$ with $\alpha \in G$.



The question

Over an infinitary group,
characterise (effectively) the register complexity
of a function computed by
a finite-valued weighted automaton.

The question

for all $\alpha, \beta, \gamma \in G$ such that $\alpha\beta\gamma \neq \beta$, $|\{\alpha^n\beta\gamma^n \mid n \in \mathbb{N}\}| = +\infty$

ex: $(\mathbb{Z}, +)$, $(\mathbb{R}, \times)$, free group generated by a finite alphabet

Over an infinitary group,
characterise (effectively) the register complexity
of a function computed by
a finite-valued weighted automaton.

The question

for all $\alpha, \beta, \gamma \in G$ such that $\alpha\beta\gamma \neq \beta$, $|\{\alpha^n \beta \gamma^n \mid n \in \mathbb{N}\}| = +\infty$

ex: $(\mathbb{Z}, +)$, $(\mathbb{R}, \times)$, free group generated by a finite alphabet

Over an infinitary group,
characterise (effectively) the register complexity
of a function computed by
a finite-valued weighted automaton.

Minimal number of registers
needed to compute
the function by a CRA

special case in [Alur, Raghothaman]

The question

for all $\alpha, \beta, \gamma \in G$ such that $\alpha\beta\gamma \neq \beta$, $|\{\alpha^n\beta\gamma^n \mid n \in \mathbb{N}\}| = +\infty$

ex: $(\mathbb{Z}, +)$, $(\mathbb{R}, \times)$, free group generated by a finite alphabet

Over an infinitary group,
characterise (effectively) the register complexity
of a function computed by
a finite-valued weighted automaton.

There is ℓ s.t. for all words w, $|f(w)| \leqslant \ell$

ℓ-valued = ℓ-ambiguous

[Filiot, Gentilini, Raskin]

Minimal number of registers needed to compute the function by a CRA

special case in [Alur, Raghothaman]

The question

for all $\alpha, \beta, \gamma \in G$ such that $\alpha\beta\gamma \neq \beta$, $|\{\alpha^n \beta \gamma^n \mid n \in \mathbb{N}\}| = +\infty$

ex: $(\mathbb{Z}, +)$, $(\mathbb{R}, \times)$, free group generated by a finite alphabet

Over an infinitary group,
characterise (effectively) the register complexity
of a function computed by
a finite-valued weighted automaton.

There is ℓ s.t. for all words w, $|f(w)| \leqslant \ell$

ℓ-valued = ℓ-ambiguous

[Filiot, Gentilini, Raskin]

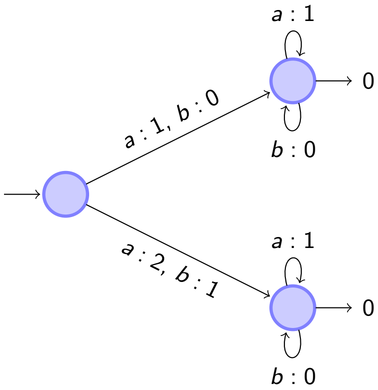
Minimal number of registers needed to compute the function by a CRA

special case in [Alur, Raghothaman]

Why infinitary group ?... See later !

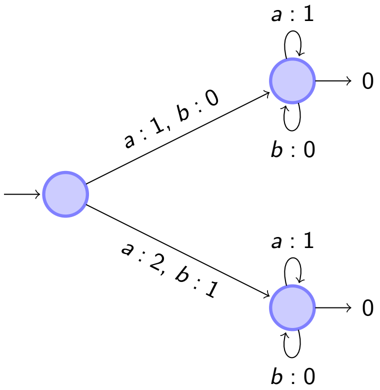
A very very simple example

$$w \mapsto \{|w|_a, |w|_a + 1\} \text{ in } (\mathbb{Z}, +)$$



A very very simple example

$$w \mapsto \{|w|_a, |w|_a + 1\} \text{ in } (\mathbb{Z}, +)$$



Definition

Given $\alpha,\ \beta \in G$, the **delay** between α and β is $\alpha^{-1}\beta$. It is denoted by *delay*(α, β).

Twinning property [Choffrut]

Definition

A weighted automaton satisfies the **twinning property** if for all initial states p, p' and co-accessible states q, q', for all words u, v such that:

$$p \xrightarrow{u:\alpha} q \xrightarrow{v:\beta} q$$

$$p' \xrightarrow{u:\alpha'} q' \xrightarrow{v:\beta'} q'$$

then $\mathit{delay}(\alpha, \alpha') = \mathit{delay}(\alpha\beta, \alpha'\beta')$

A weighted automaton $\mathcal{W}$ satisfies the twinning property
iff $[\![\mathcal{W}]\!]$ has register complexity 1
iff $[\![\mathcal{W}]\!]$ is computed by a deterministic weighted automaton

Generalisation

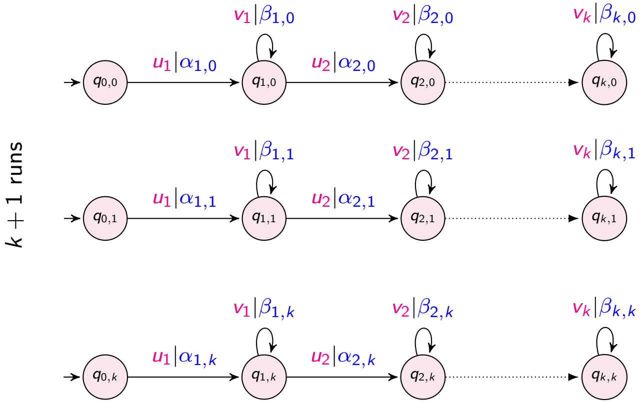
Theorem

Let $\mathcal{W}$ be a finite-valued weighted automaton over an infinitary group, and k be a positive integer.
The following assertions are equivalent:

- $\mathcal{W}$ satisfies the twinning property of order k,
- $[\![\mathcal{W}]\!]$ has register complexity k.

Twinning Property of order k

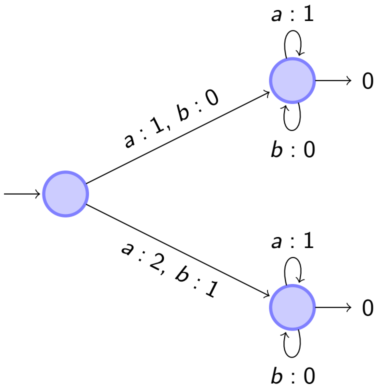
The weighted automaton satisfies the **twinning property of order k** if for all $q_{0,j}$ initial and $q_{k,j}$ co-accessible such that:



there are $j \neq j'$ such that for all $i \in \{1, \dots, k\}$,

$$\mathit{delay}(\alpha_{1,j} \cdots \alpha_{i,j}, \alpha_{1,j'} \cdots \alpha_{i,j'}) = \mathit{delay}(\alpha_{1,j} \cdots \alpha_{i,j}\beta_{i,j}, \alpha_{1,j'} \cdots \alpha_{i,j'}\beta_{i,j'})$$

Twinning property of order k



- Commutative case Vs non commutative case
- Decidability
- Infinitary here !!!

Main result

Theorem

Let $\mathcal{W}$ be a finite-valued weighted automaton over an infinitary group, and k be a positive integer.
The following assertions are equivalent:

- $\mathcal{W}$ satisfies the twinning property of order k,
- $[\![\mathcal{W}]\!]$ has register complexity k,

And everything is effective...

Main result

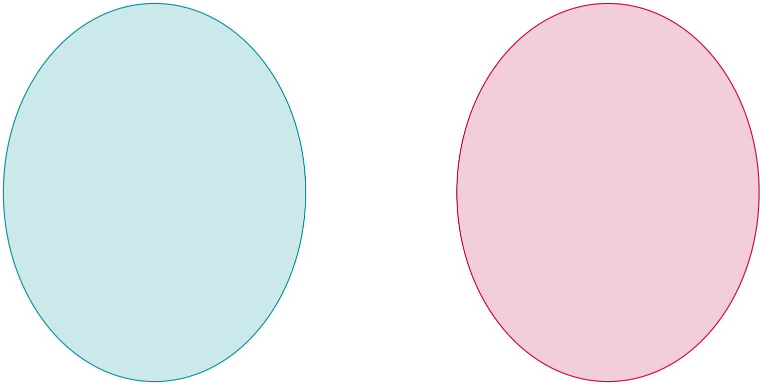
Theorem

Let $\mathcal{W}$ be a finite-valued weighted automaton over an infinitary group, and k be a positive integer.
The following assertions are equivalent:

- $\mathcal{W}$ satisfies the twinning property of order k,
- $[\![\mathcal{W}]\!]$ has register complexity k,
- $[\![\mathcal{W}]\!]$ satisfies the k-bounded variation property.

And everything is effective...

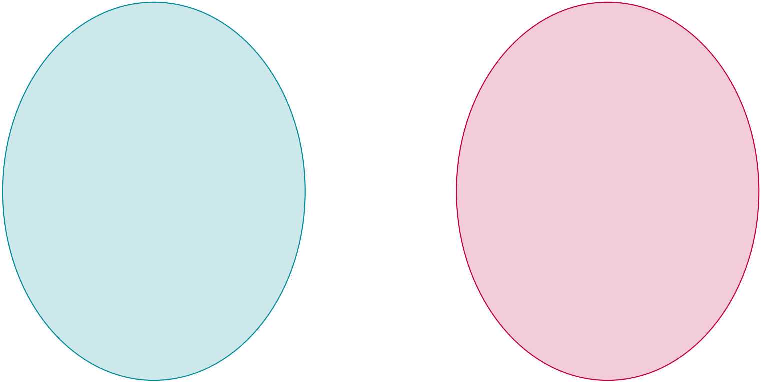
Bounded variation property Special case



A^+ - distance d G - distance d'

Bounded variation property Special case

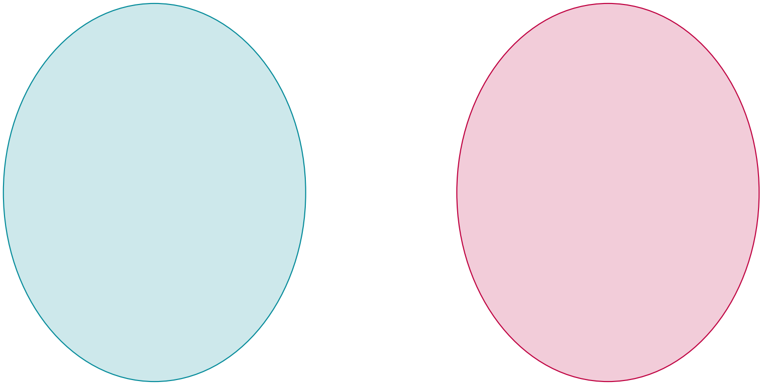
A function $f : A^+ \to G$ satisfies the 1-bounded variation prop if:



A^+ - distance d $\qquad$ G - distance d'

Bounded variation property Special case

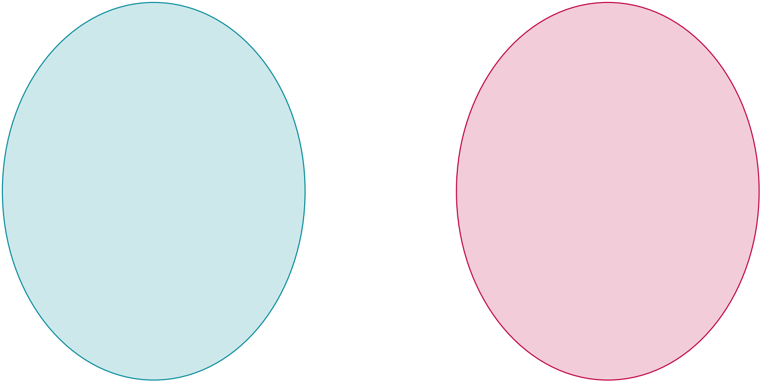
A function $f : A^+ \to G$ satisfies the 1-bounded variation prop if:
for all n,



A^+ - distance d G - distance d'

Bounded variation property Special case

A function $f : A^+ \to G$ satisfies the 1-bounded variation prop if:
for all n, there is N

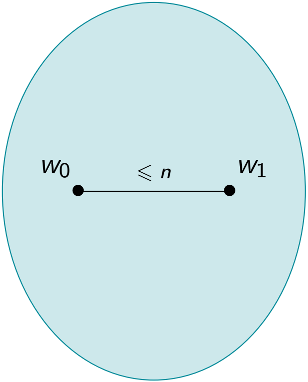


A^+ - distance d G - distance d'

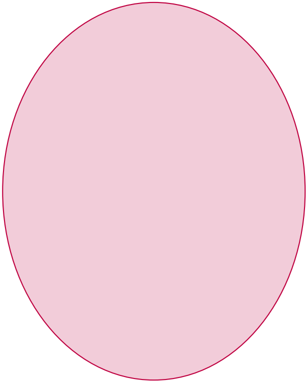
Bounded variation property Special case

A function $f : A^+ \to G$ satisfies the 1-bounded variation prop if:
for all n, there is N such that for all $w_0, w_1 \in A^+$,

$$d(w_0, w_1) \leqslant n$$



A^+ - distance d

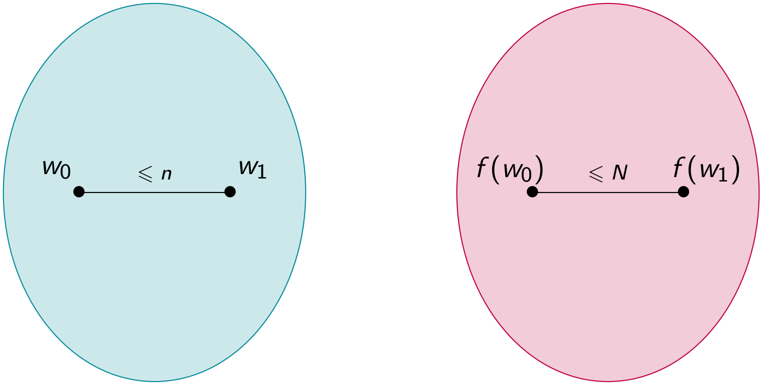


G - distance d'

Bounded variation property Special case

A function $f : A^+ \to G$ satisfies the 1-bounded variation prop if:
for all n, there is N such that for all $w_0, w_1 \in A^+$,

$$d(w_0, w_1) \leqslant n \implies d'(f(w_0), f(w_1)) \leqslant N$$



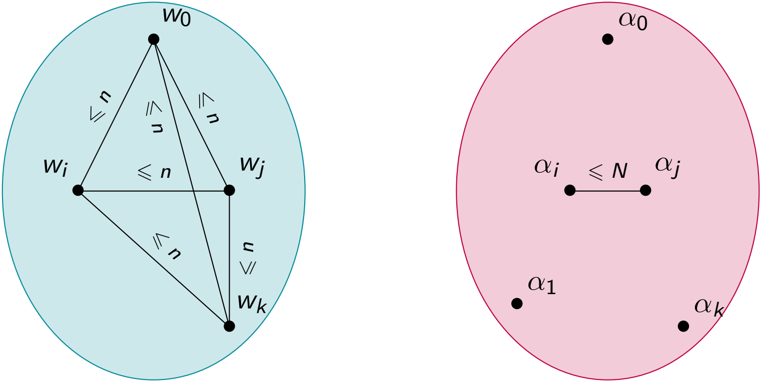
A^+ - distance d

G - distance d'

Bounded variation property

A function $f : A^+ \to \mathcal{P}_f(G)$ satisfies the k-bounded variation if:
for all n, there is N such that
for all $w_0, \ldots, w_k \in A^+$ and all $\alpha_0 \in f(w_0), \ldots, \alpha_k \in f(w_k)$,

for all i, j, $d(w_i, w_j) \leqslant n \implies$ there are $i \neq j$, $d'(f(w_i), f(w_j)) \leqslant N$



Main result

Theorem

Let $\mathcal{W}$ be a finite-valued weighted automaton over an infinitary group, and k be a positive integer.
The following assertions are equivalent:

- $\mathcal{W}$ satisfies the twinning property of order k,
- $[\![\mathcal{W}]\!]$ has register complexity k,
- $[\![\mathcal{W}]\!]$ satisfies the k-bounded variation property.

Also true for transducers !!! but no time to explain it...

Conclusion

. TPk ⇔ BVk ⇔ Register complexity k
for infinitary groups and transducers... (at least)

. Minimisation of cost register automata