The validity of weighted automata

Jacques Sakarovitch

CNRS / Telecom ParisTech

The results presented in this talk are based on a joint work with

Sylvain Lombardy (Univ. Bordeaux)

and have been published in

International Journal of Algebra and Computation, vol. 23 (4).

Work supported by ANR Project 10-INTB-0203 VAUCANSON 2.

Outline

This work addresses, and proposes a solution to,
the problem of ε-transition removal in weighted automata.

The problem lies in effectivity.

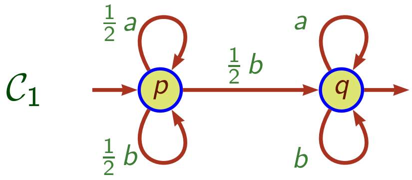
The solution is based on a new, and more constrained,
definition of the validity of weighted automata.

The definition insures that
algorithms are successful on valid automata.

In some (interesting) cases, we are able to establish that
success of algorithms implies validity of automata.

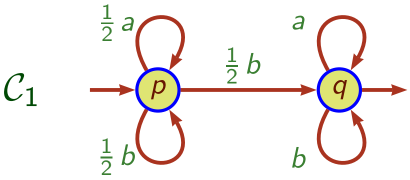
This solution provides a sound theoretical framework for
the algorithms implemented in VAUCANSON.

The weighted automaton model



- Weight of a path c: *product* of the weights of transitions in c
- Weight of a word w: *sum* of the weights of paths with label w

The weighted automaton model



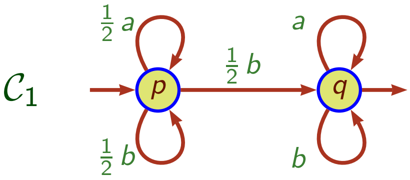
- Weight of a path c: *product* of the weights of transitions in c
- Weight of a word w: *sum* of the weights of paths with label w

$$\xrightarrow{1} p \xrightarrow{\frac{1}{2}\,b} p \xrightarrow{\frac{1}{2}\,a} p \xrightarrow{\frac{1}{2}\,b} q \xrightarrow{1}$$

$$\xrightarrow{1} p \xrightarrow{\frac{1}{2}\,b} q \xrightarrow{a} q \xrightarrow{b} q \xrightarrow{1}$$

$$b\,a\,b \quad \longmapsto \quad \frac{1}{2} + \frac{1}{8} = \frac{5}{8} \; = \langle 0.101 \rangle_2 \qquad |\mathcal{C}_1| \colon A^* \longrightarrow \mathbb{Q}$$

The weighted automaton model



$|\mathcal{C}_1| \in \mathbb{Q}\langle\langle A^* \rangle\rangle$

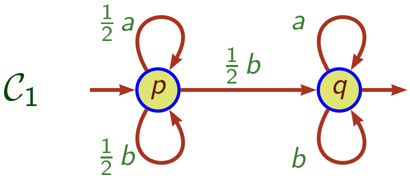
- Weight of a path c: *product* of the weights of transitions in c
- Weight of a word w: *sum* of the weights of paths with label w

$$\xrightarrow{1} p \xrightarrow{\frac{1}{2}\,b} p \xrightarrow{\frac{1}{2}\,a} p \xrightarrow{\frac{1}{2}\,b} q \xrightarrow{1}$$

$$\xrightarrow{1} p \xrightarrow{\frac{1}{2}\,b} q \xrightarrow{a} q \xrightarrow{b} q \xrightarrow{1}$$

$$|\mathcal{C}_1| = \frac{1}{2}\,b + \frac{1}{4}\,a\,b + \frac{1}{2}\,b\,a + \frac{3}{4}\,b\,b + \frac{1}{8}\,a\,a\,b + \frac{1}{4}\,a\,b\,a + \frac{3}{8}\,a\,b\,b + \frac{1}{2}\,b\,a\,a + \ldots$$

The weighted automaton model



$$\mathcal{C}_1 = \langle\, I_1, \underline{E_1}, T_1 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2}\,a + \frac{1}{2}\,b & \frac{1}{2}\,b \\ 0 & a + b \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

The weighted automaton model

$$\mathcal{A} = \langle\, I, \underline{E}, T\,\rangle \qquad \underline{E} = \text{incidence matrix}$$

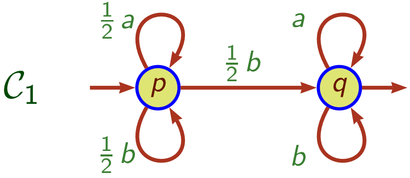
$$\underline{E}_{p,q} = \sum \{\mathbf{wl}(e) \mid e \quad \text{transition from } p \text{ to } q\}$$

$$\underline{E}^{n}_{p,q} = \sum \{\mathbf{wl}(c) \mid c \quad \text{computation from } p \text{ to } q \text{ of length } n\}$$

$$\underline{E}^{*} = \sum_{n\in\mathbb{N}} \underline{E}^{n}$$

$$\underline{E}^{*}_{p,q} = \sum \{\mathbf{wl}(c) \mid c \quad \text{computation from } p \text{ to } q\}$$

The weighted automaton model

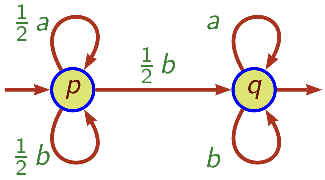


$$\mathcal{C}_1 = \langle\, I_1, \underline{E_1}, T_1 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2}\,a + \frac{1}{2}\,b & \frac{1}{2}\,b \\ 0 & a+b \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

$$\left|\mathcal{C}_1\right| = I_1 \cdot \underline{E_1}^{*} \cdot T_1$$

The weighted automaton model

$\mathcal{C}_1$

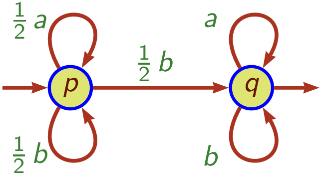


$$\mathcal{C}_1 = \langle\, I_1, \underline{E_1}, T_1 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2}\,a + \frac{1}{2}\,b & \frac{1}{2}\,b \\ 0 & a+b \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

$$\left|\mathcal{C}_1\right| = I_1 \cdot \underline{E_1}^{*} \cdot T_1$$

Every $\mathbb{K}$-automaton defines a series in $\mathbb{K}\langle\langle A^{*}\rangle\rangle$ whose coefficients are effectively computable

The weighted automaton model



$\mathcal{C}_1$

$$\mathcal{C}_1 = \langle\, I_1, \underline{E_1}, T_1 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} \frac{1}{2}\,a + \frac{1}{2}\,b & \frac{1}{2}\,b \\ 0 & a+b \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

$$\left|\mathcal{C}_1\right| = I_1 \cdot \underline{E_1}^{*} \cdot T_1$$

Every $\mathbb{K}$-automaton defines a series in $\mathbb{K}\langle\langle A^* \rangle\rangle$ whose coefficients are effectively computable

Where is the problem ?

The weighted automaton model

We want to deal with automata whose transitions may be labelled by the empty word ε

A basic result in (classical) automata theory

Theorem

Every ε-NFA is equivalent to an NFA

A basic result in (classical) automata theory

Theorem

Every ε-NFA is equivalent to an NFA

Usefulness of ε-transitions:

Preliminary step for many constructions on NFA's:

- Product and star of position automata
- Thompson construction
- Construction of the universal automaton
- Computation of the image of a transducer
- ...

A basic result in (classical) automata theory

Theorem

Every ε-NFA is equivalent to an NFA

Usefulness of ε-transitions:

Preliminary step for many constructions on NFA's:

- Product and star of position automata
- Thompson construction
- Construction of the universal automaton
- Computation of the image of a transducer
- ...

May correspond to the *structure* of the computations

A basic result in (classical) automata theory

Theorem

Every ε-NFA is equivalent to an NFA

Usefulness of ε-transitions:

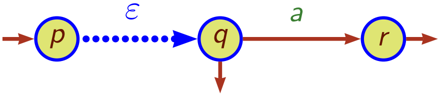
Preliminary step for many constructions on NFA's:

- Product and star of position automata
- Thompson construction
- Construction of the universal automaton
- Computation of the image of a transducer
- ...

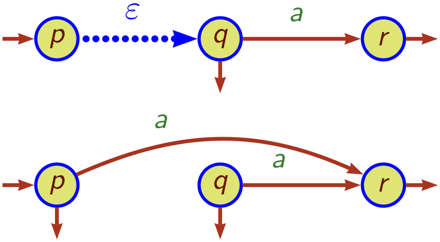
May correspond to the *structure* of the computations

Removal of ε-transitions is implemented in all automata software

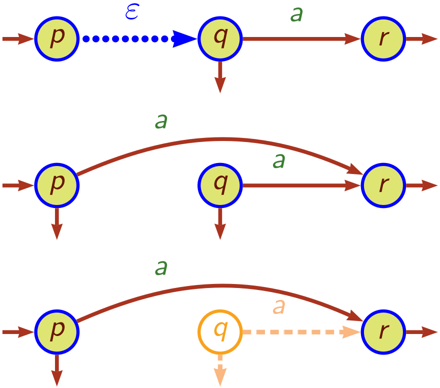
A basic result in (classical) automata theory



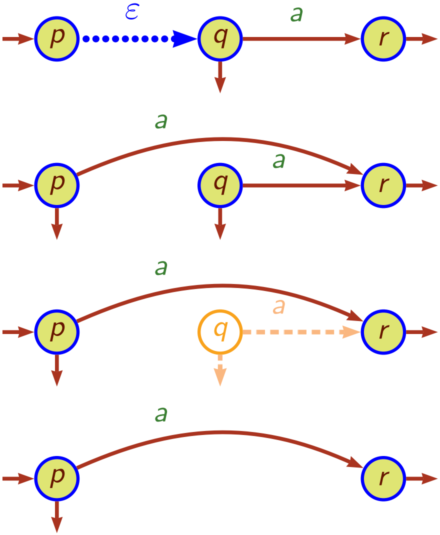
A basic result in (classical) automata theory



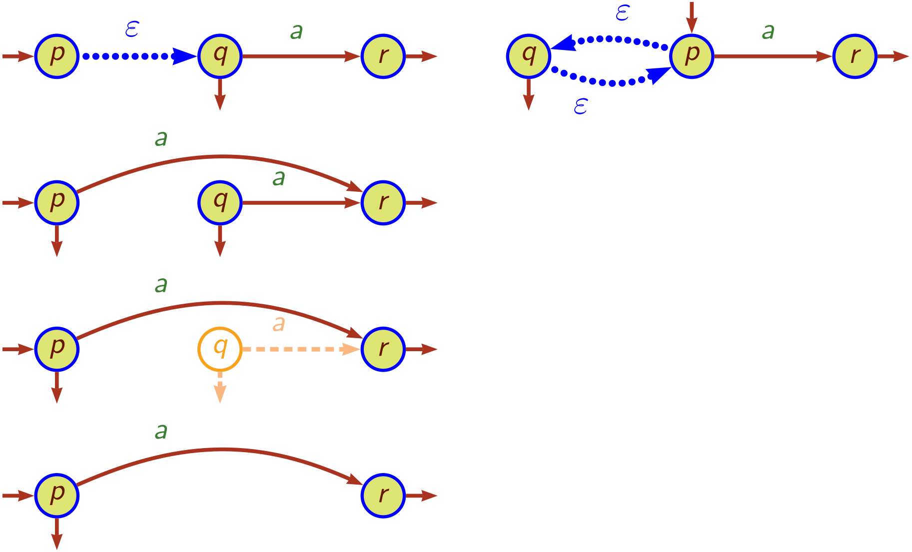
A basic result in (classical) automata theory



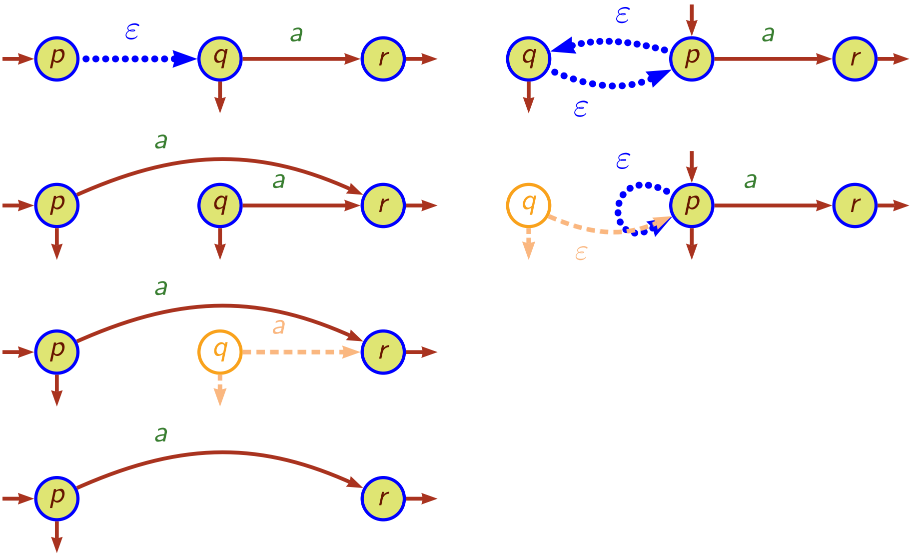
A basic result in (classical) automata theory



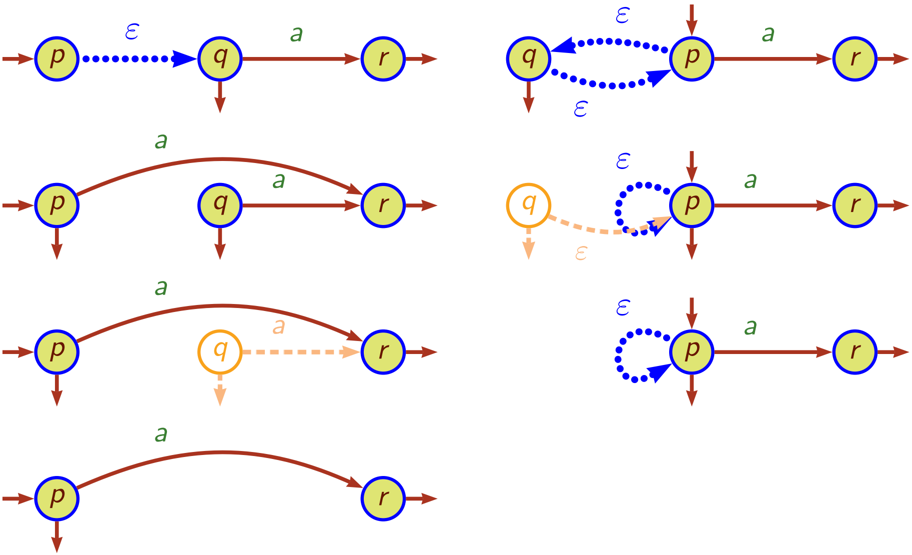
A basic result in (classical) automata theory



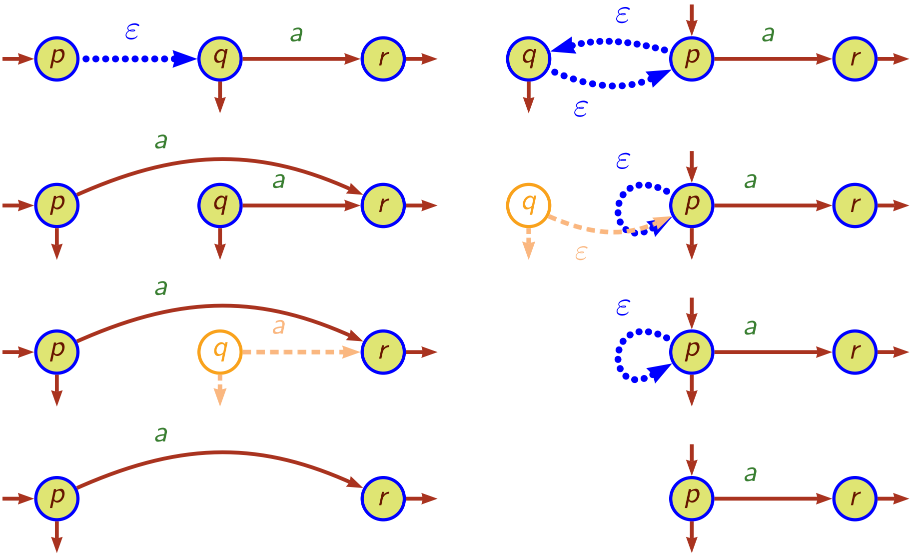
A basic result in (classical) automata theory



A basic result in (classical) automata theory



A basic result in (classical) automata theory



A basic result in (classical) automata theory

Theorem

Every ε-NFA is equivalent to an NFA

A proof

$\mathcal{A} = \langle\, I, \underline{E}, T \,\rangle$ $\qquad$ $\underline{E}$ transition matrix of $\mathcal{A}$

Entries of $\underline{E}$ = subsets of $A \cup \{\varepsilon\}$

$$L(\mathcal{A}) = I \cdot \underline{E}^* \cdot T$$

$$\underline{E} = \underline{E}_0 + \underline{E}_{\mathsf{p}}$$

$$L(\mathcal{A}) = I \cdot (\underline{E}_0 + \underline{E}_{\mathsf{p}})^* \cdot T = I \cdot (\underline{E}_0^* \cdot \underline{E}_{\mathsf{p}})^* \cdot \underline{E}_0^* \cdot T$$

$\mathcal{A} = \langle\, I, \underline{E}, T \,\rangle$ equivalent to $\mathcal{B} = \left\langle\, I, \underline{E}_0^* \cdot \underline{E}_{\mathsf{p}}, \underline{E}_0^* \cdot T \,\right\rangle$

A basic result in (classical) automata theory

Theorem

Every ε*-NFA is equivalent to an NFA*

A proof

$\mathcal{A} = \langle\, I, \underline{E}, T\,\rangle$ $\qquad$ $\underline{E}$ transition matrix of $\mathcal{A}$

Entries of $\underline{E}$ = subsets of $A \cup \{\varepsilon\}$

$$L(\mathcal{A}) = I \cdot \underline{E}^{*} \cdot T$$

$$\underline{E} = \underline{E}_{0} + \underline{E}_{\mathsf{p}}$$

$$L(\mathcal{A}) = I \cdot (\underline{E}_{0} + \underline{E}_{\mathsf{p}})^{*} \cdot T = I \cdot (\underline{E}_{0}^{*} \cdot \underline{E}_{\mathsf{p}})^{*} \cdot \underline{E}_{0}^{*} \cdot T$$

$\mathcal{A} = \langle\, I, \underline{E}, T\,\rangle$ equivalent to $\mathcal{B} = \left\langle\, I, \underline{E}_{0}^{*} \cdot \underline{E}_{\mathsf{p}}, \underline{E}_{0}^{*} \cdot T\,\right\rangle$

One *proof* = several *algorithms* for *computing* $\underline{E}_{0}^{*}$ or $\underline{E}_{0}^{*} \cdot \underline{E}_{\mathsf{p}}$

A basic question in weighted automata theory

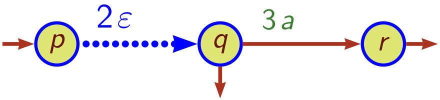
Question

Is every ε-WFA is equivalent to a WFA?

A basic question in weighted automata theory

Question

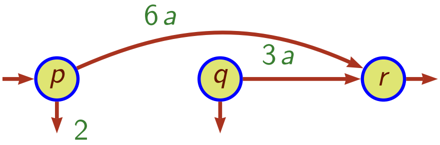
Is every ε-WFA is equivalent to a WFA?



A basic question in weighted automata theory

Question

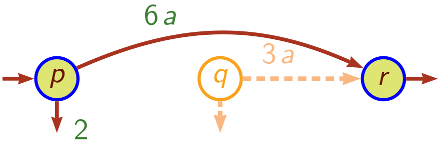
Is every ε-WFA is equivalent to a WFA?



A basic question in weighted automata theory

Question

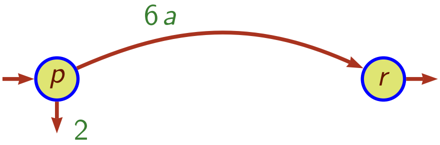
Is every ε-WFA is equivalent to a WFA?



A basic question in weighted automata theory

Question

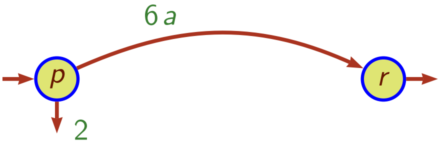
Is every ε-WFA is equivalent to a WFA?

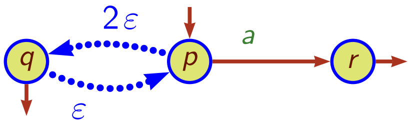


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

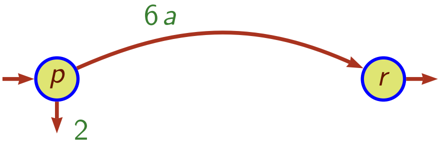


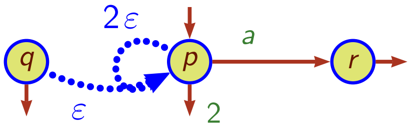


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

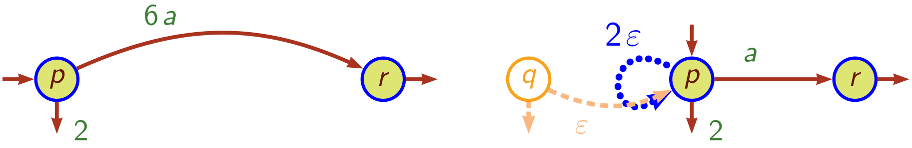




A basic question in weighted automata theory

Question

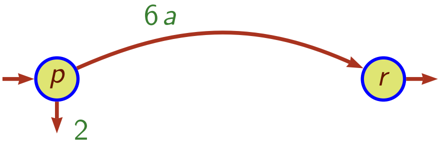
Is every ε-WFA is equivalent to a WFA?

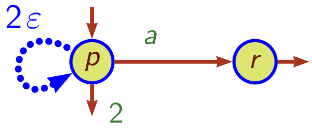


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

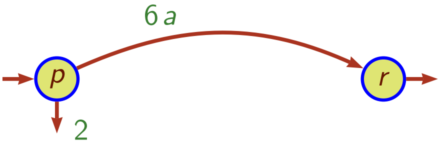


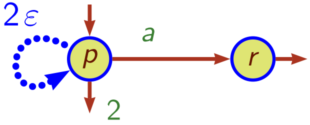


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?



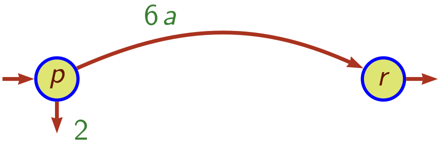


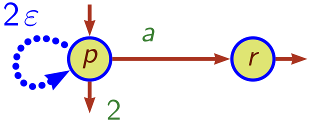
$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?



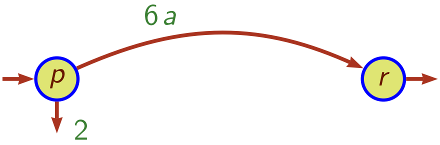


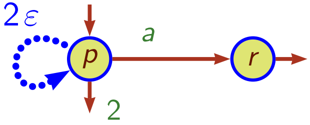
$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

A basic question in weighted automata theory

Question

Is every ε*-WFA is equivalent to a WFA?*





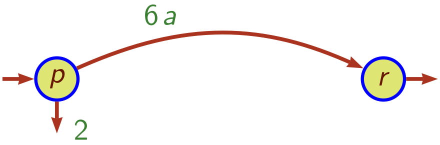
$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

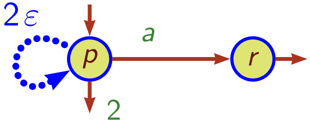
$$\xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \ldots$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

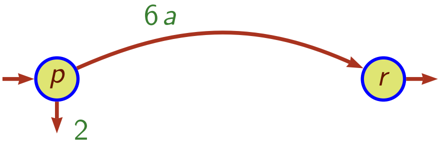
$$\xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

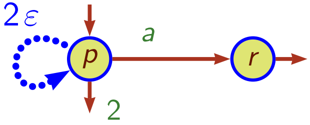
$$a \quad \longmapsto \quad 1 + 2 + 4 + \cdots$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

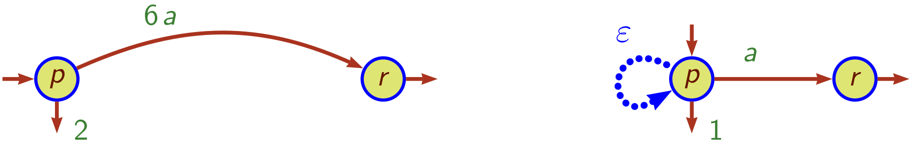
$$\xrightarrow{1} q \xrightarrow{2\,\varepsilon} q \xrightarrow{2\,\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

$a \quad \longmapsto \quad 1 + 2 + 4 + \cdots$ undefined

A basic question in weighted automata theory

Question

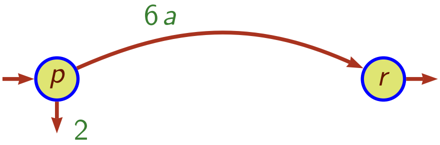
Is every ε-WFA is equivalent to a WFA?

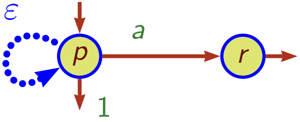


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





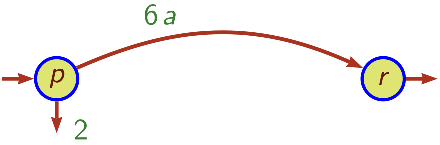
$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \xrightarrow{1} q \xrightarrow{\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

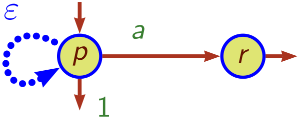
$$\xrightarrow{1} q \xrightarrow{\varepsilon} q \xrightarrow{\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \ldots$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \xrightarrow{1} q \xrightarrow{\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

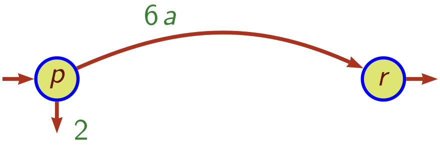
$$\xrightarrow{1} q \xrightarrow{\varepsilon} q \xrightarrow{\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \ldots$$

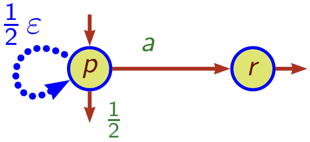
$a \quad \longmapsto \quad 1 + 1 + 1 + \cdots$ undefined

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

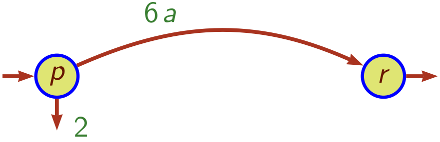


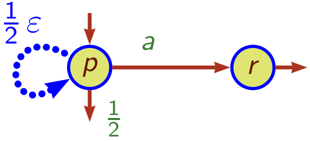


A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





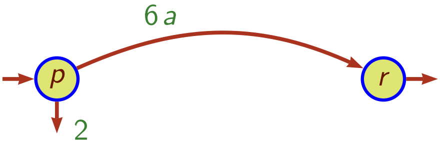
$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

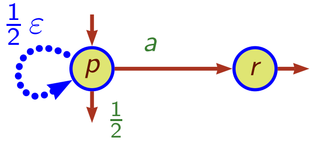
$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

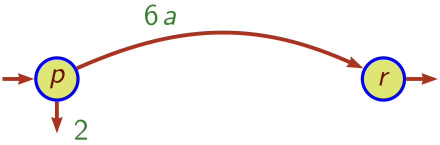
$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

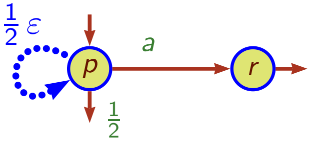
$$a \quad \longmapsto \quad 1 + \frac{1}{2} + \frac{1}{4} + \cdots$$

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

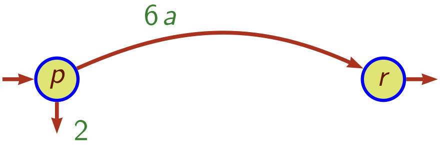
$$a \quad \longmapsto \quad 1 + \frac{1}{2} + \frac{1}{4} + \cdots$$

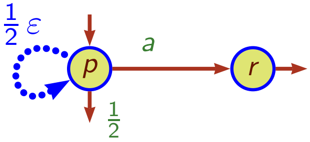
undefined?

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

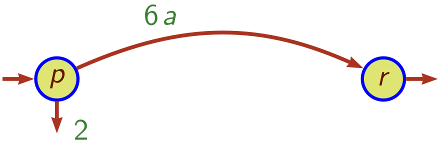
$$a \quad \longmapsto \quad 1 + \frac{1}{2} + \frac{1}{4} + \cdots$$

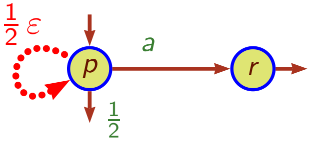
undefined? defined?

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \qquad \ldots$$

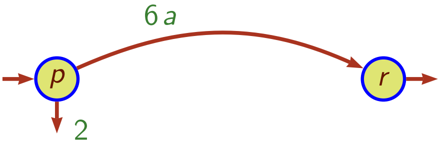
$$a \quad \longmapsto \quad 1 + \frac{1}{2} + \frac{1}{4} + \cdots$$

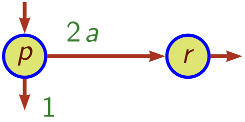
defined

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$$\xrightarrow{1} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad ,$$

$$\xrightarrow{1} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{\frac{1}{2}\varepsilon} q \xrightarrow{a} r \xrightarrow{1} \quad , \quad \ldots$$

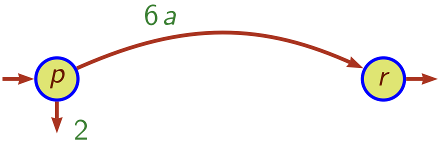
$$a \quad \longmapsto \quad 1 + \frac{1}{2} + \frac{1}{4} + \cdots$$

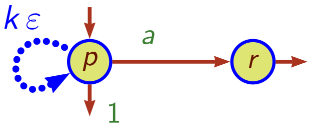
defined

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

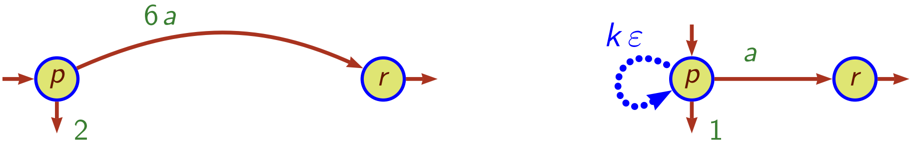




A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

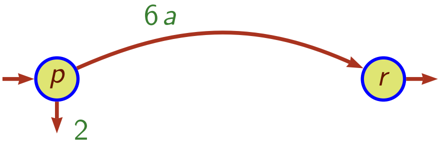


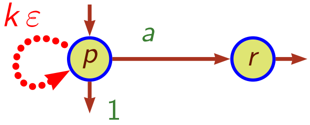
$a \quad \longmapsto \quad k^*$ undefined? defined?

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?



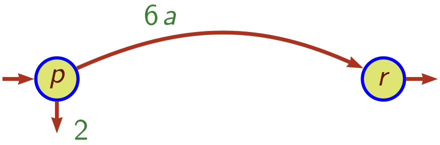


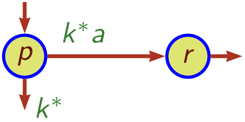
$a \longmapsto k^*$ defined

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?





$a \longmapsto k^*$ defined

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

certainly not !

A basic question in weighted automata theory

Question

Is every ε-WFA is equivalent to a WFA?

certainly not !

New questions

Which ε-WFAs have a *well-defined* behaviour? i.e. are *valid* ?

How to compute the behaviour of a *valid* ε-WFA ?

How to decide if an ε-WFA is *valid*?

A chicken and egg problem

automaton — $\mathcal{A}$ — **valid ?**

algorithm — A — **success ?**

A chicken and egg problem

automaton		**algorithm**
$\mathcal{A}$		A
valid ?		**success ?**
valid	$\Longrightarrow$	**success**

A chicken and egg problem

automaton		algorithm
$\mathcal{A}$		A
valid ?		**success ?**
valid	$\Longrightarrow$	**success**
valid	$\overset{?}{\Longleftarrow}$	**success**

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$u \in A^*$ paths labelled by u in $\mathcal{A}$ possibly infinitely many

$\langle |\mathcal{A}|, u \rangle$ sum of weights of computations labelled by u in $\mathcal{A}$

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$u \in A^*$ paths labelled by u in $\mathcal{A}$ possibly infinitely many

$\langle|\mathcal{A}|, u\rangle$ sum of weights of computations labelled by u in $\mathcal{A}$

Trivial case

Every u in A^* is the label of a finite number of paths

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$u \in A^*$ paths labelled by u in $\mathcal{A}$ possibly infinitely many

$\langle |\mathcal{A}|, u \rangle$ sum of weights of computations labelled by u in $\mathcal{A}$

Trivial case

Every u in A^* is the label of a finite number of paths

$$\Updownarrow$$

no circuits of ε-transitions in $\mathcal{A}$

acyclic $\mathbb{K}$-automata

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$u \in A^*$ paths labelled by u in $\mathcal{A}$ possibly infinitely many

$\langle|\mathcal{A}|, u\rangle$ sum of weights of computations labelled by u in $\mathcal{A}$

Trivial case

Every u in A^* is the label of a finite number of paths

$$\Updownarrow$$

no circuits of ε-transitions in $\mathcal{A}$

acyclic $\mathbb{K}$-automata

First solution

behaviour well-defined $\iff$ acyclic

(Kuich–Salomaa 86, Berstel–Reutenauer 84-88; 11)

Behaviour of weighted automata

$\mathcal{A}$ not acyclic $\Rightarrow$ weight of u in $\mathcal{A}$ may be an infinite sum.

Behaviour of weighted automata

$\mathcal{A}$ not acyclic $\Rightarrow$ weight of u in $\mathcal{A}$ may be an infinite sum.

Second solution

Accepting the idea of infinite sums

Behaviour of weighted automata

$\mathcal{A}$ not acyclic $\Rightarrow$ weight of u in $\mathcal{A}$ may be an infinite sum.

Second solution

Accepting the idea of infinite sums

Topological point of view

Infinite sums are given a meaning via a topology on $\mathbb{K}$

Topology on $\mathbb{K}$ defines a topology on $\mathbb{K}\langle\langle A^* \rangle\rangle$

Topology allows to define summable families in $\mathbb{K}\langle\langle A^* \rangle\rangle$

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T\,\rangle$ possibly with ε-transitions

$\mathsf{P}_{\mathcal{A}}$ set of all paths in $\mathcal{A}$

$|\mathcal{A}|$ well-defined $\iff$ $\mathbf{WL}(\mathsf{P}_{\mathcal{A}})$ summable

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$\mathsf{P}_{\mathcal{A}}$ set of all paths in $\mathcal{A}$

$|\mathcal{A}|$ well-defined $\iff$ $\forall p, q \in Q$ $\mathbf{WL}\big(\mathsf{P}_{\mathcal{A}}(p,q)\big)$ summable

Behaviour of weighted automata

$\mathcal{A} = \langle\, \mathbb{K}, Q, A, E, I, T \,\rangle$ possibly with ε-transitions

$\mathsf{P}_{\mathcal{A}}$ set of all paths in $\mathcal{A}$

$|\mathcal{A}|$ well-defined $\iff$ $\forall p, q \in Q$ $\mathbf{WL}\big(\mathsf{P}_{\mathcal{A}}(p,q)\big)$ summable

Definition taken in previous works (Lombardy, S. 03 –)

- Yields a consistent theory
- Two pitfalls for effectivity
  - *effective computation* of a summable family may not be possible
  - *effective computation* may give values to non summable families

Valid weighted automata

$\mathcal{A} = \langle \mathbb{K}, Q, A, E, I, T \rangle$ possibly with ε-transitions

E^* *free monoid* generated by E

$\mathsf{P}_{\mathcal{A}}$ *set of paths* in $\mathcal{A}$ (local) rational subset of E^*

Definition
R rational family of paths of $\mathcal{A}$ $R \in \mathrm{Rat} E^* \wedge R \subseteq \mathsf{P}_{\mathcal{A}}$

Definition
$\mathcal{A}$ is valid iff
$\forall R$ rational family of paths of $\mathcal{A}$, $\mathbf{WL}(R)$ is summable

Valid weighted automata

Validity implies well-definition of behaviour

Validity implies well-definition of behaviour

Theorem

$\mathcal{A}$ is valid iff the behaviour of every covering of $\mathcal{A}$ is well-defined

Validity implies well-definition of behaviour

Theorem
$\mathcal{A}$ is valid iff the behaviour of every covering of $\mathcal{A}$ is well-defined

Theorem
If $\mathcal{A}$ is valid, then 'every' removal algorithm on $\mathcal{A}$ is successful

Validity implies well-definition of behaviour

Theorem

$\mathcal{A}$ is valid iff the behaviour of every covering of $\mathcal{A}$ is well-defined

Theorem

If $\mathcal{A}$ is valid, then 'every' removal algorithm on $\mathcal{A}$ is successful

Remark

If every (rational) subfamily of a summable family in $\mathbb{K}$ is summable,
then validity is equivalent to well-definition of behaviour

Eg. $\mathbb{R}$, $\mathbb{Q}$.

Validity implies well-definition of behaviour

Theorem

$\mathcal{A}$ is valid iff the behaviour of every covering of $\mathcal{A}$ is well-defined

Theorem

If $\mathcal{A}$ is valid, then 'every' removal algorithm on $\mathcal{A}$ is successful

Remark

If every (rational) subfamily of a summable family in $\mathbb{K}$ is summable,
then validity is equivalent to well-definition of behaviour

Eg. $\mathbb{R}$, $\mathbb{Q}$.

Reminder

We do not know yet how to decide whether
a $\mathbb{Q}$ - or an $\mathbb{R}$ -automaton is valid.

Deciding validity

Straightforward cases

- Non starable semirings (eg. $\mathbb{N}$, $\mathbb{Z}$)

  $\mathcal{A}$ valid $\iff$ $\mathcal{A}$ acyclic
- Complete topological semirings (eg. $\mathcal{N}$) every $\mathcal{A}$ valid
- Rationally additive semirings (eg. $\mathrm{Rat}\, A^*$) every $\mathcal{A}$ valid
- Locally closed commutative semirings every $\mathcal{A}$ valid

Deciding validity

Definition

$\mathbb{K}$ topological, ordered, positive semiring (TOPS)

is star-domain downward closed (SDC) if

$$\forall k, h \in \mathbb{K}\,,\ k < h \qquad h \text{ starable} \implies k \text{ starable}$$

Deciding validity

Definition

$\mathbb{K}$ topological, ordered, positive semiring (TOPS)

is star-domain downward closed (SDC) if

$$\forall k, h \in \mathbb{K}\,,\ k < h \qquad h \text{ starable} \implies k \text{ starable}$$

$\mathbb{N}$, $\mathcal{N}$, $\mathbb{Q}_+$, $\mathbb{R}_+$, $\mathbb{Z}\text{min}$, $\mathrm{Rat}\, A^*$,... are TOPS SDC

$\mathbb{N}_\infty$, (binary) positive decimals,... are TOPS not SDC

Deciding validity

Definition

$\mathbb{K}$ topological, ordered, positive semiring (TOPS)
is star-domain downward closed (SDC) if

$$\forall k, h \in \mathbb{K}\,,\ k < h \qquad h \text{ starable} \implies k \text{ starable}$$

$\mathbb{N}$, $\mathcal{N}$, $\mathbb{Q}_+$, $\mathbb{R}_+$, $\mathbb{Z}\text{min}$, $\mathrm{Rat}\, A^*$,... are TOPS SDC

$\mathbb{N}_\infty$, (binary) positive decimals,... are TOPS not SDC

Theorem

$\mathbb{K}$ topological, ordered, positive, star-domain downward closed
A $\mathbb{K}$-automaton is valid if, and only if,
the state-by-state ε-removal algorithm succeeds

Deciding validity

Definition

If $\mathcal{A}$ is a $\mathbb{Q}$- or $\mathbb{R}$-automaton,

then $\mathsf{abs}(\mathcal{A})$ is a $\mathbb{Q}_+$- or $\mathbb{R}_+$-automaton

Theorem

A $\mathbb{Q}$- or $\mathbb{R}$-automaton $\mathcal{A}$ is valid if and only if $\mathsf{abs}(\mathcal{A})$ is valid.

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings
- Links with the 'axiomatic' approach (Bloom–Ésik–Kuich) :
  Th: A starable star-strong semiring is an iteration semiring.
- References to previous work (on removal algorithm):
  - *locally closed* srgs (Ésik–Kuich), *k-closed* srgs (Mohri)
  - links with other algorithms:
    shortest-distance algorithm (Mohri),
    state-elimination method (Hanneforth–Higueira)

Conclusion

- Semiring structure is weak, topology does not help so much.
- This weakness imposes a restricted definition of validity,
  in order to guarantee success of validity algorithms.
- Axiomatic approach does not allow
  to deal wit most common numerical semirings: $\mathbb{Z}$min, $\mathbb{Q}$
- On 'usual' semirings,
  the new definition of validity coincides with the former one.

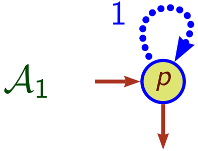
Conclusion (2)

- Apart the trivial cases, and the TOPS SCD case,
  decision of validity is never granted, and has to be established.
- On ‘usual’ semirings, validity is decidable.
- The new definition of validity
  fills the ‘effectivity gap’ left open by the former one.

Hidden parts

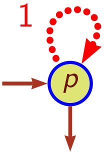
- The problematic examples

Problems in computing the behaviour of a weighted automaton



Problems in computing the behaviour of a weighted automaton

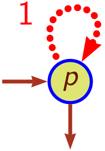
$\mathcal{A}_1$



$(1)^* =$ undefined

$\mathbb{N}$ natural integers $|\mathcal{A}_1|$ not defined

Problems in computing the behaviour of a weighted automaton

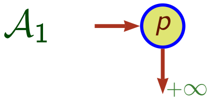


$\mathcal{A}_1$

$(1)^* = +\infty$

$\mathbb{N}$		natural integers	$\lvert\mathcal{A}_1\rvert$	not defined
$\mathcal{N}$	$\mathbb{N} \cup +\infty$	compact topology	$\lvert\mathcal{A}_1\rvert$	defined

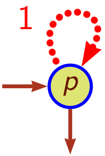
Problems in computing the behaviour of a weighted automaton



$\mathbb{N}$		natural integers	$\lvert\mathcal{A}_1\rvert$	not defined
$\mathcal{N}$	$\mathbb{N} \cup +\infty$	compact topology	$\lvert\mathcal{A}_1\rvert$	defined

Problems in computing the behaviour of a weighted automaton

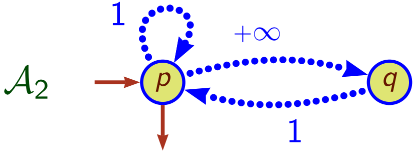
$\mathcal{A}_1$



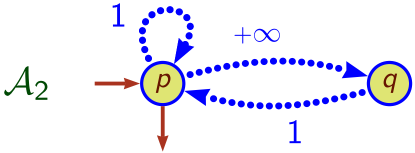
$(1)^* = \text{undefined}$

$\mathbb{N}$		natural integers	$\lvert\mathcal{A}_1\rvert$	not defined
$\mathcal{N}$	$\mathbb{N} \cup +\infty$	compact topology	$\lvert\mathcal{A}_1\rvert$	defined
$\mathbb{N}_\infty$	$\mathbb{N} \cup +\infty$	discrete topology	$\lvert\mathcal{A}_1\rvert$	not defined

Problems in computing the behaviour of a weighted automaton

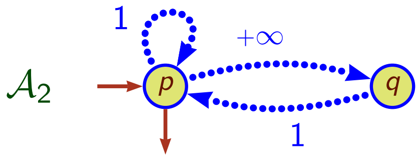


Problems in computing the behaviour of a weighted automaton



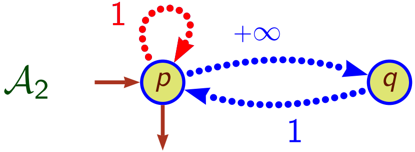
$\mathcal{N}$ $\mathbb{N} \cup +\infty$ compact topology $|\mathcal{A}_1|$ defined

Problems in computing the behaviour of a weighted automaton



$\mathcal{N}$	$\mathbb{N} \cup +\infty$	compact topology	$\lvert\mathcal{A}_1\rvert$	defined
$\mathbb{N}_\infty$	$\mathbb{N} \cup +\infty$	discrete topology	$\lvert\mathcal{A}_1\rvert$	defined

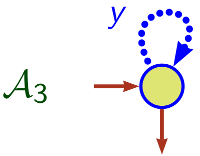
Problems in computing the behaviour of a weighted automaton



$(1)^* = \text{undefined}$

$\mathcal{N}$	$\mathbb{N} \cup +\infty$	compact topology	$\lvert\mathcal{A}_1\rvert$	defined
$\mathbb{N}_\infty$	$\mathbb{N} \cup +\infty$	discrete topology	$\lvert\mathcal{A}_1\rvert$	defined

Problems in computing the behaviour of a weighted automaton



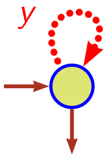
$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\qquad x = y^2 \qquad x$ not starable

Problems in computing the behaviour of a weighted automaton

$\mathcal{A}_3$



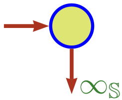
$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\qquad x = y^2 \qquad x$ not starable

Problems in computing the behaviour of a weighted automaton

$\mathcal{A}_3$

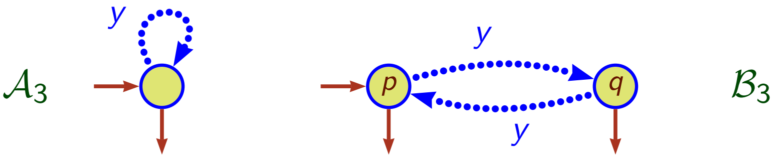


$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\qquad x = y^2 \qquad x$ not starable

Problems in computing the behaviour of a weighted automaton

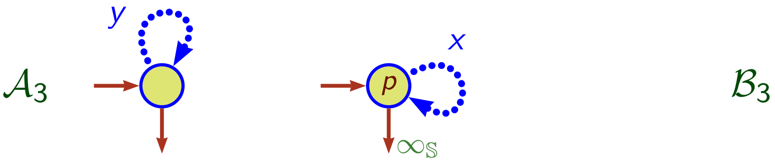


$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\quad$ $x = y^2$ $\quad$ x not starable

Problems in computing the behaviour of a weighted automaton



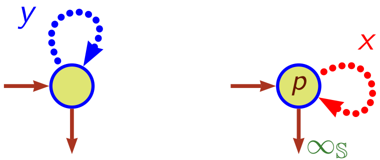
$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\quad$ $x = y^2$ $\quad$ x not starable

Problems in computing the behaviour of a weighted automaton

$\mathcal{A}_3$



$\mathcal{B}_3$

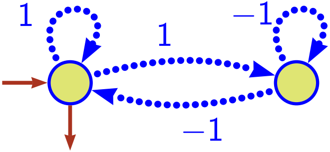
$$\mathbb{S} \subset \mathbb{N}^{2\times 2}, \quad x = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = 1_{\mathbb{S}}, \quad y = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad x+y = \infty_{\mathbb{S}} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$$

$\mathbb{S}$ equipped with the discrete topology

$0_{\mathbb{S}}$, y, and $\infty_{\mathbb{S}}$ starable $\qquad x = y^2 \qquad x$ not starable

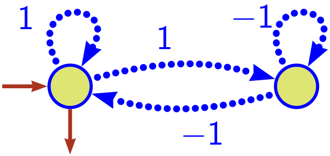
Problems in computing the behaviour of a weighted automaton

$\mathcal{A}_4$



Problems in computing the behaviour of a weighted automaton

$\mathcal{A}_4$

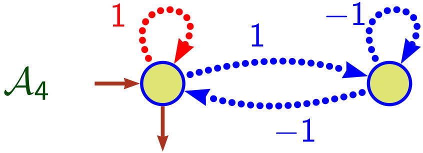


$$\mathcal{A}_4 = \langle\, I_4, \underline{E_4}, T_4 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\rangle$$

$$|\mathcal{A}_4| = I_4 \cdot \underline{E_4}^* \cdot T_4$$

$$\underline{E_4}^2 = 0 \quad \Longrightarrow \quad \underline{E_4}^* = \begin{pmatrix} 2 & 1 \\ -1 & 0 \end{pmatrix} \quad \Longrightarrow \quad |\mathcal{A}_4| = 2$$

Problems in computing the behaviour of a weighted automaton

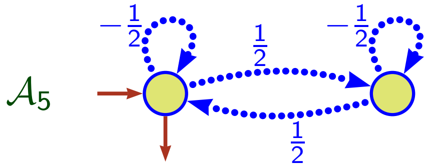


$$\mathcal{A}_4 = \langle\, I_4, \underline{E_4}, T_4 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\rangle$$

$$|\mathcal{A}_4| = I_4 \cdot \underline{E_4}^* \cdot T_4$$

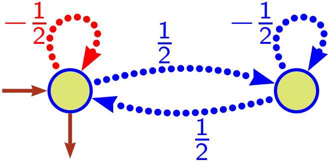
$$\underline{E_4}^2 = 0 \quad \Longrightarrow \quad \underline{E_4}^* = \begin{pmatrix} 2 & 1 \\ -1 & 0 \end{pmatrix} \quad \Longrightarrow \quad |\mathcal{A}_4| = 2$$

Problems in computing the behaviour



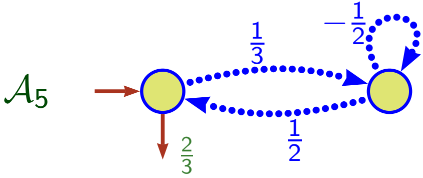
Problems in computing the behaviour

$\mathcal{A}_5$

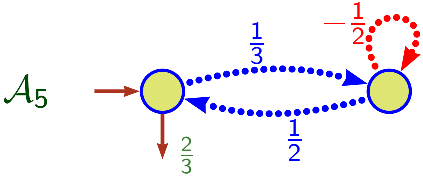


$$\left(-\frac{1}{2}\right)^* = \frac{2}{3}$$

Problems in computing the behaviour

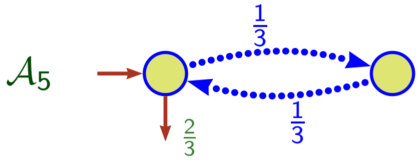


Problems in computing the behaviour



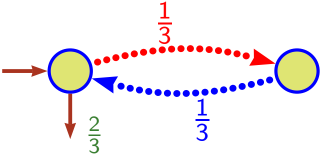
$$\left(-\frac{1}{2}\right)^* = \frac{2}{3}$$

Problems in computing the behaviour

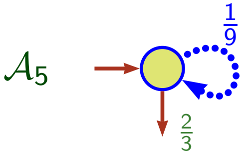


Problems in computing the behaviour

$\mathcal{A}_5$

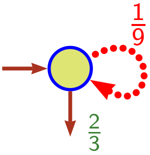


Problems in computing the behaviour



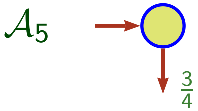
Problems in computing the behaviour

$\mathcal{A}_5$

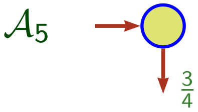


$$\left(\frac{1}{9}\right)^* = \frac{9}{8}$$

Problems in computing the behaviour



Problems in computing the behaviour



$$\mathcal{A}_5 = \langle\, I_5, \underline{E_5}, T_5 \,\rangle = \left\langle \begin{pmatrix} 1 & 0 \end{pmatrix}, \begin{pmatrix} -\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\rangle$$

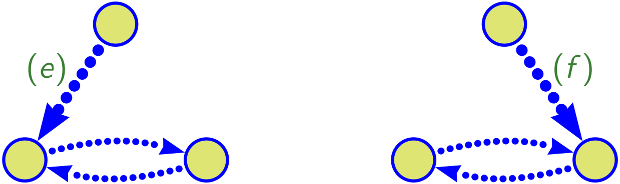
$$|\mathcal{A}_5| = I_5 \cdot \underline{E_5}^* \cdot T_5$$

$$\underline{E_5}^3 = \underline{E_5} \quad \Longrightarrow \quad \underline{E_5}^* \quad \text{undefined} \quad \Longrightarrow \quad |\mathcal{A}_5| \quad \text{undefined}$$

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues

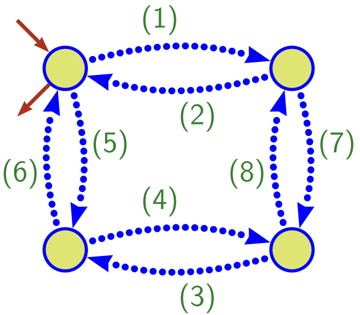
Termination issues



weighted ε-removal procedure does not terminate

if newly created ε-transitions are stored in a stack

Termination issues



weighted ε-removal procedure does not terminate

if newly created ε-transitions are stored in a queue

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity

Automata and expressions validity

'Kleene' theorem

Automata	$\Longleftrightarrow$	Expressions
$\mathcal{A}$	$\Longleftrightarrow$	E
Weighted automata	$\Longleftrightarrow$	Weighted expressions

Automata and expressions validity

'Kleene' theorem

Automata	$\Longleftrightarrow$	Expressions
$\mathcal{A}$	$\Longleftrightarrow$	E
Weighted automata	$\Longleftrightarrow$	Weighted expressions

Notion of a valid expression

E *valid* $\Longleftrightarrow$ c(E) well-defined

c(E) computed by a bottom-up traversal of the syntactic tree of E

Automata and expressions validity

Valid $\mathcal{A}$ yields valid E

Valid E yields valid $\mathcal{A}$ with Glushkov construction

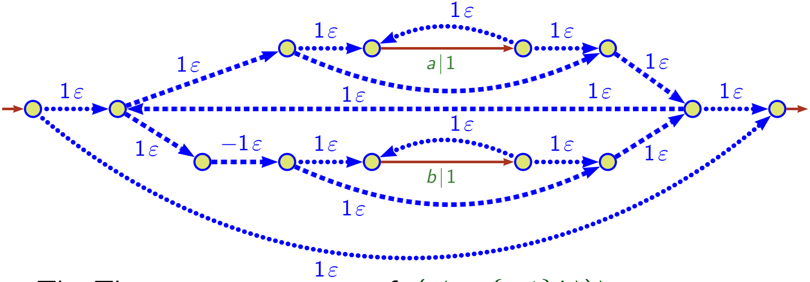
Valid E may yield non valid $\mathcal{A}$ with Thompson construction

Automata and expressions validity

Valid $\mathcal{A}$ yields valid E

Valid E yields valid $\mathcal{A}$ with Glushkov construction

Valid E may yield non valid $\mathcal{A}$ with Thompson construction



The Thompson automaton of $(a^* + \{-1\}b^*)^*$

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings

Definition

A topological semiring is a *strong* semiring
if the product of two summable families is a summable family

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- ‘Infinitary’ axioms : *strong*, *star-strong* semirings

Definition

A topological semiring is a *strong* semiring
if the product of two summable families is a summable family

Theorem

$\mathbb{K}$ *strong semiring* $s \in \mathbb{K}\langle\langle A^* \rangle\rangle$ *starable iff* $s_0 \in \mathbb{K}$ *starable*

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- ‘Infinitary’ axioms : *strong*, *star-strong* semirings

Definition

A topological semiring is a *strong* semiring
if the product of two summable families is a summable family

Definition

A topological semiring is a *star-strong* semiring if
the star of a summable family, whose sum is starable, is summable

Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings
- Links with the 'axiomatic' approach (Bloom–Ésik–Kuich):

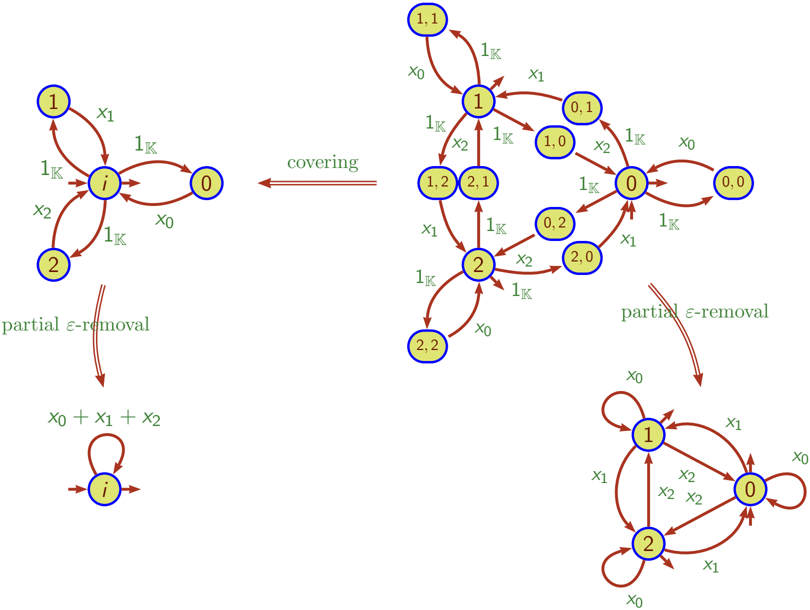
Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings
- Links with the 'axiomatic' approach (Bloom–Ésik–Kuich):

Theorem

A starable star-strong semiring is an iteration semiring

Group identities



Hidden parts

- The problematic examples
- The removal algorithm itself:
  - Termination issues (weighted versus Boolean cases)
  - Complexity issues
- Automata and expressions validity
- 'Infinitary' axioms : *strong*, *star-strong* semirings
- Links with the 'axiomatic' approach (Bloom–Ésik–Kuich):
- References to previous work (on removal algorithm):
  - *locally closed* srgs (Ésik–Kuich), *k-closed* srgs (Mohri)
  - links with other algorithms:
    shortest-distance algorithm (Mohri),
    state-elimination method (Hanneforth–Higueira)