# Equivalence of regular expressions with converse on relations

An alternative presentation of the proof by **Bloom**, **Ésik** and **Stefanescu**

Paul Brunet and Damien Pous

ENS de Lyon

17 juin 2013

# Converse

## Converse on languages : Mirror

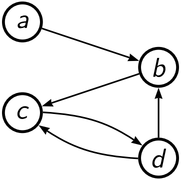
$$1^{\vee} := 1$$

$$(x \cdot w)^{\vee} := w^{\vee} \cdot x$$

$$L^{\vee} := \{w^{\vee} \mid w \in L\}$$

## Converse on relations

$$R^{\vee} := \{(y, x) \mid (x, y) \in R\}$$



# Converse

## Converse on languages : Mirror

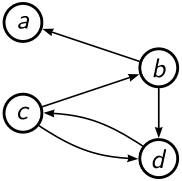
$$1^\vee := 1$$

$$(x \cdot w)^\vee := w^\vee \cdot x$$

$$L^\vee := \{w^\vee \mid w \in L\}$$

## Converse on relations

$$R^\vee := \{(y, x) \mid (x, y) \in R\}$$



# Plan

1. Kleene Algebrae with converse
2. Construction of the closure of an automaton
3. On examples

# Table of Contents

# Equivalence

We'll use different notions of equivalence on expressions $e, f$ on an alphabet $X$ :

We write $[\![e]\!]$ for the language denoted by a regular expression $e$.

# Equivalence

We'll use different notions of equivalence on expressions $e, f$ on an alphabet $X$ :

Language equality : $[\![e]\!] = [\![f]\!]$ ;

We write $[\![e]\!]$ for the language denoted by a regular expression $e$.

# Equivalence

We'll use different notions of equivalence on expressions $e, f$ on an alphabet $X$ :

Language equality : $[\![e]\!] = [\![f]\!]$ ;
Equivalence for language models : $\forall \Sigma, \forall \sigma \in {\Sigma^*}^X, \hat{\sigma}(e) = \hat{\sigma}(f)$ : $\boxed{e \equiv_{Lang} f}$ ;

We write $[\![e]\!]$ for the language denoted by a regular expression $e$.

# Equivalence

We'll use different notions of equivalence on expressions $e, f$ on an alphabet $X$ :

Language equality : $[\![e]\!] = [\![f]\!]$ ;
Equivalence for language models : $\forall \Sigma, \forall \sigma \in {\Sigma^*}^X, \hat{\sigma}(e) = \hat{\sigma}(f)$ : $\boxed{e \equiv_{Lang} f}$ ;
Equivalence for relation models : $\forall S, \forall \sigma \in \mathscr{P}\left(S^2\right)^X, \hat{\sigma}(e) = \hat{\sigma}(f)$ : $\boxed{e \equiv_{Rel} f}$.

We write $[\![e]\!]$ for the language denoted by a regular expression $e$.

# Kleene Algebrae I

A Kleene Algebra [(i)] is an algebraic structure $\langle K,+,\cdot,^*,0,1\rangle$ satifying :

1. $\langle K,+,\cdot,0,1\rangle$ is an idempotent semiring :

$\langle K,+,0\rangle$ is a commutative idempotent monoid
$$\begin{cases} a+(b+c)=(a+b)+c \\ a+b=b+a \\ a+0=a \\ a+a=a \end{cases}$$

$\langle K,\cdot,1\rangle$ is a monoid
$$\begin{cases} a(bc)=(ab)c \\ 1a=a \\ a1=a \end{cases}$$

Distributivity laws
$$\begin{cases} a(b+c)=ab+ac \\ (a+b)c=ac+bc \\ 0a=0 \\ a0=0 \end{cases}$$

# Kleene Algebrae II

2. The $^*$ operation satisfy :

$$
\begin{aligned}
&1 + aa^* \leqslant a^* \\
&1 + a^*a \leqslant a^* \\
&b + ax \leqslant x \Rightarrow a^*b \leqslant x \\
&b + xa \leqslant x \Rightarrow ba^* \leqslant x
\end{aligned}
$$

Where $a \leqslant b \stackrel{\Delta}{\Leftrightarrow} a + b = b$.

The last axioms can be replaced by a number of things.

(i). As presented in [Koz94].

# All is well

$$[\![e]\!] = [\![f]\!]$$

# All is well

$$[\![e]\!] = [\![f]\!] \quad \Leftrightarrow \quad KA \vdash e = f$$

# All is well

$$[\![e]\!] = [\![f]\!] \quad \Leftrightarrow \quad KA \vdash e = f \quad \Leftrightarrow \quad e \equiv_{Lang} f$$

# All is well

$$[\![e]\!] = [\![f]\!] \quad \Leftrightarrow \quad KA \vdash e = f \quad \Leftrightarrow \quad e \equiv_{Lang} f \quad \Leftrightarrow \quad e \equiv_{Rel} f$$

# Kleene Algebra with Converse

**Theorem [BES95]**

A complete axiomatization of the variety $L^\vee$ generated by regular languages with converse consists of the axioms for KA and the following :

- $(a+b)^\vee = a^\vee + b^\vee$
- $(a \cdot b)^\vee = b^\vee \cdot a^\vee$
- $(a^*)^\vee = (a^\vee)^*$
- $a^{\vee\vee} = a.$

# Equivalence in $L^\vee$

Let $e, f \in Reg^\vee(X)$.

# Equivalence in $L^{\vee}$

Let $e, f \in Reg^{\vee}(X)$.
We compute $\tau(e), \tau(f) \in Reg(X \cup X')$.
Where

| | |
|---|---|
| $\forall x \in X, \tau(x) = x$ | $\forall x \in X, \nu(x) = x'$ |
| $\tau(\mathbb{1}) = \mathbb{1}$ | $\nu(\mathbb{1}) = \mathbb{1}$ |
| $\tau(e_1 \cdot e_2) = \tau(e_1) \cdot \tau(e_2)$ | $\nu(e_1 \cdot e_2) = \nu(e_2) \cdot \nu(e_1)$ |
| $\tau(e_1 + e_2) = \tau(e_1) + \tau(e_2)$ | $\nu(e_1 + e_2) = \nu(e_1) + \nu(e_2)$ |
| $\tau(e^*) = \tau(e)^*$ | $\nu(e^*) = \nu(e)^*$ |
| $\tau(e^{\vee}) = \nu(e)$ | $\nu(e^{\vee}) = \tau(e)$ |

and

$X' := \{x' \mid x \in X\}$ is a disjoint copy of $X$.

# Equivalence in $L^\vee$

Let $e, f \in Reg^\vee(X)$.
We compute $\tau(e), \tau(f) \in Reg(X \cup X')$.
Where

| | |
|---|---|
| $\forall x \in X, \tau(x) = x$ | $\forall x \in X, \nu(x) = x'$ |
| $\tau(\mathbb{1}) = \mathbb{1}$ | $\nu(\mathbb{1}) = \mathbb{1}$ |
| $\tau(e_1 \cdot e_2) = \tau(e_1) \cdot \tau(e_2)$ | $\nu(e_1 \cdot e_2) = \nu(e_2) \cdot \nu(e_1)$ |
| $\tau(e_1 + e_2) = \tau(e_1) + \tau(e_2)$ | $\nu(e_1 + e_2) = \nu(e_1) + \nu(e_2)$ |
| $\tau(e^*) = \tau(e)^*$ | $\nu(e^*) = \nu(e)^*$ |
| $\tau(e^\vee) = \nu(e)$ | $\nu(e^\vee) = \tau(e)$ |

and

$X' := \{x' \mid x \in X\}$ is a disjoint copy of $X$.
Furthermore

$$\boxed{[\![\tau(e)]\!] = [\![\tau(f)]\!] \Leftrightarrow e \equiv_{Lang} f}$$

# Languages *vs.* Relations

$$a \leqslant aa^{\vee}a \text{ ???}$$

| in $L^{\vee}$ | in $REL^{\vee}$ |
|---|---|
| $a \not\subseteq aa^{\vee}a = aaa$ | if $xRy$ then $xRyR^{\vee}xRy$ |
| | which means $xRR^{\vee}Ry$ |
| | so $R \subseteq RR^{\vee}R$ |

# Languages *vs.* Relations

$$a \leqslant aa^{\vee}a \text{ ???}$$

| in $L^{\vee}$ | in $REL^{\vee}$ |
|---|---|
| $a \not\subseteq aa^{\vee}a = aaa$ | if $xRy$ then $xRyR^{\vee}xRy$<br>which means $xRR^{\vee}Ry$<br>so $R \subseteq RR^{\vee}R$ |

**Theorem [EB95]**

A complete set of axioms for the variety $REL^{\vee}$ generated by regular relations with converse consists on the axioms for $L^{\vee}$ and the axiom $a \leqslant aa^{\vee}a$.

# Equivalence in $REL^{\vee}$

**Theorem [BES95]**

$$cl([\![\tau(e)]\!]) = cl([\![\tau(f)]\!]) \Leftrightarrow e \equiv_{Rel} f$$

# Equivalence in $REL^{\vee}$

## Theorem [BES95]

$$cl(\llbracket\tau(e)\rrbracket) = cl(\llbracket\tau(f)\rrbracket) \Leftrightarrow e \equiv_{Rel} f$$

Let $\bar{\cdot}$ be the function $\begin{cases} (X \cup X')^* & \rightarrow & (X \cup X')^* \\ \epsilon & \mapsto & \epsilon \\ xw & \mapsto & \overline{w}x' \\ x'w & \mapsto & \overline{w}x \end{cases}$

# Equivalence in $REL^{\vee}$

## Theorem [BES95]

$$cl([\![\tau(e)]\!]) = cl([\![\tau(f)]\!]) \Leftrightarrow e \equiv_{Rel} f$$

Let $\bar{\cdot}$ be the function $\begin{cases} (X \cup X')^* & \rightarrow & (X \cup X')^* \\ \epsilon & \mapsto & \epsilon \\ xw & \mapsto & \overline{w}x' \\ x'w & \mapsto & \overline{w}x \end{cases}$

We define the relation $\rightsquigarrow$ on $(X \cup X')^*$ :

$$u \rightsquigarrow v \stackrel{\Delta}{\Leftrightarrow} \exists u_1, w, u_2 : u = u_1 w \overline{w} w u_2 \wedge v = u_1 w u_2.$$

# Equivalence in $REL^{\vee}$

**Theorem [BES95]**

$$cl(\llbracket\tau(e)\rrbracket) = cl(\llbracket\tau(f)\rrbracket) \Leftrightarrow e \equiv_{Rel} f$$

Let $\bar{.}$ be the function $\left\{ \begin{array}{rcl} (X \cup X')^* & \to & (X \cup X')^* \\ \epsilon & \mapsto & \epsilon \\ xw & \mapsto & \overline{w}x' \\ x'w & \mapsto & \overline{w}x \end{array} \right.$

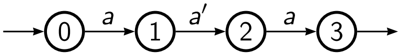
We define the relation $\rightsquigarrow$ on $(X \cup X')^*$ :

$$u \rightsquigarrow v \stackrel{\Delta}{\Leftrightarrow} \exists u_1, w, u_2 : u = u_1 w \overline{w} w u_2 \wedge v = u_1 w u_2.$$

$cl(A)$ is the closure of $A$ for $\rightsquigarrow$ : $cl(A) = \{v \mid \exists u \in A : u \rightsquigarrow^* v\}$.

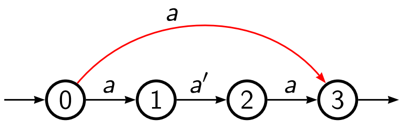
# Table of Contents

# Closure of an automaton

- 

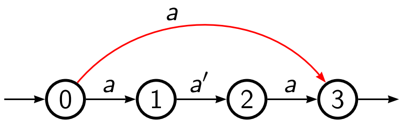
- 

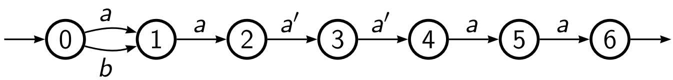
# Closure of an automaton

- 

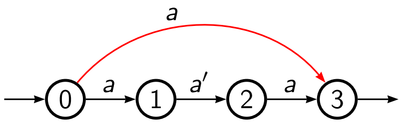
- 

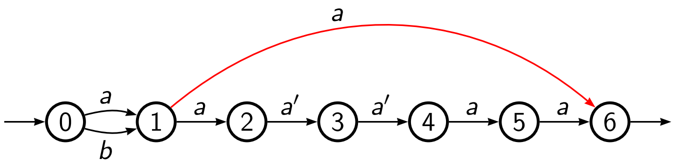
# Closure of an automaton

- 

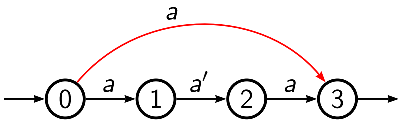

- 

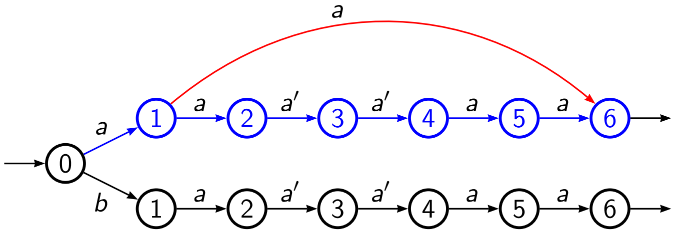
# Closure of an automaton

- 


- 

# Closure of an automaton

- 


- 


And it gets worse...

# Idea of the construction

If $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{\overline{wx}wx} q_3$ ,

*i.e.* pattern $wx$

# Idea of the construction

If $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{\overline{wx}wx} q_3$ ,

*i.e.* pattern $wx$

then : $q_0, [\epsilon] \xrightarrow{uw} q_1, [uw] \xrightarrow{x} q_3, [uwx]$

# Idea of the construction

If $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{\overline{wx}wx} q_3$ ,

*i.e.* pattern $wx$

then : $q_0, [\epsilon] \xrightarrow{uw} q_1, [uw] \xrightarrow{x} q_3, [uwx]$

Furthermore, if $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{v} q_3$ ,

such that $v \Rightarrow^* x'\overline{w}wx$,

# Idea of the construction

If $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{\overline{wx}wx} q_3$ ,

*i.e.* pattern $wx$

then : $q_0, [\epsilon] \xrightarrow{uw} q_1, [uw] \xrightarrow{x} q_3, [uwx]$

Furthermore, if $q_0 \xrightarrow{uw} q_1 \xrightarrow{x} q_2 \xrightarrow{v} q_3$ ,

such that $v \Rightarrow^* x'\overline{w}wx$,

then : $q_0, [\epsilon] \xrightarrow{uw} q_1, [uw] \xrightarrow{x} q_3, [uwx]$

# Construction : Γ

We write $\bar{X} := X \cup X'$.
For $w \in \bar{X}^*$, consider the language defined inductively :

$$\begin{array}{rrcl} \Gamma : & \bar{X}^* & \longrightarrow & \mathscr{P}(\bar{X}^*) \\ & \epsilon & \longmapsto & \epsilon \\ & wx & \longmapsto & (x'\Gamma(w)x)^* \end{array}$$

## Properties of $\Gamma(w)$

# Construction : Γ

We write $\bar{X} := X \cup X'$.
For $w \in \bar{X}^*$, consider the language defined inductively :

$$\begin{array}{rccl} \Gamma : & \bar{X}^* & \longrightarrow & \mathscr{P}(\bar{X}^*) \\ & \epsilon & \longmapsto & \epsilon \\ & wx & \longmapsto & \left(x'\Gamma(w)x\right)^* \end{array}$$

## Properties of $\Gamma(w)$

1. $\Gamma(w)$ is *upward* $\rightsquigarrow$*-closed* : $u \rightsquigarrow v \in \Gamma(w) \Rightarrow u \in \Gamma(w)$ ;

# Construction : $\Gamma$

We write $\bar{X} \coloneqq X \cup X'$.
For $w \in \bar{X}^*$, consider the language defined inductively :

$$\begin{array}{rccl} \Gamma : & \bar{X}^* & \longrightarrow & \mathscr{P}(\bar{X}^*) \\ & \epsilon & \longmapsto & \epsilon \\ & wx & \longmapsto & (x'\Gamma(w)x)^* \end{array}$$

## Properties of $\Gamma(w)$

1. $\Gamma(w)$ is *upward* $\rightsquigarrow$*-closed* : $u \rightsquigarrow v \in \Gamma(w) \Rightarrow u \in \Gamma(w)$ ;
2. $u \in \Gamma(w) \Rightarrow wu \rightsquigarrow^* w$ ;

# Construction : Γ

We write $\bar{X} := X \cup X'$.
For $w \in \bar{X}^*$, consider the language defined inductively :

$$\begin{array}{rccl} \Gamma : & \bar{X}^* & \longrightarrow & \mathscr{P}(\bar{X}^*) \\ & \epsilon & \longmapsto & \epsilon \\ & wx & \longmapsto & (x'\Gamma(w)x)^* \end{array}$$

## Properties of $\Gamma(w)$

1. $\Gamma(w)$ is *upward* $\rightsquigarrow$*-closed* : $u \rightsquigarrow v \in \Gamma(w) \Rightarrow u \in \Gamma(w)$ ;
2. $u \in \Gamma(w) \Rightarrow wu \rightsquigarrow^* w$ ;
3. $\exists v$ suffix of $w$ : $u \rightsquigarrow^* \overline{v}v \Rightarrow u \in \Gamma(w)$ ;

# Construction : $\Gamma$

We write $\bar{X} := X \cup X'$.
For $w \in \bar{X}^*$, consider the language defined inductively :

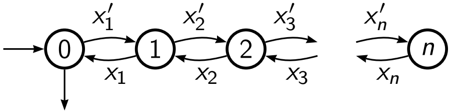
$$\begin{array}{rccl} \Gamma : & \bar{X}^* & \longrightarrow & \mathscr{P}(\bar{X}^*) \\ & \epsilon & \longmapsto & \epsilon \\ & wx & \longmapsto & (x'\Gamma(w)x)^* \end{array}$$

## Properties of $\Gamma(w)$

1. $\Gamma(w)$ is *upward $\rightsquigarrow$-closed* : $u \rightsquigarrow v \in \Gamma(w) \Rightarrow u \in \Gamma(w)$ ;
2. $u \in \Gamma(w) \Rightarrow wu \rightsquigarrow^* w$ ;
3. $\exists v$ suffix of $w$ : $u \rightsquigarrow^* \overline{v}v \Rightarrow u \in \Gamma(w)$ ;
4. subsequently : $\Gamma(w) = cl^{\uparrow}\left(\{\overline{v}v \mid v \text{ suffix of } w\}\right)$

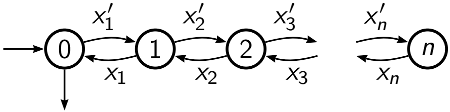
# $\Gamma(w)$ is upward-closed

$\Gamma(x_n \cdots x_1)$ is recognized by :



# $\Gamma(w)$ is upward-closed

$\Gamma(x_n \cdots x_1)$ is recognized by :

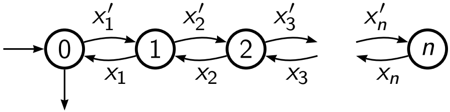



## Property

In this automaton, if $q_1 \xrightarrow{x} q_2$, then $q_2 \xrightarrow{x'} q_1$.

# $\Gamma(w)$ is upward-closed

$\Gamma(x_n \cdots x_1)$ is recognized by :




## Property

In this automaton, if $q_1 \xrightarrow{x} q_2$, then $q_2 \xrightarrow{x'} q_1$.

## Corollary

If $0 \xrightarrow{u_1} q_1 \xrightarrow{w} q_2 \xrightarrow{u_2} 0$ , then

$$0 \xrightarrow{u_1} q_1 \xrightarrow{w} q_2 \xrightarrow{\overline{w}} q_1 \xrightarrow{w} q_2 \xrightarrow{u_2} 0 .$$

# Construction : $\gamma$

## Notations

We consider an automaton $\mathscr{A} = \langle Q, \bar{X}, I, F, \Delta \rangle$,
and define $\forall x \in \bar{X},\ R_x := \{(q,q') \mid (q,x,q') \in \Delta\}$.

## Definition : $\gamma$

$$\begin{array}{rccc} \gamma : & \bar{X}^* & \longrightarrow & Rel_Q \\ & \epsilon & \longmapsto & Id_Q \\ & w \cdot x & \longmapsto & (R_{x'} \circ \gamma(w) \circ R_x)^* \end{array}$$

# Construction : $\gamma$

## Notations

We consider an automaton $\mathscr{A} = \langle Q, \bar{X}, I, F, \Delta \rangle$,
and define $\forall x \in \bar{X},\ R_x := \{(q, q') \mid (q, x, q') \in \Delta\}$.

## Definition : $\gamma$

$$\begin{array}{rccc} \gamma : & \bar{X}^* & \longrightarrow & Rel_Q \\ & \epsilon & \longmapsto & Id_Q \\ & w \cdot x & \longmapsto & (R_{x'} \circ \gamma(w) \circ R_x)^* \end{array}$$

And $G = \bar{X}^*/\gamma = \{[w] \mid w \in \bar{X}^*\} = \{\{u \mid \gamma(u) = \gamma(w)\} \mid w \in \bar{X}^*\}$ (finite)

# Construction : $\gamma$

## Notations

We consider an automaton $\mathscr{A} = \langle Q, \bar{X}, I, F, \Delta \rangle$,
and define $\forall x \in \bar{X},\ R_x \coloneqq \{(q, q') \mid (q, x, q') \in \Delta\}$.

## Definition : $\gamma$

$$\gamma : \begin{array}{ccc} \bar{X}^* & \longrightarrow & Rel_Q \\ \epsilon & \longmapsto & Id_Q \\ w \cdot x & \longmapsto & (R_{x'} \circ \gamma(w) \circ R_x)^* \end{array}$$

And $G = \bar{X}^*/\gamma = \{[w] \mid w \in \bar{X}^*\} = \{\{u \mid \gamma(u) = \gamma(w)\} \mid w \in \bar{X}^*\}$ (finite)

## Property

$\gamma(w) = \hat{\sigma}(\Gamma(w))$ where $\sigma(x) = R_x$, which means

$$(q_1, q_2) \in \gamma(w) \Leftrightarrow (\exists u \in \Gamma(w) : q_1 \xrightarrow{u} q_2)$$

# Closure Automaton

## $cl(\mathscr{A})$

$$cl(\mathscr{A}) \coloneqq \langle Q \times G, \bar{X}, I \times \{\mathbb{1}\}, F \times G, \Delta' \rangle$$

where $\Delta' \coloneqq \{((q_1, [w]), x, (q_2, [wx])) \mid (q_1, q_2) \in R_x \circ \gamma(wx)\}$ Which is to say $(q_1, [w]) \xrightarrow{x} (q_2, [wx])$ if :

$$q_1 \xrightarrow{x} q_3 \text{ and } (q_3, q_2) \in \gamma(wx)$$

# Closure Automaton

## $cl(\mathscr{A})$

$$cl(\mathscr{A}) \coloneqq \langle Q \times G, \bar{X}, I \times \{\mathbb{1}\}, F \times G, \Delta' \rangle$$

where $\Delta' \coloneqq \{((q_1, [w]), x, (q_2, [wx])) \mid (q_1, q_2) \in R_x \circ \gamma(wx)\}$ Which is to say $(q_1, [w]) \xrightarrow{x} (q_2, [wx])$ if :

$$q_1 \xrightarrow{x} q_3 \text{ and } (q_3, q_2) \in \gamma(wx)$$

- The $\gamma$ function needs to (or at least should) be pre-computed, but the rest of the contruction can be done *on-the-fly*.

# Closure Automaton

## $cl(\mathscr{A})$

$$cl(\mathscr{A}) := \langle Q \times G, \bar{X}, I \times \{\mathbb{1}\}, F \times G, \Delta' \rangle$$

where $\Delta' := \{((q_1, [w]), x, (q_2, [wx])) \mid (q_1, q_2) \in R_x \circ \gamma(wx)\}$ Which is to say $(q_1, [w]) \xrightarrow{x} (q_2, [wx])$ if :

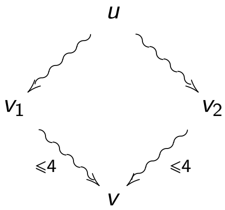
$$q_1 \xrightarrow{x} q_3 \text{ and } (q_3, q_2) \in \gamma(wx)$$

- The $\gamma$ function needs to (or at least should) be pre-computed, but the rest of the contruction can be done *on-the-fly*.
- One can build a *deterministic* closure automaton with set of states $\mathscr{P}(Q) \times G$ and transition *function* :

$$\delta(\{q_1 \dots q_n\}, [w]) := (\{p \mid \exists q_i : (q_i, p) \in R_x \circ \gamma(wx)\}, [wx]).$$

# To conclude

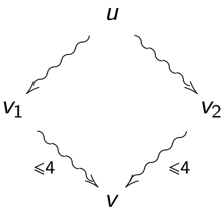
- $\rightsquigarrow$ is confluent. Furthermore,

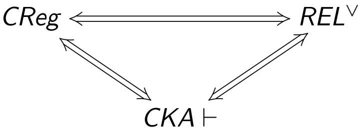



-

# To conclude

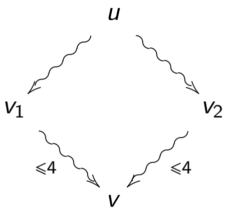
- $\rightsquigarrow$ is confluent. Furthermore,




- 

# To conclude

- $\rightsquigarrow$ is confluent. Furthermore,

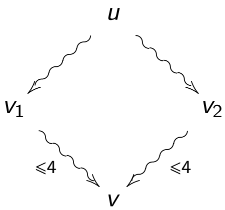



- $$CReg \Longrightarrow REL^{\vee}$$

$$CKA \vdash$$

# To conclude

- $\rightsquigarrow$ is confluent. Furthermore,

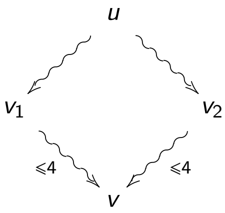



- $$CReg \Longleftrightarrow REL^{\vee}$$

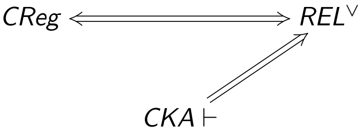
$$CKA \vdash$$

# To conclude

- $\rightsquigarrow$ is confluent. Furthermore,

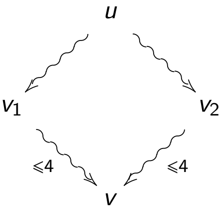


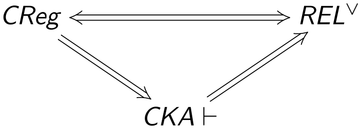

- 



# To conclude

- $\rightsquigarrow$ is confluent. Furthermore,




- 

# Table of Contents

# Bibliography


S.L. Bloom, Z. Ésik, and Gh. Stefanescu.
Notes on equational theories of relations.
*algebra universalis*, 33(1) :98–126, 1995.

Z. Ésik and L. Bernátsky.
Equational properties of kleene algebras of relations with conversion.
*Theor. Comput. Sci.*, 137(2) :237–251, January 1995.

Dexter Kozen.
A completeness theorem for kleene algebras and the algebra of regular events.
*Information and Computation*, 110 :366–390, 1994.