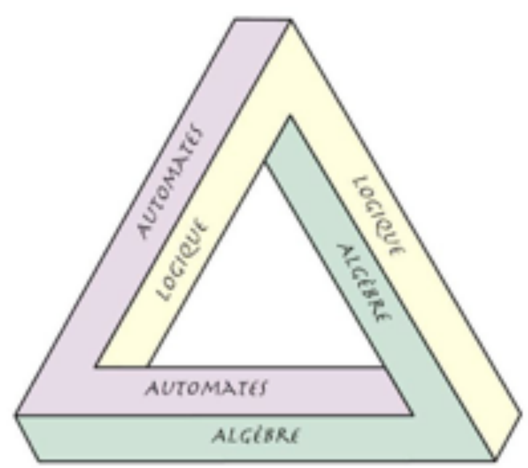
On monoids outputs

Thomas Colcombet
28 Mars 2018

ongoing thoughts with Adrien Boiret and Daniela Petrişan



Automata with outputs

Fix an input alphabet A and a monoid M.

finite or infinite

finite or infinite
e.g. free monoid
groupn, traces,
finite monoid

An automaton with outputs in M has:

finite or infinite

- a set of states p,q,r ∈ Q
- initial transitions of the form $\xrightarrow{x} q$
- transitions of the form $p \xrightarrow{a:x} q$
- final transitions of the form $p \xrightarrow{x}$

maps
Q×A → M×Q

partial maps

maps over second projection,
unique output (functionality)

relations, but unique output
(functionality)

An automaton with outputs in M computes a (partial) map F : A* → M

Examples: M = ({*}, · ,*), partial maps → deterministic automata
M = ({0,1},∧,1), maps → (more or less) deterministic automata
M = B*, partial maps → subsequential transducers

How it is related to Schützenberger's weighted automata?

Question: how can we minimize such automata?

Subsequential transducers

Subsequential transducers are automata with output in M = B* (the free monoid) and transitions are partial maps.

[Choffrut77] Subsequential transducers can be effectively/efficiently minimized.

Previous talk: the existence of this minimal automaton can all be phrased in categorical terms.

What does it mean to minimize ?

What do you mean by minimize ?

- minimize the number of states ?
    - It is always possible to minimize, but not always efficiently.
    - It provides no information on what it « means ».
- minimize algebraically ?
    - This requires to define what is a « sub automaton » and a « quotient automaton ». (factorization system in the previous talk).
    - There are examples that one wants to capture and that do not fall in an obvious manner in this case.
- give an algorithm of a certain form ?
    - using efficient basic steps (elimination, local modifications, merging),
    - greedy.

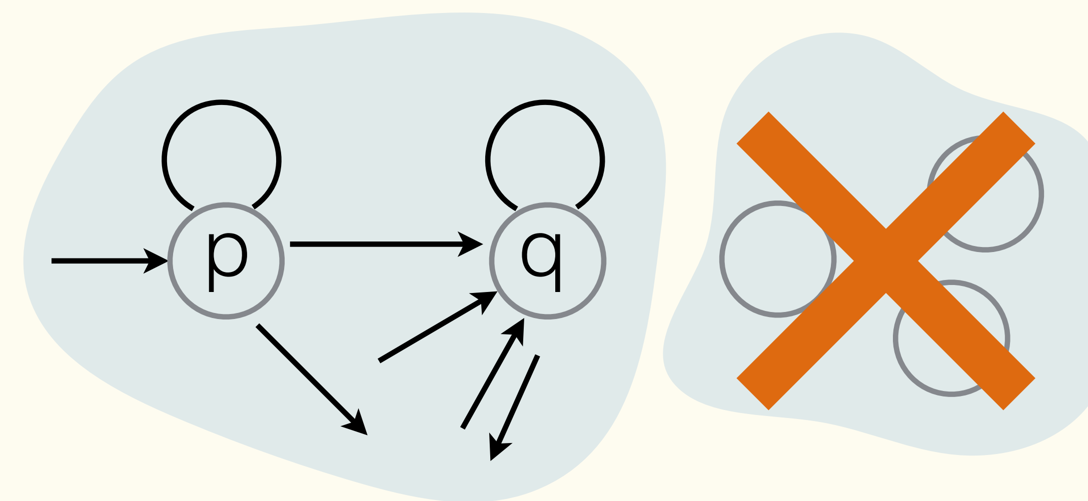
Our reference point:

Conclusion: the question is vaguely phrased.
And one does not want to make it clearer.

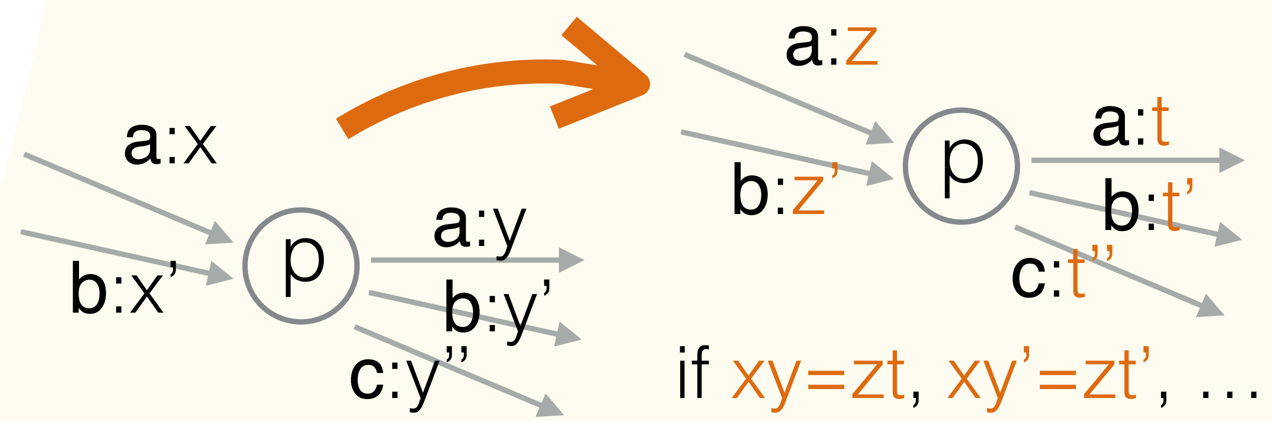
Subsequential transducers

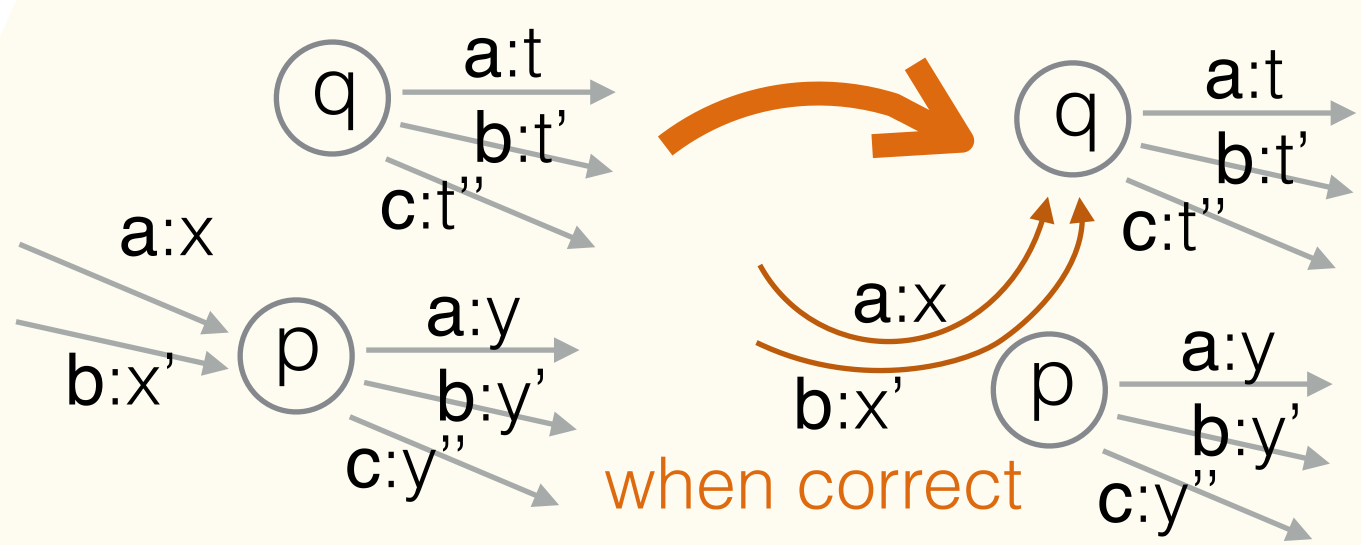
let's forget this for the moment

A subsequential transducer is
- a deterministic automaton, ~~partial,~~
- with output in B*



[Choffrut] Given a map F:A* → B* computed by a subsequential transducer,
- there exists a unique subsequential transducer that computes F, and that divides any subsequential transducer for F,
- is is unique up to isomorphism,
- it is possible to compute it efficiently,
- this algorithm can be seen as only:
  - removing unreachable states,
  - shifting weights,
  - redirecting transitions
  - never backtracks





First ingredients

A monoid $(M, \cdot, 1)$ may have the following properties:

LC. It is left cancellative: $ax=ay \rightarrow x=y$

RC. It is right cancellative: $xa=ya \rightarrow x=y$

WUP. every non-empty set has a « weakly universal prefix », i.e.

$\forall$ f:I $\rightarrow$ M with I nonempty
$\exists$ u$\in$M $\exists$ g: I $\rightarrow$ M such that $f = u \cdot g$
$\forall$ v$\in$M $\forall$ h:I $\rightarrow$ M such that $f = v \cdot h$
$h = w \cdot g$ for some w$\in$M

where
$(u \cdot g)(i) = u \cdot g(i)$
for all $i \in I$

Examples
- free monoids
- groups
- trace monoids

Co-examples
- all finite monoids that are not groups

[Choffrut++] Whenever a monoid has properties LC+RC+WUP, there exists a unique up to isomorphism automaton with outputs in M that « divides » any other automaton with outputs in M that recognizes the same map.

as in previous talk

(Effectiveness relies on the effectiveness of WUP computation).

Construction

Assume M has properties LC+RC+WUP:

LC. It is left cancellative: $ax=ay \rightarrow x=y$

RC. It is right cancellative: $xa=ya \rightarrow x=y$

WUP. every non-empty set has a « weakly universal prefix ».

Each state p of the automaton computes a map

$f_p : A^* \rightarrow M$

which is the map computed by setting p initial.

Minimization (non-effective) of an automaton with output in M computing F:

1. remove all states that are not reachable
2. define $p \sim q$ if there is g such that $f_p = u \cdot g$ and $f_q = v \cdot g$.
   Note that this is an equivalence relation (consequence of property WUP).
3. For all equivalence classes, choose a state p and a weak universal prefix g_C for f_p. (Rk: It is a weak universal prefix for all f_q, $q \in C$.
4. Shift the weights such that every state $q \in C$ computes the same g_C.
5. merge all states in the same equivalence class.

This automaton with outputs in M computes the same function.

It has the minimal number of states (LC property).

It divides any other automaton for the same language (RC property).

On right cancellation

RC. It is right cancellative: $xa=ya \rightarrow x=y$

Remark: If an automaton has its output in a monoid satisfying LC+WUP, then one can construct an automaton:
- which is minimal in number of states
- but which is not algebraically minimal.

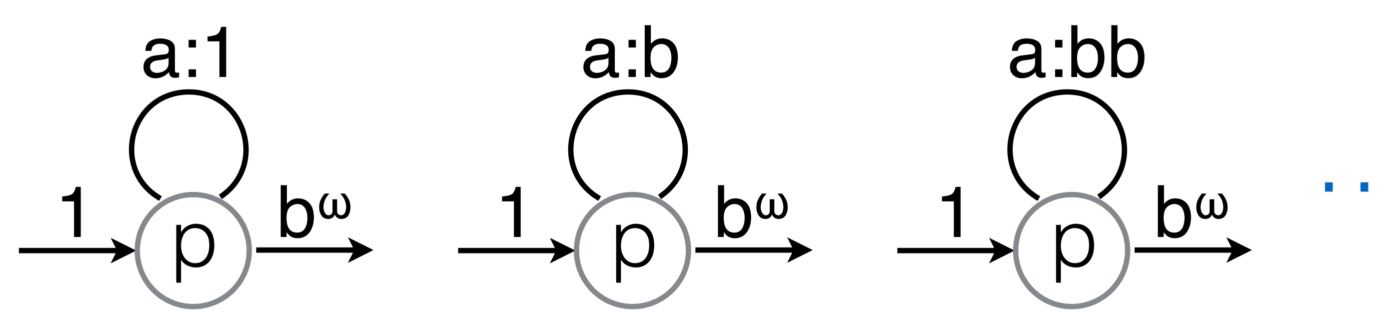
Example:
In finite monoids, left cancellative → group → right cancellative)

Let $A=\{a\}$ and M=(countable ordinal words over $B=\{b\}$,concatenation,1).

It satisfies LC+WUP, but not RC.

Let $f : u \longmapsto b^\omega$.

There are **infinitely many automata** that are incomparable under division, and in fact disconnected under the RST of morphisms.



All the structure of the « minimal automaton » is fixed, but there is no canonical choice of the output labels.

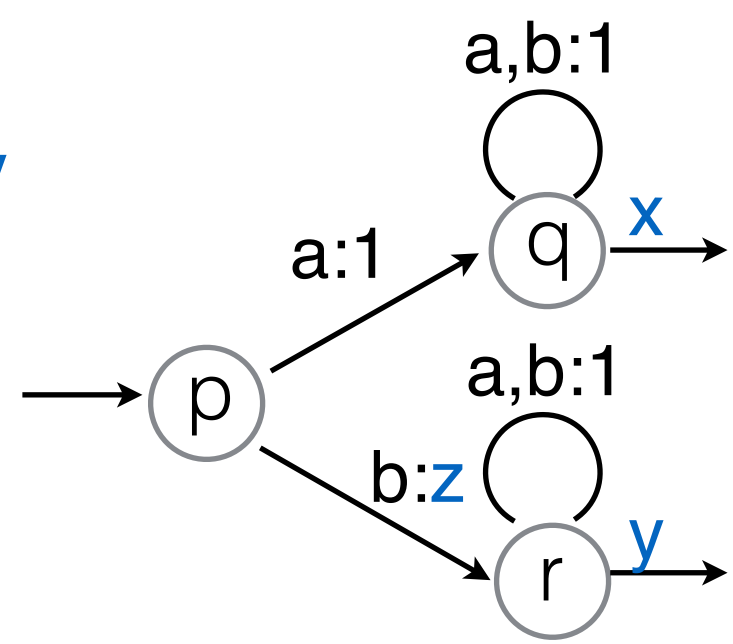
On left cancellation

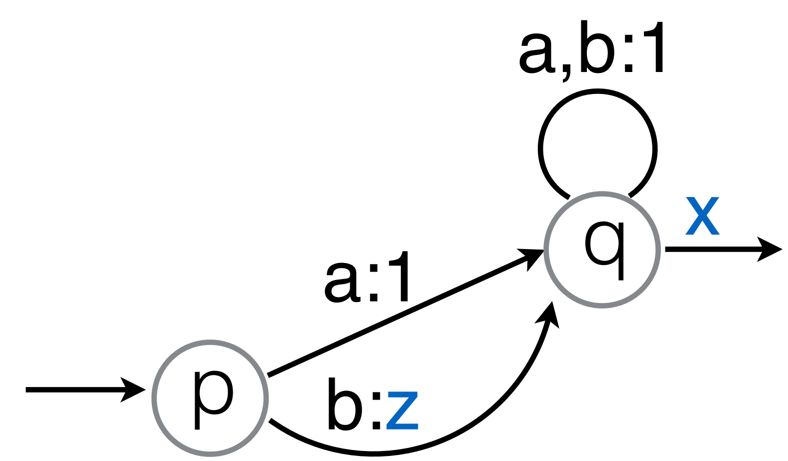
LC. It is left cancellative: $ax=ay \to x=y$

Not having left cancellation is much more dangerous because merging states depends on the context.

Assume ¬LC:
for some x,y,z, $x \neq y$ and $zx=zy$

q and r are not equivalent
(cannot be merged)





Max-monoid

Consider the monoid Max = ([1,n],max,1).

It satisfies WUP + ¬LC + ¬RC

Thm: There exists a polynomial algorithm which given an automaton with outputs in Max computes an equivalent automaton with outputs in Max that has the minimal number of states with this property.

However it is not unique: there is some choice to be made
- for the output labels
- for the transition targets

In particular ({0,1},∧,1) is of this form.

Continuations

Can we generalize the construction for Max to other cases.

A condition that seems interesting:

TOEC. Totally ordered equivalence congruences:

- the preorder $u \preccurlyeq v$ if $\forall x,y\ ux=uy \rightarrow vx=vy$ is total
- equivalently the equivalences $E_u = \{(x,y) : ux=uy\}$ are totally ordered under inclusion.

Remark: LC if and only if $\preccurlyeq$ is M^2
Hence TOEC is a weakening of LC.

Remark: adding $\bot$ to a monoid that is TOEC turns it into a monoid that is TOEC.

Conjecture: TOEC + WUP is sufficient for having a « good minimization algorithm ».

Conj-sequence: partiality can be added for free !

Question: How to formalize these with categories?
How to accommodate with algebraic divisibility ?

Thank you !

questions?