Varieties of cost functions.

Laure Daviaud[1], **Denis Kuperberg**[2], Jean-Éric Pin[3]

[1]ENS Lyon [2]TU Munich [3]LIAFA, Paris

STACS, Orléans
18-02-2015

Varieties of regular languages

A variety of regular languages is a class closed under:

- Union, intersection, complement,
- Left and right quotients ($u^{-1}L$),
- Inverses of morphisms between free monoids.

A variety of finite monoids is a class closed under:

- Finite products,
- Sub-monoids,
- Quotients.

Theorem (Eilenberg '76)

Varieties of regular languages are in bijection with varieties of finite monoids.

Equations

Example: The variety of star-free languages corresponds to aperiodic monoids.
It is characterized by the equation $x^\omega = x^\omega x$,
where ω is the *idempotent power*.

This allows one to decide membership in this variety.

Question: Is there a set of equations for each variety ?

Equations

Example: The variety of star-free languages corresponds to aperiodic monoids.
It is characterized by the equation $x^\omega = x^\omega x$,
where ω is the *idempotent power*.

This allows one to decide membership in this variety.

Question: Is there a set of equations for each variety ?

Theorem (Reiterman '82)

Yes, but the domain of these equations is the set of profinite words.

Profinite words

Distance between words u and v in A^*:

$$d(u,v) = \max\{\frac{1}{2^n} \mid \exists \text{ automaton of size } n \text{ separating } u \text{ and } v\}.$$

Examples:
$d(a^{100}, a^{101}) = 1/4$: automaton of size 2 computing the parity.
$d(a^{100!}, a^{101!}) = 2^{-101}$: requires an automaton of size 101.

Profinite words: $\widehat{A^*}$ is the completion of A^* with respect to d.

Examples:
$a^\omega \in \widehat{A^*}$ is the limit of $a^{n!}$ when $n \to \infty$.
$a^{\omega-1} \in \widehat{A^*}$ is the limit of $a^{n!-1}$ when $n \to \infty$.

We can now see $x^\omega = x^{\omega+1}$ as a profinite equation characterizing aperiodic monoids.

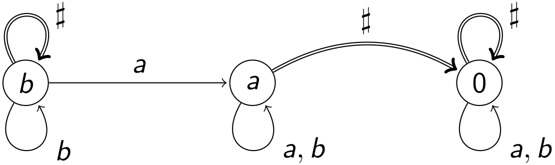
Question: Can we generalize this to cost functions ?

Stabilisation monoids

Generalise monoids to a quantitative setting.
A subset I identifies the "big" elements.
Stabilisation $\sharp$ means "repeat many times" the element.

- $\sharp$ only defined on idempotents ($xx = x$).
- if we "count" a, then $a^\sharp \neq a$, otherwise $a^\sharp = a$.

Example: Stabilisation Monoid for $number_a$
$M = \{b, a, 0\}$, $I = \{0\}$,
b: "no a", a: "a small number of a", 0: "a lot of a".



Cayley graph

Axioms for stabilisation monoids [Colcombet '09]

M monoid, idempotents elements $E(M) = \{x \in M \mid xx = x\}$.

Stabilisation monoid:
ordered monoid M with operation $\sharp : E(M) \to E(M)$ satisfying :

- **S1** for all $s, t \in M$ such that $st \in E(M)$ and $ts \in E(M)$, one has $(st)^\sharp s = s(ts)^\sharp$,
- **S2** for all $e \in E(M)$, $(e^\sharp)^\sharp = e^\sharp = ee^\sharp = e^\sharp e$,
- **S3** for all $e, f \in E(M)$, $e \leqslant f$ implies $e^\sharp \leqslant f^\sharp$,
- **S4** $1^\sharp = 1$.

Language: subset of the free monoid A^*
Cost function: ?

Problems:

- No free stabilisation monoid
- Cost function is not a canonic object.

We need a clean algebraic framework for cost functions.

Language: subset of the free monoid A^*
Cost function: ?

Problems:

- No free stabilisation monoid
- Cost function is not a canonic object.

We need a clean algebraic framework for cost functions.

Solution: replace stabilisation monoids by stabilisation algebras.

- Same behaviour for finite structures,
- Free stabilisation algebras $F(A)$,
- Cost function = subset of $F(A)$.

We can now define varieties of cost functions and stabilisation algebras in the same way as for languages.

Theorem

Varieties of cost functions are in one-to-one correspondence with varieties of stabilisation algebras.

We can also generalise profinite words to profinite $\omega\sharp$-expression, and obtain:

Theorem

Every variety of cost functions is characterized by a set of profinite equations.

Link with varieties of languages

Any variety of languages is in particular a variety of cost functions.

Canonic generalisation of a variety of languages:
Keep the same equations.

Projection of a variety of cost functions to languages:
Add $x^\omega = x^\sharp$ to the set of equations.

Examples:
Commutative languages: $\{xy = yx,\ x^\omega = x^\sharp\}$
Commutative cost functions: $xy = yx$

Examples of varieties:

- Min (Distance automata) and Max varieties have sets of equations [Colcombet+K.+Manuel+Toruńczyk '16].
- $x^\omega = x^{\omega+1}$: aperiodic cost functions [K. '11]
- $(xy^\sharp z)^\omega = (xy^\sharp z)^\sharp$: temporal [Colcombet+K.+Lombardy '10]
- $xy = yx$: Commutative
- $\{xy = yx\ ,\ x^\sharp y^\sharp = (xy)^\sharp\}$: $\sharp$-commutative

Example: $maxblock_a$ is commutative but not $\sharp$-commutative. Many permutations are needed to blow up the length of the biggest block.

Thank you !