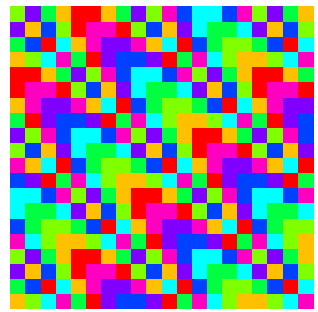
Is Büchi's theorem useful for you?

Michel Rigo

http://www.discmath.ulg.ac.be/
http://orbi.ulg.ac.be/

Model Theory and Applications, Mons, 17th January 2017



I gave a talk on the same subject in May 2015
Workshop on automatic sequences.

But the audience was mostly interested in

COW

I gave a talk on the same subject in May 2015
Workshop on automatic sequences.

But the audience was mostly interested in

Combinatorics On Words MSC:68R15

With an audience of logicians, model theorists, ...
I will briefly sketch what COW is about.

- Let A be a finite *alphabet*, e.g., $A = \{0, 1\}$;
- Let A^* be the set of *finite words* over A.
- An element in $A^{\mathbb{N}}$ is an *infinite word*.

$$1100100100001111110110101010001000100001011010000\cdots$$

$$\texttt{hotellidohotellidohotellidohotellidohotellidohotellido}\cdots$$

We search for, try to avoid, count, compare, ...
factors, repetitions, patterns, ...occurring in (infinite) words.

Example

The word coconut starts with a square.

J. Berstel, D. Perrin, The origins of combinatorics on words, Europ. J. Combin. 28 (2007)

M. Lothaire, *Combinatorics on Words*, CUP (1983)

V. Berthé, M.R., *Combinatorics, Automata and Number Theory*, CUP (2010)

How to advertise COW for students:

The simplest theorem

Over a 2-letter alphabet, every word of length ≥ 4 contains a square, i.e., squares are *unavoidable*.

Proof.

□

Questions:

- What about a 3-letter alphabet?
- What about avoiding cubes or other patterns?

How to advertise COW for students:

The simplest theorem

Over a 2-letter alphabet, every word of length ≥ 4 contains a square, i.e., squares are *unavoidable*.

Proof.

□

Questions:

- What about a 3-letter alphabet?
- What about avoiding cubes or other patterns?

Theorem (A. Thue 1906)

Over a 2-letter alphabet, cubes (even overlaps) are avoidable.

See, for instance, Lothaire (1983).

An *overlap* is a word of the form: $auaua$, `alfalfa` (lucerne)

$$f : 0 \mapsto 01, \quad 1 \mapsto 10$$

$$0 \mapsto 0\,1 \mapsto 01\,10 \mapsto 0110\,1001 \mapsto 01101001\,10010110 \cdots$$

The sequence $(f^n(0))_{n \geq 0}$ *converges* to an infinite word

$$\mathbf{t} = 01101001100101101001011001101001100101100110100 1 \cdots$$

We say that $\mathbf{t}$ is generated by iterating a 2-uniform morphism.

J.-P. Allouche, J. Shallit, The ubiquitous Prouhet-Thue-Morse sequence, *Sequences and their applications* (1998).

Corollary

Over a 3-letter alphabet, squares are avoidable.

$$\{0, 01, 011\} \text{ is a code}$$

every word over $\{0, 01, 011\}$ has a unique factorization.

$$\mathbf{t} = 011|01|0|011|0|01|011|01|0|01|011|0|011|01|0|01\cdots$$

$$321312321231321\cdots \text{ is square-free.}$$

Words obtained by iterating a (uniform) morphism are extensively studied in COW

Theorem (Cobham 1972)

An infinite word is generated by a k-uniform morphism (+ an extra letter-to-letter coding) if and only if it is *k-automatic*.

J.-P. Allouche, J. Shallit, *Automatic Sequences : Theory, Applications, Generalizations*, CUP (2003)

Corollary

Over a 3-letter alphabet, squares are avoidable.

$$\{0, 01, 011\} \text{ is a code}$$

every word over $\{0, 01, 011\}$ has a unique factorization.

$$\mathbf{t} = 011|01|0|011|0|01|011|01|0|01|011|0|011|01|0|01\cdots$$

$$321312321231321\cdots \text{ is square-free.}$$

Words obtained by iterating a (uniform) morphism are extensively studied in COW

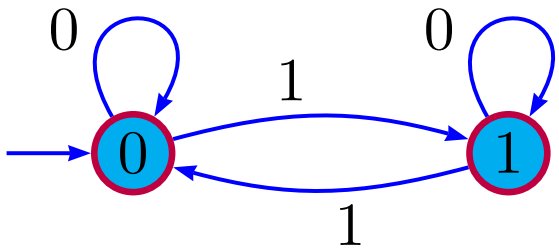
Theorem (Cobham 1972)

An infinite word is generated by a k-uniform morphism (+ an extra letter-to-letter coding) if and only if it is *k-automatic*.

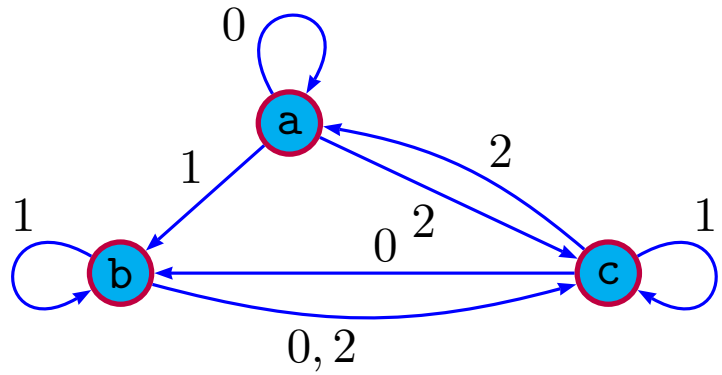
J.-P. Allouche, J. Shallit, *Automatic Sequences : Theory, Applications, Generalizations*, CUP (2003)

$$\mathbf{t} = 01101001100101101001011001101001100101100110100101\cdots$$

Due to Cobham's theorem, it is also be generated by a finite automaton:



$$f : \begin{cases} \mathtt{a} & \mapsto & \mathtt{abc} \\ \mathtt{b} & \mapsto & \mathtt{cbc} \\ \mathtt{c} & \mapsto & \mathtt{bca} \end{cases}$$



$$f^\omega(\mathtt{a}) = \mathtt{abccbcbcabcacbcbcacbcbcaabc}\cdots$$

$$g : \begin{cases} \mathtt{a} & \mapsto & 0 \\ \mathtt{b} & \mapsto & 1 \\ \mathtt{c} & \mapsto & 0 \end{cases}$$

$$g(f^\omega(\mathtt{a})) = 01001010010001010010100010\cdots$$

Back to Erdős in the 60's

Other kinds of questions: an abelian square, e.g., abcbca.

Can we avoid abelian square over a 3-letter alphabet?

$$0102010, \quad 0102101$$

simple exhaustive search.

Back to Erdős in the 60's

Other kinds of questions: an abelian square, e.g., abcbca.

Can we avoid abelian square over a 3-letter alphabet?

$$0102010, \quad 0102101$$

simple exhaustive search.

V. Keränen (ICALP'1992) gave a 85-uniform morphism answering the question!

$$a \mapsto abcacdcbcdcadcdbdabacabadbabcbdbcbacbcdcacbabd$$
$$abacadcbcdcacdbcbacbcdcacdcbdcdadbdcbca;$$

$$b \mapsto bcdbdadcdadbadacabcbdbcbacbcdcacdcbdcdadbdcbca$$
$$bcbdbadcdadbdacdcbdcdadbdadcadabacadcdb;$$

$$c \mapsto cdacabadabacbabdbcdcacdcbdcdadbdadcadabacadcdb$$
$$cdcacbadabacabdadcadabacabadbabcbdbadac;$$

$$d \mapsto dabdbcbabcbdcbcacdadbdadcadabacabadbabcbdbadac$$
$$dadbdcbabcbdbcabadbabcbdbcbacbcdcacbabd;$$

J. Cassaigne, J. D. Currie, L. Schaeffer, J. Shallit, *Avoiding Three Consecutive Blocks of the Same Size and Same Sum*, arXiv:1106.5204

$$\varphi : 0 \mapsto 03,\ 1 \mapsto 43,\ 3 \mapsto 1,\ 4 \mapsto 01$$

$$\varphi^{\omega}(0) = 0314301103434303101101103143034343034\cdots$$

has no additive cube, e.g., 041340.

$$\{k - \text{automatic words}\} \subsetneq \{\text{morphic words}\}$$

A case-study problem in COW

The ultimate periodicity problem (3 variants):

- Given a k-uniform morphism f (prolongable on 0) and a coding g, is the infinite word $g(f^\omega(0))$ ultimately periodic, i.e., of the form $uvvv\cdots$?
- Given a morphism f (prolongable on 0), is the infinite word $f^\omega(0)$ ultimately periodic?
- Given a morphism f (prolongable on 0)and a coding g, is the infinite word $g(f^\omega(0))$ ultimately periodic?

Using only COW techniques, the three problems are known to be decidable:

- J. Honkala, RAIRO 1986
- T. Harju, M. Linna, RAIRO 1986 / J.-J. Pansiot, RAIRO 1986
- F. Durand, RAIRO 2013 / I. Mitrofanov, arXiv 2011

Timeline

- In the 90's, V. Bruyère and the "Mons team" promoted a lot the "logical setting" but mostly in relation with *recognizable sets of integers*
  - V. Bruyère, G. Hansel, C. Michaux, R. Villemaire, Bull. BMS 1992
  - G. Hansel, V. Bruyère, TCS 1997
  - C. Michaux, R. Villemaire, APAL 1996
  - A. Bès, JSL 2000
  - F. Point, V. Bruyère, ToCS 1997
- 2010–2012 Renewal of interest mostly by J. Shallit and his co-authors but oriented towards *decidability in COW*
  - J.-P. Allouche, N. Rampersad, J. Shallit, TCS 2009
  - E. Charlier, N. Rampersad, J. Shallit, IJCS 2012
- Then move to "automatic theorem-proving"
  - D. Goč, D. Henshall, J. Shallit, 2012
  - D. Goč, H. Mousavi, J. Shallit, 2012
  - D. Goč, L. Schaeffer, J. Shallit, 2013
  - D. Goč, N. Rampersad, P. Salimov, M.R., 2013
  - H. Mousavi, J. Shallit, arxiv 2014, ...

M. Presburger (1929)

The first order theory $Th(\langle \mathbb{N}, + \rangle)$ of the natural numbers with addition is decidable.

Proof: $\langle \mathbb{N}, + \rangle$ admits quantifier elimination
$\rightarrow$ *check a finite number of equalities (possibly modulo m) or inequalities of linear combination of integers and variables.*

$$=,\ (\exists x),\ \neg,\ \vee$$

Example of formula (here, a sentence)

$$(\exists x)(\exists y)\neg(\exists z)\neg\big\{\neg(x+y=z\vee x=y+y)\\ \vee(\forall u)[(x=u)\vee\neg(y=u+z)]\big\}$$

All variables are in the scope of a quantifier $\rightarrow$ `True`/`False`.

M. Presburger (1929)

The first order theory $Th(\langle \mathbb{N}, + \rangle)$ of the natural numbers with addition is decidable.

Proof: $\langle \mathbb{N}, + \rangle$ admits quantifier elimination
$\rightarrow$ *check a finite number of equalities (possibly modulo m) or inequalities of linear combination of integers and variables.*

$$\boxed{=,\ (\exists x),\ \neg,\ \vee,\ (\forall x),\ \wedge,\ \rightarrow,\ \leftrightarrow,\ \leq, <}$$

$$x \leq y \equiv (\exists z)(x + z = y)$$

$$x < y \equiv (x \leq y) \wedge \neg(x = y)$$

Example of formula (here, a sentence)

$$(\exists x)(\exists y)(\forall z)\big\{(x + y = z \vee x = y + y) \rightarrow (\forall u)[(x = u) \vee \neg(y = u + z)]\big\}$$

All variables are in the scope of a quantifier $\rightarrow$ `True`/`False`.

Example

The following sentence is true

$$(\forall x)(\exists y)[x = y + y \vee x = \mathcal{S}(y + y)]$$

where $\mathcal{S}(x) = y \equiv (x < y) \wedge (\forall z)(x < z \rightarrow (y \leq z))$.

We can define *constants*

$$x = 0 \equiv (\forall y)(x \leq y), \quad 1 = \mathcal{S}(0), \quad 2 = \mathcal{S}(1), \ldots$$

and we can define *multiplication by a constant* and *congruences*

$$2x \equiv x + x, \quad k.x = \underbrace{x + \cdots + x}_{k \text{ times}},$$

$$x \equiv_k y \equiv (\exists z)(x = y + k.z \vee y = x + k.z).$$

A less trivial example (Frobenius' problem)

Chicken McNuggets can be purchased only in 6, 9, or 20 pieces.
The largest number of nuggets that cannot be purchased is 43.

$$(\forall n)(n > 43 \rightarrow (\exists x, y, z \geq 0)(n = 6x + 9y + 20z))$$

$$\wedge \neg((\exists x, y, z \geq 0)(43 = 6x + 9y + 20z)) .$$

We can also define subsets of $\mathbb{N}$

Defining a subset of $\mathbb{N}$

$$\varphi(x) \equiv (\exists y)[\overbrace{x}^{\text{free variable}} = \mathcal{S}(y+y)]$$

$$\{n \in \mathbb{N} \mid \langle \mathbb{N}, + \rangle \models \varphi(n)\} = 2\mathbb{N} + 1$$

Remark

A subset of $\mathbb{N}$ is definable in $\langle \mathbb{N}, + \rangle$ if and only if it is ultimately periodic, i.e., a finite union of arithmetic progressions along with a finite set.

We can also define subsets of $\mathbb{N}^d$

Presburger definable sets

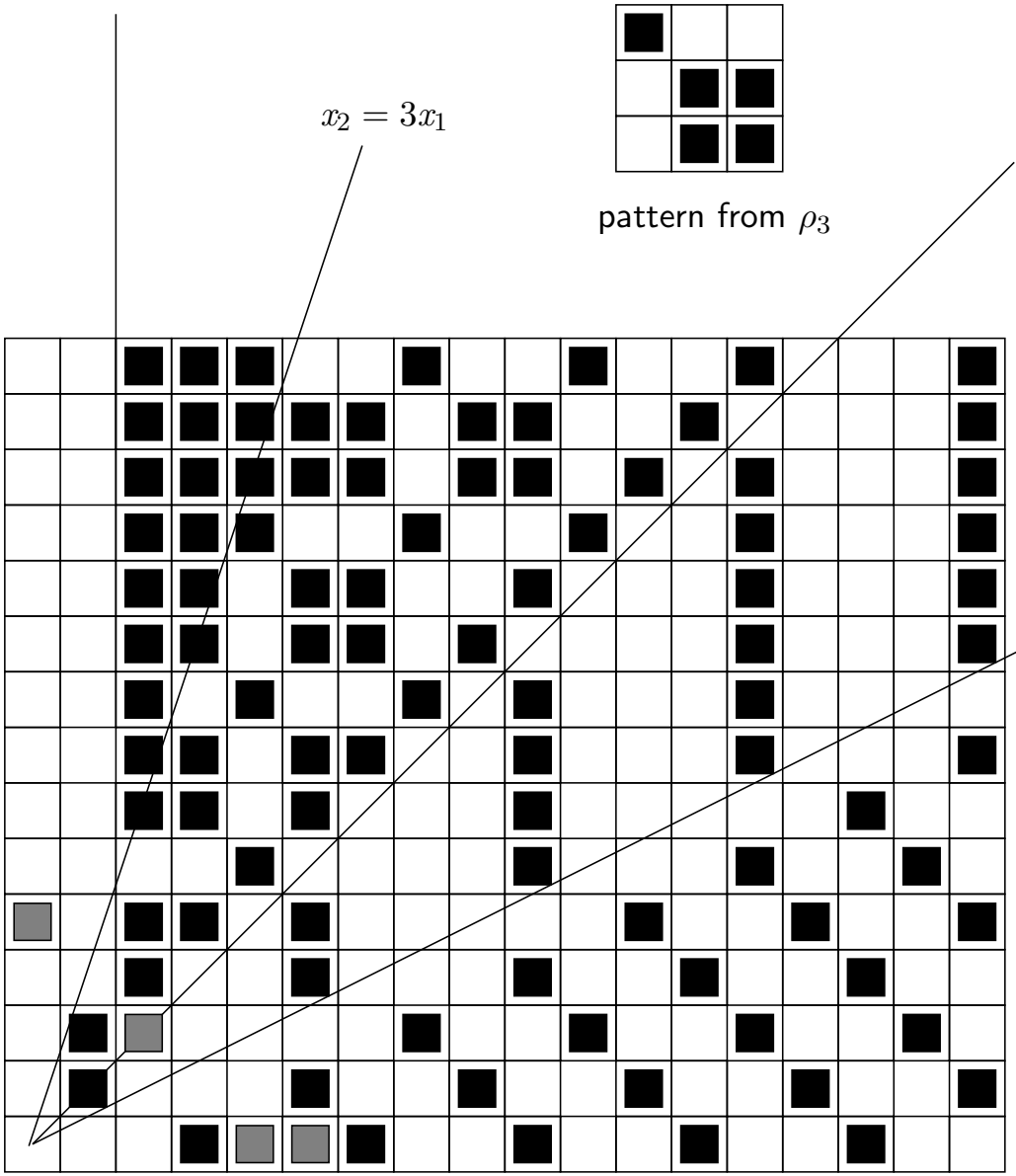
A formula $\varphi(x_1, \ldots, x_d)$ with d free variables,

$$\{(n_1, \ldots, n_d) \in \mathbb{N}^d \mid \langle \mathbb{N}, + \rangle \models \varphi(n_1, \ldots, n_d)\}$$

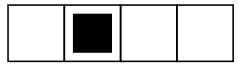
$\varphi(x_1, x_2) \equiv \rho_1(x_1, x_2) \vee \rho_2(x_1, x_2) \vee \rho_3(x_1, x_2) \vee \rho_4(x_1, x_2) \vee \phi(x_1, x_2)$
where

$$\begin{aligned}
\rho_1(x_1, x_2) &\equiv (2x_2 < x_1) \wedge (x_1 + x_2 \equiv_3 0)\,, \\
\rho_2(x_1, x_2) &\equiv (2x_2 \geq x_1) \wedge (x_2 < x_1) \wedge (x_1 \equiv_4 1)\,, \\
\rho_3(x_1, x_2) &\equiv \underbrace{(x_2 > x_1) \wedge (x_2 < 3x_1)}_{\text{a region}} \wedge \underbrace{((2x_1 + x_2 \equiv_3 1) \vee (x_1 + x_2 \equiv_3 0))}_{\text{a pattern}}\,, \\
\rho_4(x_1, x_2) &\equiv (x_2 \geq 3x_1) \wedge (x_1 \geq 2)\,, \\
\phi(x_1, x_2) &\equiv (x_1 = 0 \wedge x_2 = 4) \vee (x_1 = 2 \wedge x_2 = 2) \vee (x_1 = 4 \wedge x_2 = 0) \\
&\quad \underbrace{\vee (x_1 = 5 \wedge x_2 = 0)}_{\text{a few isolated points}}\,.
\end{aligned}$$

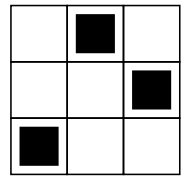
Generalization of ultimately periodic sets



pattern from ρ_3



pattern from ρ_2



pattern from ρ_1

Expansion

J.R. Büchi 1960

Using finite automata constructions, the first order theory of the extension of $\langle \mathbb{N}, + \rangle$ with V_k is still decidable.

Let $k \geq 2$, $V_k(x)$ is the largest power of k dividing x; $V_k(0) = 1$.

Characterization of k-automatic sequences

The infinite word $\mathbf{x}$ *over* A *is* k*-automatic if and only if, for each* $a \in A$*, there exists a formula* $\varphi_{\mathbf{x},a}(n)$ *of* $\langle \mathbb{N}, +, V_k \rangle$ *such that* $\varphi_{\mathbf{x},a}(n)$ *holds if and only if* $\mathbf{x}(n) = a$.

We can still define subsets of $\mathbb{N}$ or $\mathbb{N}^d$, e.g.,

$$fiber_a(\mathbf{x}) = \{n \in \mathbb{N} \mid \langle \mathbb{N}, +, V_k \rangle \models \varphi_{\mathbf{x},a}(n)\}$$

Example 1 in $\langle \mathbb{N}, + \rangle$

Let $A = \{a, b\}$ and $\varphi_a(n) \equiv (\exists y)(n = 2y)$.
We get the sequence $(ab)^\omega = abababab \cdots$ which is k-automatic for all $k \geq 2$.

$$f : a \mapsto aba, \;\; b \mapsto bab$$

Example 2 in $\langle \mathbb{N}, +, V_2 \rangle$

Let $A = \{a, b, c\}$ and

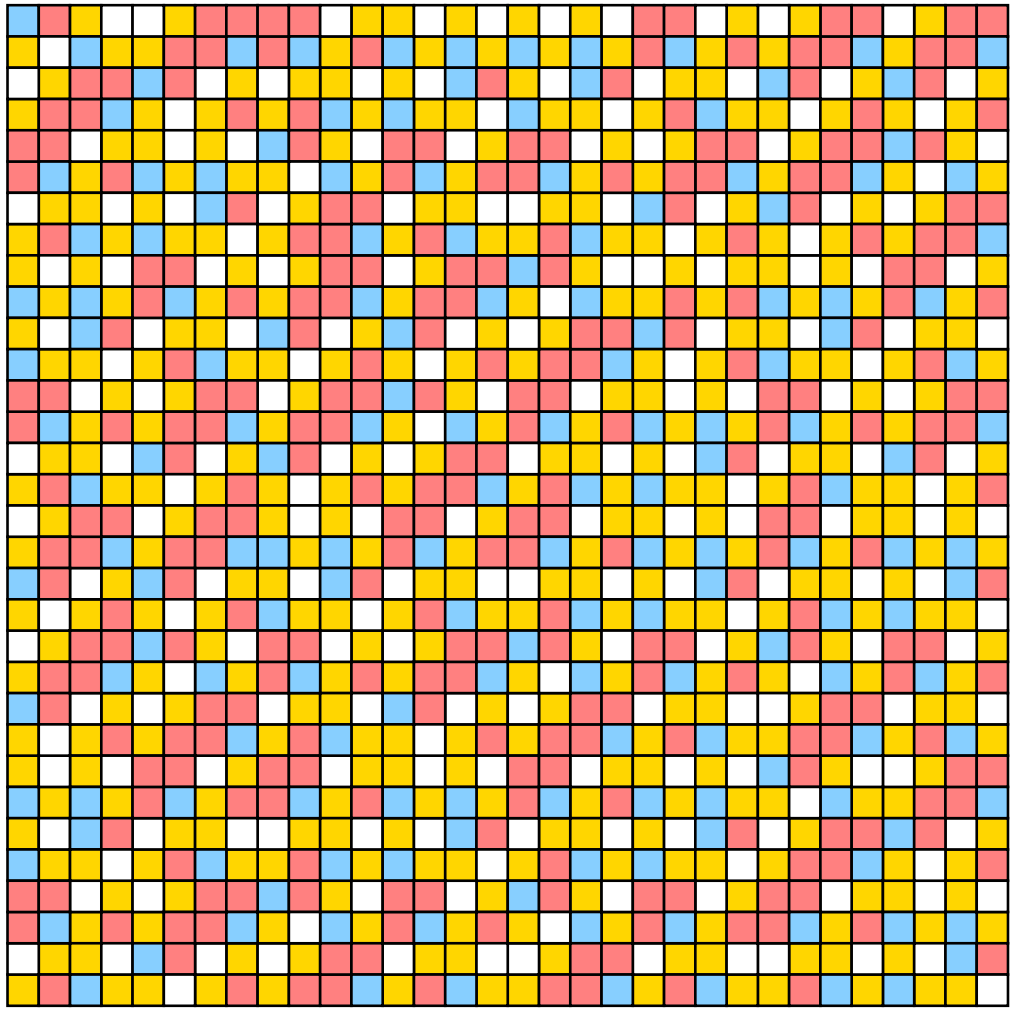
$$\varphi_b(n) \equiv V_2(n) = n, \quad \varphi_c(n) \equiv (n \geq 1) \wedge \neg \varphi_b(n).$$

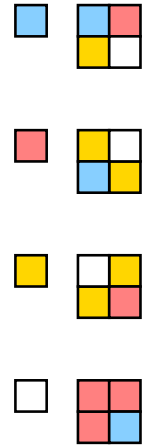
$$f : a \mapsto ab, \;\; b \mapsto bc, \;\; c \mapsto cc, \quad g : b \mapsto 1, \; a, c \mapsto 0$$

$$f^\omega(a) = a\,b\,b\,c\,b\,ccc\,b\,ccccccc\,b\,cccc \cdots$$

$g(f^\omega(a))$ is the characteristic sequence of $\{2^n \mid n \geq 1\}$.

An example of 2-dimensional 2-automatic sequence





We have four formulas of the kind $\varphi_{\square}(x, y)$,
$\{(x, y) \in \mathbb{N}^2 \mid \langle \mathbb{N}, +, V_k \rangle \models \varphi_{\square}(x, y)\}$

Sketch of the proof of Büchi's thm.

From automata to formula

Idea: given a DFA accepting r-tuples of base-k expansions *conveniently padded*, obtain a formula ψ from $\langle \mathbb{N}, +, V_k \rangle$ with r free variables coding exactly the behavior of the automaton:

$$\psi(x_1, \ldots, x_r) \equiv (\exists n_1) \cdots (\exists n_{\#Q}) \varphi(x_1, \ldots, x_r, n_1, \ldots, n_{\#Q}).$$

Similar to the proof showing that every function computable by a Turing machine is recursive.

- states are coded by vectors in $\{0,1\}^{\#Q}$
- a path is thus coded by $\#Q$ base-k expansions of integers
- start in the initial state (least significant digits)
- end in a final state (most significant digits)
- compatible with the transition function of the DFA

See, for instance, Bruyère, Hansel, Michaux, Villemaire (1992).

from formula to automata (i.e., the most interesting part for us)

Example

Consider $\varphi(n) \equiv (\exists x)(\exists y)(\, V_2(x) = x \wedge n = x + 3.y)$.
Find a DFA accepting the base-2 expansions of the elements in

$$\{n \in \mathbb{N} \mid \langle \mathbb{N}, +, V_2 \rangle \models \varphi(n)\}$$

$1, 2, 4, 5, 7, 8, 10, 11, 13, 14, 16, 17, 19, 20, 22, 23, 25, 26, \ldots$

Example

Consider
$\psi(n, m) \equiv \varphi(n) \wedge (n \equiv_2 0 \to m = 2.n) \wedge (n \equiv_2 1 \to m = 3.n)$.
Find a DFA accepting the base-2 expansions of the elements in

$$\{(n_1, n_2) \in \mathbb{N} \mid \langle \mathbb{N}, +, V_2 \rangle \models \psi(n_1, n_2)\}$$

1	2	4	5	7	8	10	11	13	14	16	17	19	20	$\ldots$
3	4	8	15	21	16	20	33	39	28	32	51	57	40	$\ldots$

Formulas are defined inductively, thus start with atomic formulas, proceed by induction on the length of the formula.
Construction of automata, at least, for $\neg$, $\vee$, $=$, $(\exists x)$, V_k, $+$

- complementation of automata
- union of automata

$\longrightarrow$ Build bigger automata from smaller ones, determinize when needed, and also minimize.

Remark

This provides an alternative proof of Presburger's result.
Given a sentence, there is an outermost quantifier, e.g., $(\exists x)\varphi(x)$.

Deciding whether a DFA accepts at least one word is decidable (empty problem/universality problem for DFA).

Formulas are defined inductively, thus start with atomic formulas, proceed by induction on the length of the formula.
Construction of automata, at least, for $\neg$, $\vee$, $=$, $(\exists x)$, V_k, $+$

- complementation of automata
- union of automata

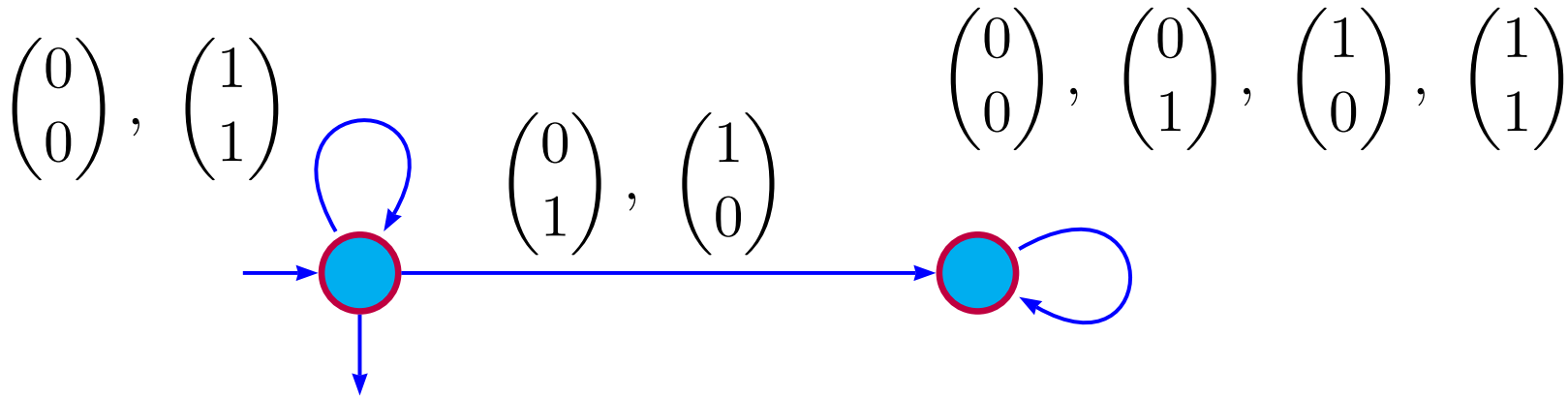
$\longrightarrow$ Build bigger automata from smaller ones, determinize when needed, and also minimize.

Remark

This provides an alternative proof of Presburger's result.
Given a sentence, there is an outermost quantifier, e.g., $(\exists x)\varphi(x)$.

Deciding whether a DFA accepts at least one word is decidable (empty problem/universality problem for DFA).

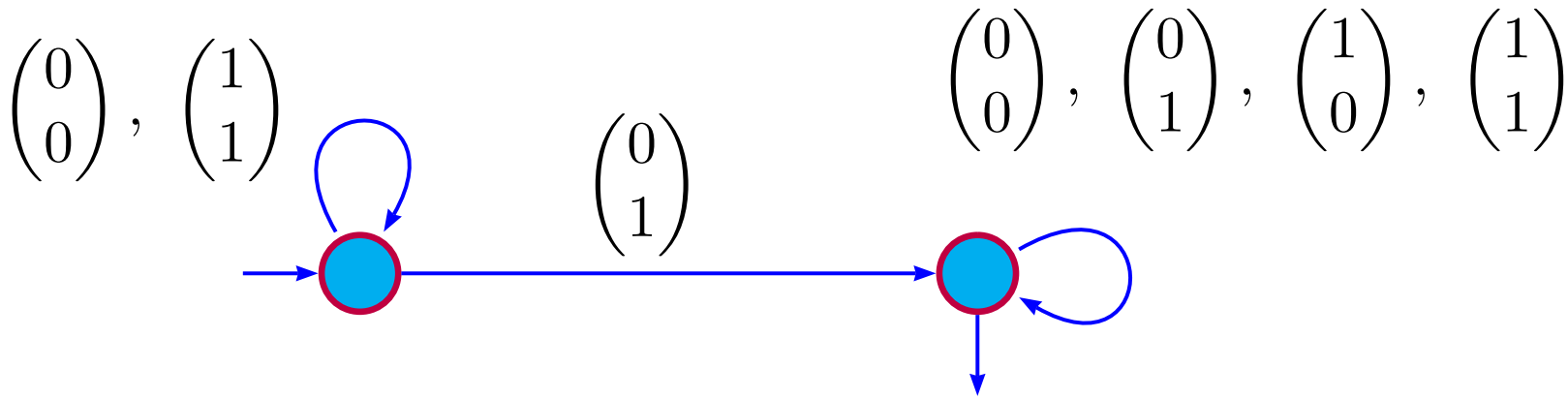
- DFA for $=$, $\{(x, y) \in \mathbb{N}^2 \mid x = y\}$



In every figure, we will consider base-2 expansions

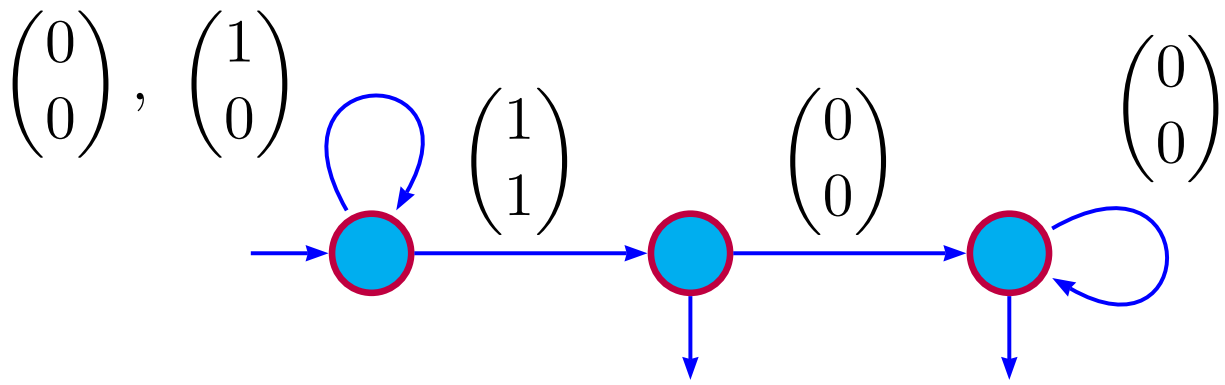
- DFA reading m.s.d.f. first for $<$ (extra construction),

$\{(x, y) \in \mathbb{N}^2 \mid x < y\}$



$$x < y \Leftrightarrow \mathsf{rep}_2(x) <_{\text{gen}} \mathsf{rep}_2(y).$$

- DFA reading m.s.d. first for V_k, $\{(x, y) \in \mathbb{N}^2 \mid y = V_2(x)\}$



$$\begin{array}{ccccccccc} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}$$

about existential quantifier

$$\begin{pmatrix} x \\ y_1 \\ \vdots \\ y_r \end{pmatrix}$$

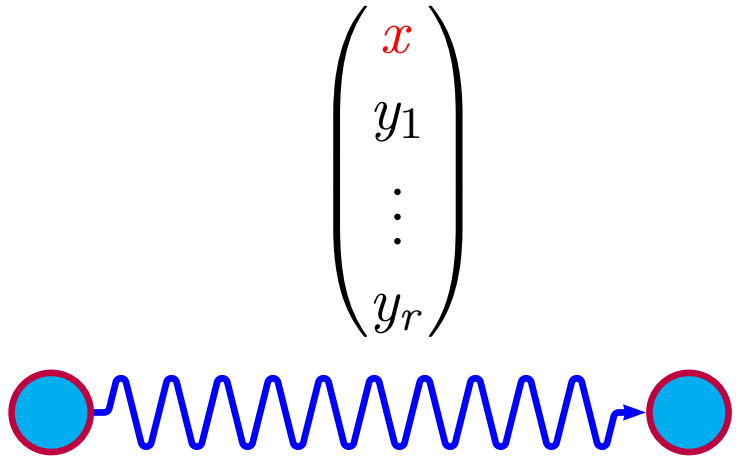
$$\psi(x, y_1, \ldots, y_r)$$

$$\begin{pmatrix} y_1 \\ \vdots \\ y_r \end{pmatrix}$$

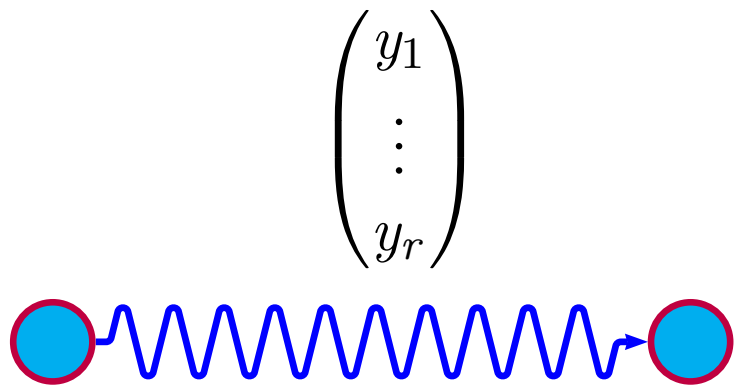
$$(\exists x)\psi(x, y_1, \ldots, y_r)$$

Get a nondeterministic automaton!

about existential quantifier



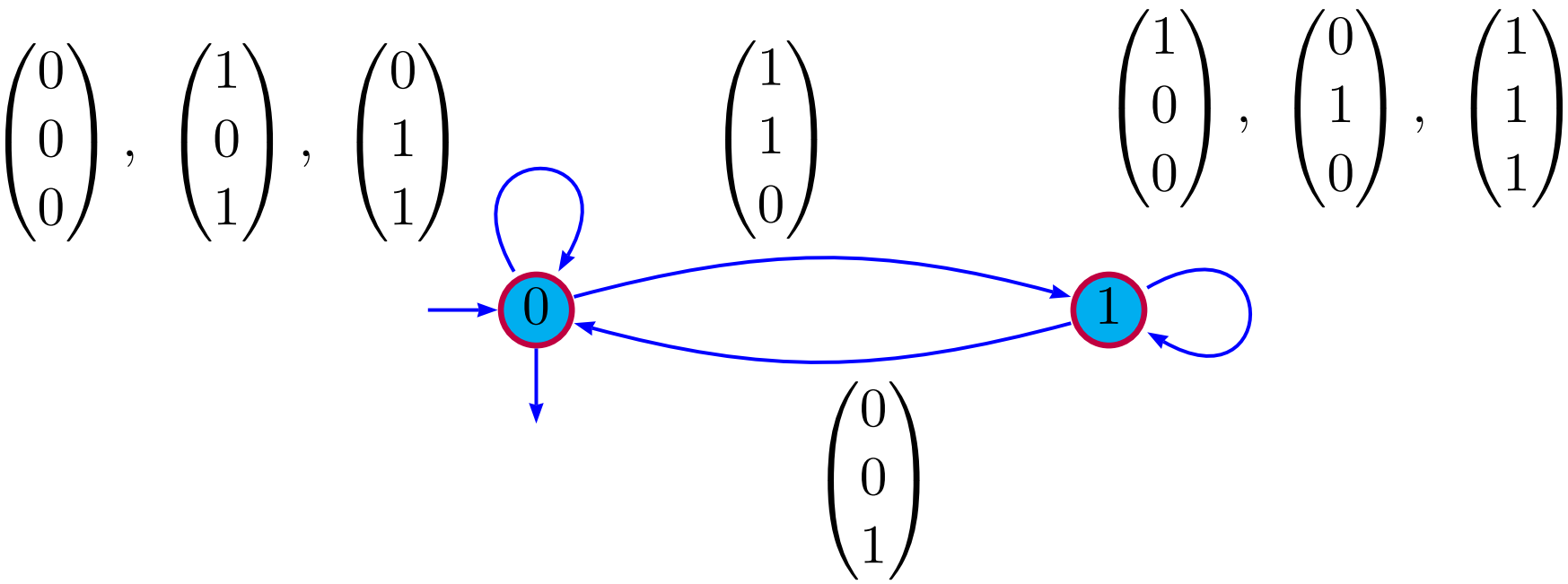
$\psi(x, y_1, \ldots, y_r)$



$(\exists x)\psi(x, y_1, \ldots, y_r)$

Get a nondeterministic automaton!

- DFA reading l.s.d. first for $+$, $\{(x, y, z) \in \mathbb{N}^3 \mid x + y = z\}$

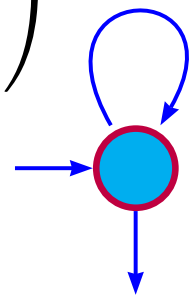


Such a DFA is easily obtained in every base (limited carry propagation).

Remark about normalization

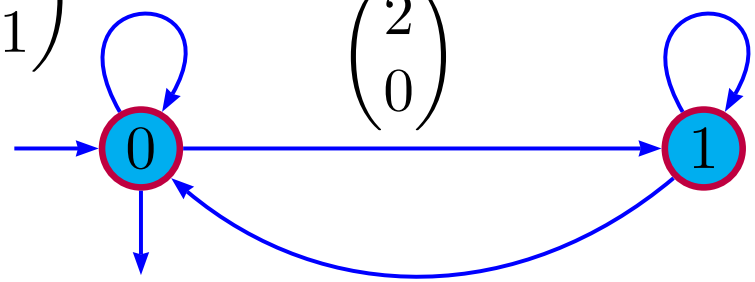
(i) add digit without carry (alphabet twice bigger)

$$\binom{0}{0}, \binom{0}{1}, \binom{1}{0}, \binom{1}{1}$$



(ii) *normalize*, DFA reading l.s.d. first, e.g. $(0121, 1001)$

$$\binom{0}{0}, \binom{1}{1}$$



$$\binom{2}{0}$$

$$\binom{1}{0}, \binom{2}{1}$$

$$\binom{0}{1}$$

Without logical techniques (Honkala 1986)

Ultimate periodicity problem (version 1)

INSTANCE: a k-uniform morphism f prolongable on a, a coding g
DECIDE whether $\mathbf{x} = g(f^\omega(a))$ is ultimately periodic?

Since $\mathbf{x}$ is k-automatic, for each a in A, we have a formula $\varphi_{\mathbf{x},a}(n)$ which holds iff $\mathbf{x}(n) = a$.

$$\mathbf{x}(i) = \mathbf{x}(j) \equiv \bigvee_{a \in A} (\varphi_{\mathbf{x},a}(i) \wedge \varphi_{\mathbf{x},a}(j))$$

$$(\exists p)(\exists N)(\forall i \geq N)\, \mathbf{x}(i) = \mathbf{x}(i+p).$$

We can decide with automata.

Without logical techniques (Honkala 1986)

Ultimate periodicity problem (version 1)

INSTANCE: a k-uniform morphism f prolongable on a, a coding g
DECIDE whether $\mathbf{x} = g(f^\omega(a))$ is ultimately periodic?

Since $\mathbf{x}$ is k-automatic, for each a in A, we have a formula $\varphi_{\mathbf{x},a}(n)$ which holds iff $\mathbf{x}(n) = a$.

$$\mathbf{x}(i) = \mathbf{x}(j) \equiv \bigvee_{a \in A} (\varphi_{\mathbf{x},a}(i) \wedge \varphi_{\mathbf{x},a}(j))$$

$$(\exists p)(\exists N)(\forall i \geq N)\, \mathbf{x}(i) = \mathbf{x}(i+p).$$

We can decide with automata.

As a summary

Reformulation by Charlier, Rampersad, Shallit

Theorem

Let $k \geq 2$.
If one can express a property of a k-automatic sequence $\mathbf{x}$ using:

(first-order) quantifiers, logical operations, integer variables, addition, subtraction,
indexing into $\mathbf{x}$ and comparison of integers or elements of $\mathbf{x}$,

then this property is decidable.

Some applications

A. Thue (1906)

The Thue–Morse word $\mathbf{t}$ is overlap-free.

$$\neg(\exists i)(\exists \ell \geq 1)[(\forall j < \ell)(\mathbf{t}(i+j) = \mathbf{t}(i+\ell+j)) \wedge \mathbf{t}(i) = \mathbf{t}(i+2\ell)]$$

Exercise

Write a formula that expresses the (non)existence of a square, a cube, a fixed n-power, a factor xxx^R, in a k-automatic word.

J. Currie, N. Rampersad, Growth rate of binary words avoiding xxx^R, TCS (2016)

Charlier-Rampersad-Shallit

It is decidable if a k-automatic sequence contains powers of arbitrarily large exponent.

The formula

$$\psi(n,j) \equiv (\exists i)(\forall t < n)\mathbf{x}(i+t) = \mathbf{x}(i+j+t)$$

should hold for arbitrarily large n/j

How to check $(\forall i)(\exists n)(\exists j)[n > j.k^i \wedge \psi(n,j)]$?

If the DFA for $\psi(n,j)$ reads l.s.d. first, we should have strings ending in

$$\cdots \underbrace{\begin{pmatrix}\star\\0\end{pmatrix}\begin{pmatrix}\star\\0\end{pmatrix}\cdots\begin{pmatrix}\star\\0\end{pmatrix}}_{i}\begin{pmatrix}\neq 0\\0\end{pmatrix}$$

One can decide if a DFA accepts such arbitrarily long strings.

Charlier-Rampersad-Shallit

It is decidable if a k-automatic sequence contains powers of arbitrarily large exponent.

The formula

$$\psi(n,j) \equiv (\exists i)(\forall t < n)\mathbf{x}(i+t) = \mathbf{x}(i+j+t)$$

should hold for arbitrarily large n/j

How to check $(\forall i)(\exists n)(\exists j)[n > j.k^i \wedge \psi(n,j)]$?

If the DFA for $\psi(n,j)$ reads l.s.d. first, we should have strings ending in

$$\cdots \underbrace{\binom{\star}{0}\binom{\star}{0}\cdots\binom{\star}{0}}_{i}\binom{\neq 0}{0}$$

One can decide if a DFA accepts such arbitrarily long strings.

Quite a few properties that can be checked for k-automatic sequences:

- (arbitrarily large) unbordered factors
- reccurrent word
- linearly recurrent word
- $\mathrm{Fac}(\mathbf{x}) \subset \mathrm{Fac}(\mathbf{y})$
- $\mathrm{Fac}(\mathbf{x}) = \mathrm{Fac}(\mathbf{y})$
- existence of an unbordered factor of length n

$\vdots$

Computer Science > Formal Languages and Automata Theory

Automatic Theorem Proving in Walnut

Hamoon Mousavi

(Submitted on 18 Mar 2016)

Walnut is a software package that implements a mechanical decision procedure for deciding certain combinatorial properties of some special words referred to as automatic words or automatic sequences. Walnut is written in Java and is open source. It is licensed under GNU General Public License.

Subjects: **Formal Languages and Automata Theory (cs.FL)**; Logic in Computer Science (cs.LO); Mathematical Software (cs.MS); Combinatorics (math.CO)

Cite as: **arXiv:1603.06017 [cs.FL]**

(or **arXiv:1603.06017v1 [cs.FL]** for this version)

Submission history

From: Seyyed Hamoon Mousavi Haji [view email]

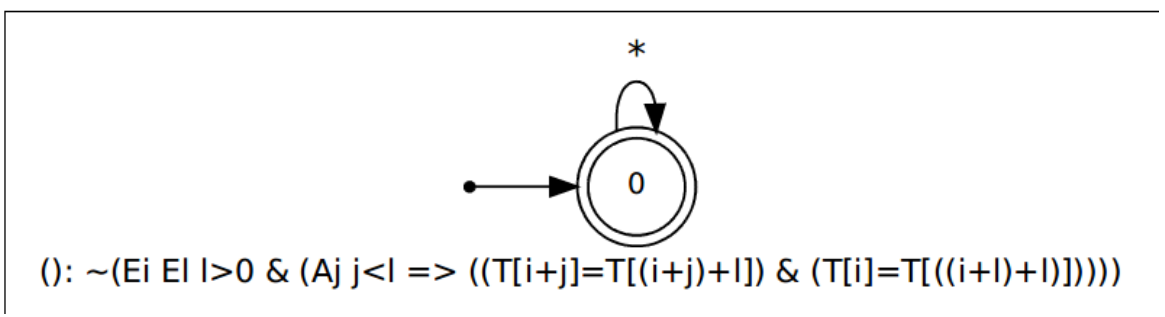
[v1] Fri, 18 Mar 2016 23:53:10 GMT (684kb,D)

$$\neg(\exists i)(\exists \ell \geq 1)[(\forall j < \ell)(\mathbf{t}(i+j) = \mathbf{t}(i+\ell+j)) \wedge \mathbf{t}(i) = \mathbf{t}(i+2\ell)]$$

Up to 97 states in an intermidiate step

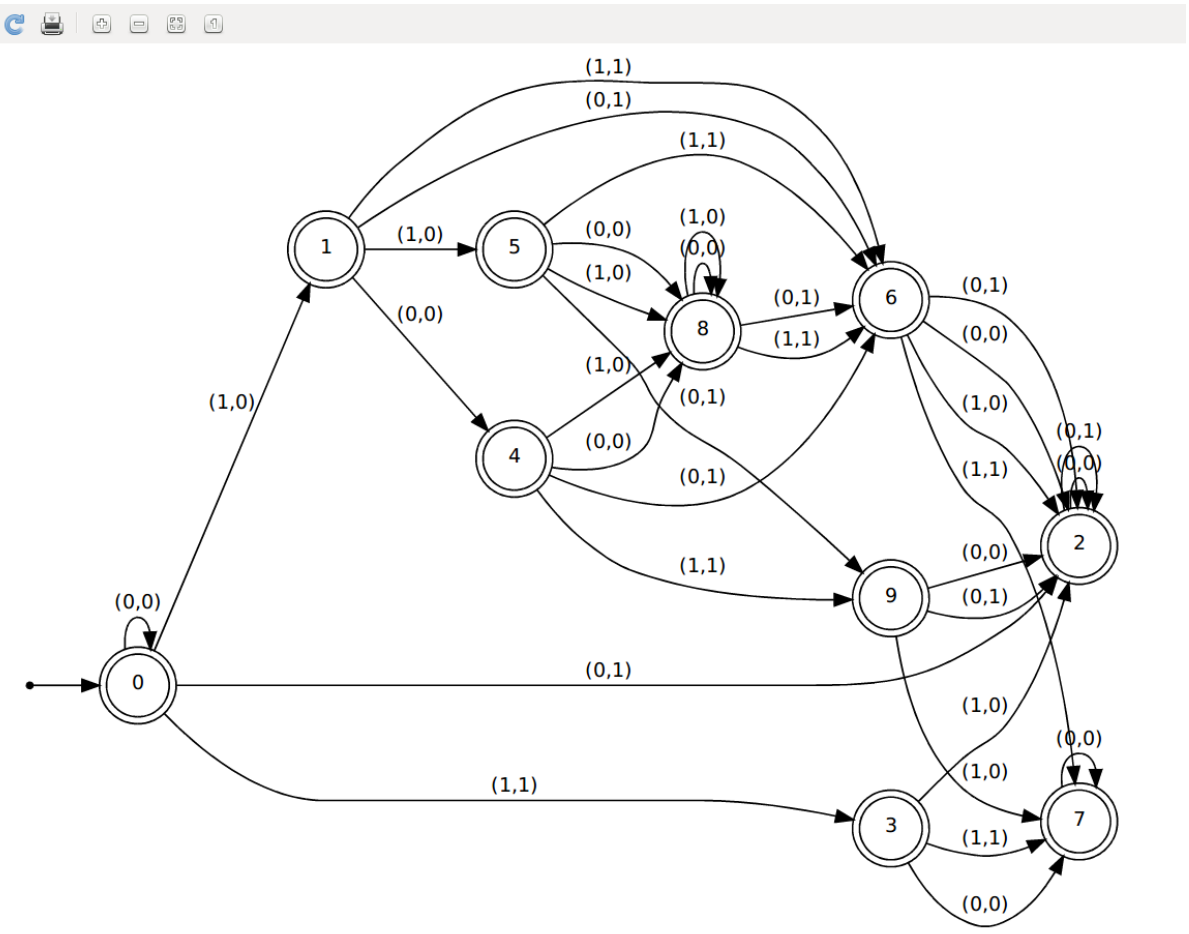
```
rigo@X1:~/Walnut/Walnut/Walnut/bin$ java Main.prover
eval test "~(Ei El l>0 & (Aj j<l => ((T[i+j]=T[(i+j)+l]) & (T[i]=T[((i+l)+l)])))))":
l>0 has 2 states: 14ms
 j<l has 2 states: 0ms
  T[(i+j)]=T[((i+j)+l)] has 12 states: 136ms
   T[i]=T[((i+l)+l)] has 6 states: 39ms
    (T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)]) has 72 states: 41ms
     (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)])) has 97 states: 62ms
      (A j (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)]))) has 1 states: 186ms
       (l>0&(A j (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)])))) has 1 states: 1ms
        (E l (l>0&(A j (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)]))))) has 1 states: 0ms
         (E i (E l (l>0&(A j (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)])))))) has 1 states: 1ms
          ~(E i (E l (l>0&(A j (j<l=>(T[(i+j)]=T[((i+j)+l)]&T[i]=T[((i+l)+l)])))))) has 1 states: 0ms
total computation time: 506ms
```

GraphViz / `xdot ../Result/test.gv`



$$\psi(n,j) \equiv (\exists i)(\forall t < n)\mathbf{x}(i+t) = \mathbf{x}(i+j+t)$$

```
rigo@X1:~/Walnut/Walnut/Walnut/bin$ java Main.prover
eval test "Ei At (t<n => (T[i+t]=T[i+j+t]))":
t<n has 2 states: 0ms
 T[(i+t)]=T[((i+j)+t)] has 12 states: 130ms
  (t<n=>T[(i+t)]=T[((i+j)+t)]) has 25 states: 19ms
   (A t (t<n=>T[(i+t)]=T[((i+j)+t)])) has 17 states: 369ms
    (E i (A t (t<n=>T[(i+t)]=T[((i+j)+t)]))) has 10 states: 14ms
total computation time: 556ms
```



m.s.d.f.

A Glimpse at enumeration

Let $\mathbf{x}$ be a k-automatic sequence.

- Same factor of length n occurring in position i and j

$$F_{\mathbf{x}}(n,i,j) \equiv (\forall k < n)(\mathbf{x}(i+k) = \mathbf{x}(j+k))$$

- First occurrence of a factor of length n occurring in position i

$$P_{\mathbf{x}}(n,i) \equiv (\forall j < i)\neg F_{\mathbf{x}}(n,i,j)$$

The set $\{(n,i) \mid P_{\mathbf{x}}(n,i) \text{ true}\}$ is k-recognizable and

$$\forall n \geq 0, \quad \#\{i \mid P_{\mathbf{x}}(n,i) \text{ true}\} = p_{\mathbf{x}}(n).$$

- From the paper by Charlier, Rampersad and Shallit
  $\rightarrow$ k-regular sequences

k-automatic $<$ k-synchronized $<$ k-regular sequence

- D. Goč, L. Schaeffer, J. Shallit, Subword Complexity and k-Synchronization (DLT 2013)

Let $\mathbf{x}$ be a k-automatic sequence.

- $p_{\mathbf{x}}$ is a k-synchronized function
- the function counting the number of distinct length-n factors that are powers is k-synchronized
- the function counting the number of distinct length-n factors that are primitive words is k-synchronized

Also for morphic words?

For instance, what about the ultimate periodicity problem variants?

$$a \mapsto ab, \quad b \mapsto a$$

$$abaababaabaababaababaabaab \cdots$$

Is this infinite word ultimately periodic?

Definition

A *Pisot number* is an algebraic integer $\alpha > 1$ whose conjugates have modulus less than one

Natural generalization of base-k numeration systems

Numeration basis

Let $U = (U_n)_{n \geq 0}$ be an increasing linear recurrent sequence of integers such that $U_0 = 1$.
Assume moreover that the characteristic polynomial of the recurrence relation is the minimal polynomial of a Pisot number.

Example: Fibonacci/Zeckendorf numeration system $X^2 - X - 1$, $(1 + \sqrt{5})/2 \simeq 1.618$, $|(1 - \sqrt{5})/2| < 1$

$$(U_n)_{n \geq 0} = 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \ldots$$

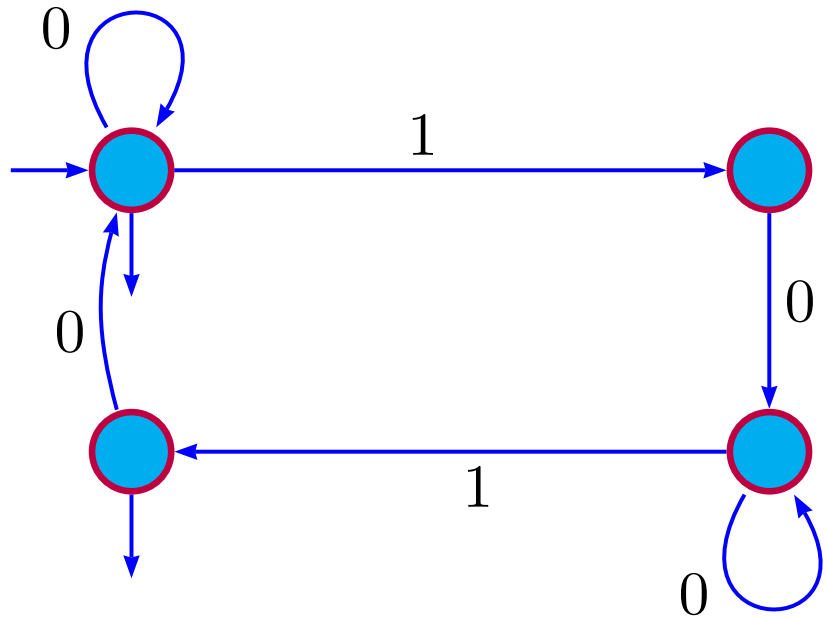
$$\varepsilon,\ 1,\ 10,\ 100,\ 101,\ 1000,\ 1001,\ 1010,\ 10000, \ldots$$

Bruyère–Hansel (1997)

Let U be a "Pisot numeration basis".
A set of $\mathbb{N}^d$ is U-recognizable iff it is definable in $\langle \mathbb{N}, +, V_U \rangle$

$V_U(n)$ is the least U_j occurring in the U-expansion of n with a non-zero digit.

An example of U-recognizable set



ε (0), 101 (4), 1001 (6), 1010 (7), 10001 (9), ...

1) Let U be a "Pisot numeration basis".
The set of all greedy U-expansions is regular.
A bit more complicated than base-k (some technicalities).

2) Frougny's normalization (1985)

Let U be a "Pisot numeration basis".
Normalization (from any finite alphabet) and thus addition, are computable by finite automata.

3) Again, from formula to automata...
Construction of automata, at least, for $\neg$, $\vee$, $=$, $(\exists x)$, V_U, $+$

1) Let U be a "Pisot numeration basis".
The set of all greedy U-expansions is regular.
A bit more complicated than base-k (some technicalities).

2) Frougny's normalization (1985)

Let U be a "Pisot numeration basis".
Normalization (from any finite alphabet) and thus addition, are computable by finite automata.

3) Again, from formula to automata...
Construction of automata, at least, for $\neg$, $\vee$, $=$, $(\exists x)$, V_U, $+$

Application to (families of) words such as

$$abaababaabaababaababaabaababaabaab\cdots$$

$\varphi_a(n)$ definable in $\langle \mathbb{N}, +, V_F \rangle$

arXiv J. Shallit et al., Decision Algorithms for Fibonacci-Automatic Words, with Applications to Pattern Avoidance

$$abacabaabacababacabaabacabacabaac\cdots$$

$\varphi_a(n)$, $\varphi_b(n)$ definable in $\langle \mathbb{N}, +, V_T \rangle$

arXiv H. Mousavi, J. Shallit, Mechanical Proofs of Properties of the Tribonacci Word

Nothing left?

What about abelian properties? Avoiding 3 consecutive blocks of the same size and sum

- Two factors of length n occurring in position i and j are abelian equivalent

$$A_{\mathbf{x}}(n,i,j) \equiv (\exists \nu \in S_n)(\forall k < n)(\mathbf{x}(i+k) = \mathbf{x}(\nu(j+k)))$$

  The length of the formula is $\simeq n!$ and grows with n.

- First occurrence (up to abelian equivalence) of a factor of length n occurring in position i

$$AP_{\mathbf{x}}(n,i) \equiv (\forall j < i)\neg A_{\mathbf{x}}(n,i,j)$$

For a constant n. The set $\{i \mid AP_{\mathbf{x}}(n,i) \text{ true}\}$ is k-recognizable and

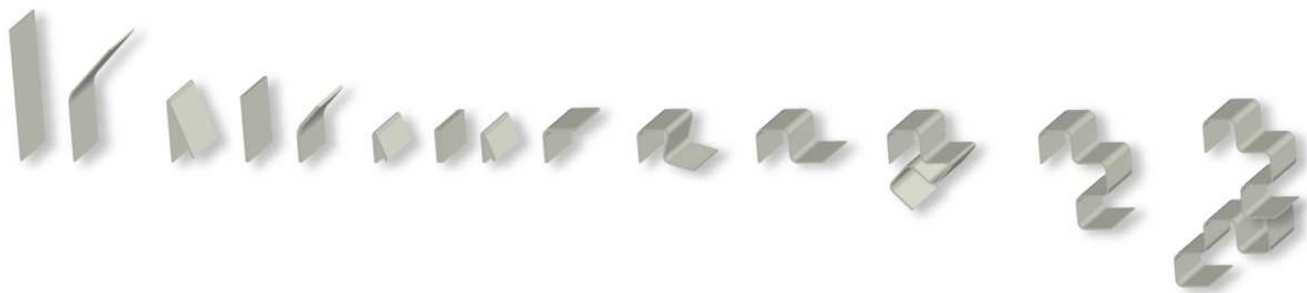
$$\#\{i \mid AP_{\mathbf{x}}(n,i) \text{ true}\} = a_{\mathbf{x}}(n).$$

For instance, Henshall and Shallit ask

- *Can the techniques be applied to detect abelian powers in automatic sequences?*

L. Schaeffer: the set of occurrences of abelian squares in the (2-automatic) paperfolding word is not 2-recognizable.

$$f : a \mapsto ab,\ b \mapsto ac,\ c \mapsto dc,\ d \mapsto db, \quad g : a, b \mapsto 1,\ c, d \mapsto 0$$



Remark

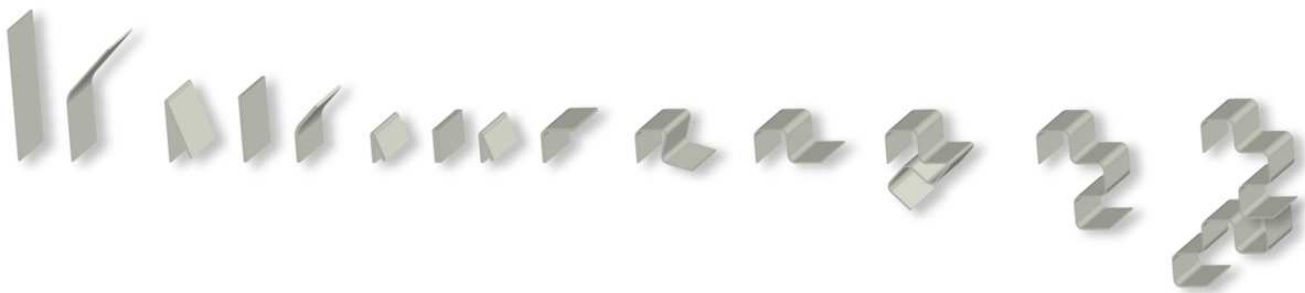
The Thue–Morse word is abelian periodic, $\mathbf{t} \in \{ab, ba\}^\omega$, therefore abelian equivalence is "easy".

For instance, Henshall and Shallit ask

- *Can the techniques be applied to detect abelian powers in automatic sequences?*

L. Schaeffer: the set of occurrences of abelian squares in the (2-automatic) paperfolding word is not 2-recognizable.

$$f : a \mapsto ab,\ b \mapsto ac,\ c \mapsto dc,\ d \mapsto db, \quad g : a, b \mapsto 1,\ c, d \mapsto 0$$



Remark

The Thue–Morse word is abelian periodic, $\mathbf{t} \in \{ab, ba\}^\omega$, therefore abelian equivalence is "easy".

Goč, Rampersad, R., Salimov, On the number of abelian bordered words, 2014

Complexity issues

Fischer and Rabin (1973) – beyond NP

There exists a constant $c > 0$ such that for every decision procedure (algorithm) $\mathtt{A}$ for Presburger arithmetic $\mathfrak{p}$, there exists an integer N so that for every $n > N$ there exists a sentence φ of length n for which $\mathtt{A}$ requires more than $2^{2^{cn}}$ computational steps to decide whether $\mathfrak{p} \models \varphi$. This statement applies also in the case of non-deterministic algorithms.

Starting with a N-state automaton, the subset construction could lead to

$$2^{2^{\cdot^{\cdot^{\cdot^{2^{p(N)}}}}}} \text{ states !}$$

a tower of exponentials depending on the number of quantifiers.

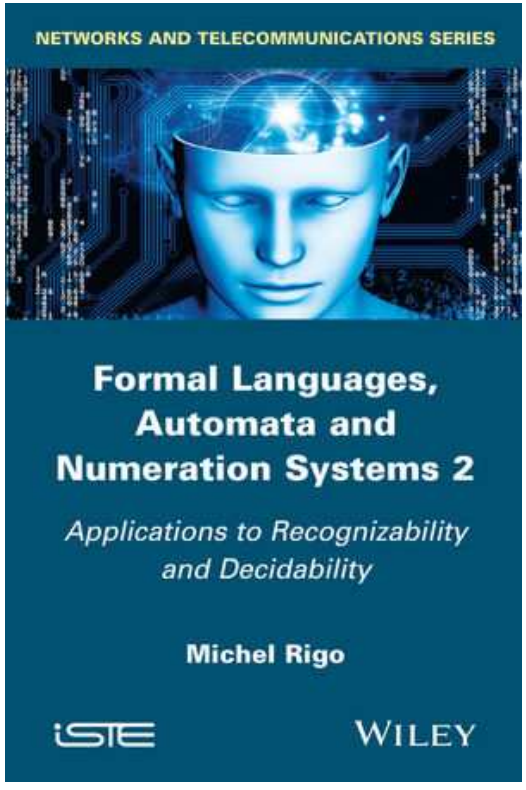
- F. Klaedtke, *Bounds on the automata size for Presburger arithmetic*, ACM Trans. Comput. Log. 9 (2008), Art. 11, 34.

Question: Study the (average) complexity with respect to formulae stemming from combinatorics on words.

Ultimate periodicity and automata

- Leroux: quadratic algorithm in $\mathbb{N}^d$ (LICS 2005)
- Marsault–Sakarovitch: quasi-linear algorithm l.s.d.f. in $\mathbb{N}$ (DLT 2013)

On n'est jamais aussi bien servi que par soi-même...
We are our own best advocates, as the saying goes



From 7th August 2017 to 11th August 2017
`www.cant.ulg.ac.be/dlt/`