Arithmetic subword complexity of automatic sequences, part I

Clemens Müllner

TU Wien

Thursday, February 29, 2024

Automatic Sequences

Definition (Automaton - DFA)

$$A = (Q, \Sigma = \{0, \dots, k-1\}, \delta, q_0, \tau)$$

Example (Thue-Morse sequence)

$$n = 22 = (10110)_2, \qquad t_{22} = 1$$

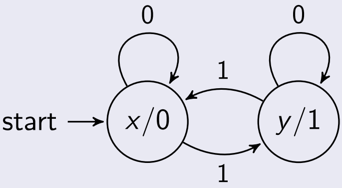
$$(t(n))_{n\geq 0} = 0110100110010110100101100110 1001\dots$$

Automatic Sequences

Definition (Automaton - DFA)

$$A = (Q, \Sigma = \{0, \ldots, k-1\}, \delta, q_0, \tau)$$

Example (Thue-Morse sequence)



$$n = 22 = (10110)_2, \qquad t_{22} = 1$$

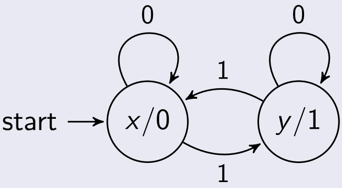
$$(t(n))_{n \geq 0} = 0110100110010110100101100110100 1\ldots$$

Automatic Sequences

Definition (Automaton - DFA)

$$A = (Q, \Sigma = \{0, \dots, k-1\}, \delta, q_0, \tau)$$

Example (Thue-Morse sequence)



$$n = 22 = (10110)_2, \qquad t_{22} = 1$$

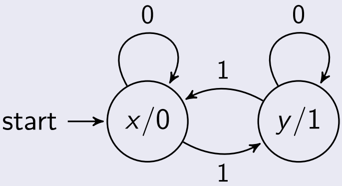
$$(t(n))_{n \geq 0} = 01101001100101101001011001101001\dots$$

Automatic Sequences

Definition (Automaton - DFA)

$$A = (Q, \Sigma = \{0, \ldots, k-1\}, \delta, q_0, \tau)$$

Example (Thue-Morse sequence)



$$n = 22 = (10110)_2, \qquad t_{22} = 1$$

$$(t(n))_{n \geq 0} = 0110100110010110100101100110100 1 \ldots$$

Different Points of View I

$$(a(n))_{n\geq 0} = 01101001100101101001011001101001\ldots$$

Automaton (Computer Science)

Substitution (Dynamics)

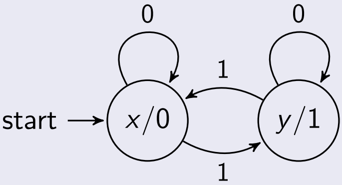
Coding of the fixpoint of a substitution:

$$x \to xy \qquad x \mapsto 0$$
$$y \to yx \qquad y \mapsto 1$$

Different Points of View I

$$(a(n))_{n\geq 0} = 01101001100101101001011001101001\ldots$$

Automaton (Computer Science)



Substitution (Dynamics)

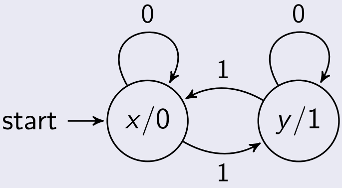
Coding of the fixpoint of a substitution:

$$x \to xy \qquad x \mapsto 0$$
$$y \to yx \qquad y \mapsto 1$$

Different Points of View I

$$(a(n))_{n\geq 0} = 01101001100101101001011001101001\ldots$$

Automaton (Computer Science)



Substitution (Dynamics)

Coding of the fixpoint of a substitution:

$$x \to xy \qquad x \mapsto 0$$
$$y \to yx \qquad y \mapsto 1$$

Different Points of View II

$$(t(n))_{n\geq 0} = 0110100110010110100101100110 1001\ldots$$

Formal Power Series (Algebra)

Algebraicity over $\mathbf{F}_q(X)$.

$$t(X) := \sum_{n\geq 0} a(n)X^n$$

$$X + (1+X)^2 t(X) + (1+X)^3 t(X)^2 = 0$$

Finite Kernel

The *k-kernel* of a sequence $a(n)$ is defined as

$$\{(a(nk^\lambda + r))_{n\geq 0} : \lambda \geq 0, 0 \leq r < k^\lambda\}.$$

Different Points of View II

$$(t(n))_{n\geq 0} = 01101001100101101001011001101001\ldots$$

Formal Power Series (Algebra)

Algebraicity over $\mathbf{F}_q(X)$.

$$t(X) := \sum_{n\geq 0} a(n)X^n$$

$$X + (1+X)^2 t(X) + (1+X)^3 t(X)^2 = 0$$

Finite Kernel

The *k-kernel* of a sequence $a(n)$ is defined as

$$\{(a(nk^\lambda + r))_{n\geq 0} : \lambda \geq 0, 0 \leq r < k^\lambda\}.$$

Different Points of View II

$(t(n))_{n\geq 0} = 01101001100101101001011001101001\ldots$

Formal Power Series (Algebra)

Algebraicity over $\mathbf{F}_q(X)$.

$$t(X) := \sum_{n\geq 0} a(n)X^n$$

$$X + (1+X)^2 t(X) + (1+X)^3 t(X)^2 = 0$$

Finite Kernel

The *k-kernel* of a sequence $a(n)$ is defined as

$$\{(a(nk^\lambda + r))_{n\geq 0} : \lambda \geq 0, 0 \leq r < k^\lambda\}.$$

Properties of Automatic Sequences

- Relatively easy to define (structured).
- Complex enough that interesting phenomena appear.
- Every subsequence $(a(xn+y))_{n\geq 0}$ along an arithmetic progression of an automatic sequence $\mathbf{a}$ is again automatic.

Properties of Automatic Sequences

- Relatively easy to define (structured).
- Complex enough that interesting phenomena appear.
- Every subsequence $(a(xn+y))_{n\geq 0}$ along an arithmetic progression of an automatic sequence **a** is again automatic.

Properties of Automatic Sequences

- Relatively easy to define (structured).
- Complex enough that interesting phenomena appear.
- Every subsequence $(a(xn + y))_{n\geq 0}$ along an arithmetic progression of an automatic sequence $\mathbf{a}$ is again automatic.

Synchronizing Automata

Definition (Synchronizing Automaton / Word)

$\exists \mathbf{w}_0 : \delta(q, \mathbf{w}_0) = x \quad \forall q.$

Example

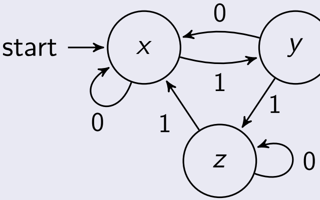
$\mathbf{w}_0 = 010.$

Synchronizing Automata

Definition (Synchronizing Automaton / Word)

$\exists \mathbf{w}_0 : \delta(q, \mathbf{w}_0) = x \quad \forall q.$

Example



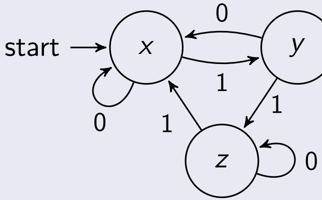
$\mathbf{w}_0 = 010.$

Synchronizing Automata

Definition (Synchronizing Automaton / Word)

$\exists \mathbf{w}_0 : \delta(q, \mathbf{w}_0) = x \quad \forall q.$

Example



$\mathbf{w}_0 = 010.$

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Synchronizing Automata

Key Property

- Any word $\mathbf{w}$ containing $\mathbf{w}_0$ is also synchronizing.
- Most words are synchronizing.
- $\mathbf{a}$ can be approximated by periodic sequences:
  Let λ be large. Most words of length λ are synchronizing.
  $a(n) = a(n \bmod k^\lambda)$ if $n \bmod k^\lambda$ is synchronizing.

"Usual" Strategy

- Understand the problem for periodic sequences.
- Transfer the result to synchronizing automatic sequences.

Subword Complexity

Let $\mathcal{A}$ be a finite alphabet and $\mathbf{u} = (u(n))_{n\in\mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$.

Definition (Subword Complexity)

The subword complexity of a sequence $\mathbf{u} \in \mathcal{A}^{\mathbb{N}}$ is defined by

$$p_{\mathbf{u}}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists k, (u(k), \ldots, u(k+L-1)) = \mathbf{w}\}.$$

$$p_{\mathbf{u}}(L) \leq |\mathcal{A}|^L$$

Subword complexity of automatic sequences

Let $\mathbf{a}$ be an automatic sequence. Then there exists $C > 0$ such that for all $L \in \mathbb{N}$

$$p_{\mathbf{a}}(L) \leq C \cdot L.$$

Subword Complexity

Let $\mathcal{A}$ be a finite alphabet and $\mathbf{u} = (u(n))_{n\in\mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$.

Definition (Subword Complexity)

The subword complexity of a sequence $\mathbf{u} \in \mathcal{A}^{\mathbb{N}}$ is defined by

$$p_{\mathbf{u}}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists k, (u(k), \ldots, u(k+L-1)) = \mathbf{w}\}.$$

$$p_{\mathbf{u}}(L) \leq |\mathcal{A}|^L$$

Subword complexity of automatic sequences

Let $\mathbf{a}$ be an automatic sequence. Then there exists $C > 0$ such that for all $L \in \mathbb{N}$

$$p_{\mathbf{a}}(L) \leq C \cdot L.$$

Subword Complexity

Let $\mathcal{A}$ be a finite alphabet and $\mathbf{u} = (u(n))_{n\in\mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$.

Definition (Subword Complexity)

The subword complexity of a sequence $\mathbf{u} \in \mathcal{A}^{\mathbb{N}}$ is defined by

$$p_{\mathbf{u}}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists k, (u(k), \ldots, u(k+L-1)) = \mathbf{w}\}.$$

$$p_{\mathbf{u}}(L) \leq |\mathcal{A}|^L$$

Subword complexity of automatic sequences

Let $\mathbf{a}$ be an automatic sequence. Then there exists $C > 0$ such that for all $L \in \mathbb{N}$

$$p_{\mathbf{a}}(L) \leq C \cdot L.$$

Subword Complexity

Let $\mathcal{A}$ be a finite alphabet and $\mathbf{u} = (u(n))_{n\in\mathbb{N}} \in \mathcal{A}^{\mathbb{N}}$.

Definition (Subword Complexity)

The subword complexity of a sequence $\mathbf{u} \in \mathcal{A}^{\mathbb{N}}$ is defined by

$$p_{\mathbf{u}}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists k, (u(k), \dots, u(k+L-1)) = \mathbf{w}\}.$$

$$p_{\mathbf{u}}(L) \leq |\mathcal{A}|^L$$

Subword complexity of automatic sequences

Let $\mathbf{a}$ be an automatic sequence. Then there exists $C > 0$ such that for all $L \in \mathbb{N}$

$$p_{\mathbf{a}}(L) \leq C \cdot L.$$

Arithmetic subword complexity

Definition (arithmetic subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{AP}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists n \geq 0, m \geq 1 : u(n+im) = \mathbf{w}(i) \text{ for } i = 0, \ldots, L-1\}.$$

Theorem (Avgustinovich, Fon-Der-Flaass and Frid; 2003)

- A certain class of invertible automatic sequences has maximal arithmetic subword complexity. (E.g. Thue-Morse sequence)
- Certain synchronizing automatic sequences have at most linear arithmetic subword complexity. (E.g. paperfolding sequence)

Arithmetic subword complexity

Definition (arithmetic subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{AP}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists n \geq 0, m \geq 1 : u(n+im) = \mathbf{w}(i) \text{ for } i = 0, \ldots, L-1\}.$$

Theorem (Avgustinovich, Fon-Der-Flaass and Frid; 2003)

- A certain class of invertible automatic sequences has maximal arithmetic subword complexity. (E.g. Thue-Morse sequence)
- Certain synchronizing automatic sequences have at most linear arithmetic subword complexity. (E.g. paperfolding sequence)

Arithmetic subword complexity

Definition (arithmetic subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{AP}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists n \geq 0, m \geq 1 : u(n+im) = \mathbf{w}(i) \text{ for } i = 0, \dots, L-1\}.$$

Theorem (Avgustinovich, Fon-Der-Flaass and Frid; 2003)

- A certain class of invertible automatic sequences has maximal arithmetic subword complexity. (E.g. Thue-Morse sequence)
- Certain synchronizing automatic sequences have at most linear arithmetic subword complexity. (E.g. paperfolding sequence)

Main Result

Definition (polynomial subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{\leq d}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists P \in \mathbb{Z}[x], P(\mathbb{N}) \subseteq \mathbb{N}, \deg P \leq d : u(P(i)) = \mathbf{w}(i) \text{ for } i = 0, \dots, L-1\}.$$

Theorem 1 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $d \geq 1$

$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(o(L)).$$

Basically the same proof: there exist $c > 0, \eta > 0$ such that

$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(cL^{1-\eta}).$$

Main Result

Definition (polynomial subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{\leq d}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists P \in \mathbb{Z}[x], P(\mathbb{N}) \subseteq \mathbb{N}, \deg P \leq d : u(P(i)) = \mathbf{w}(i) \text{ for } i = 0, \ldots, L-1\}.$$

Theorem 1 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $d \geq 1$

$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(o(L)).$$

Basically the same proof: there exist $c > 0, \eta > 0$ such that

$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(cL^{1-\eta}).$$

Main Result

Definition (polynomial subword complexity)

Let $\mathbf{u}$ be a sequence over a finite alphabet $\mathcal{A}$.

$$p_{\mathbf{u}}^{\leq d}(L) := \#\{\mathbf{w} \in \mathcal{A}^L : \exists P \in \mathbb{Z}[x], P(\mathbb{N}) \subseteq \mathbb{N}, \deg P \leq d : u(P(i)) = \mathbf{w}(i) \text{ for } i = 0, \ldots, L-1\}.$$

Theorem 1 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $d \geq 1$

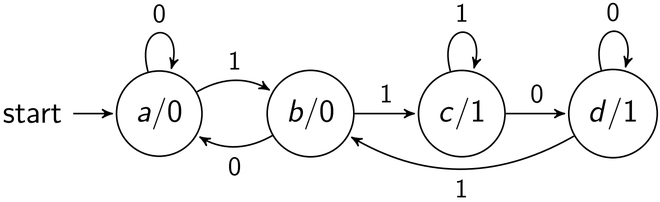
$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(o(L)).$$

Basically the same proof: there exist $c > 0, \eta > 0$ such that

$$p_{\mathbf{a}}^{\leq d}(L) \leq \exp(cL^{1-\eta}).$$

Numerical lower bounds

Paperfolding sequence:



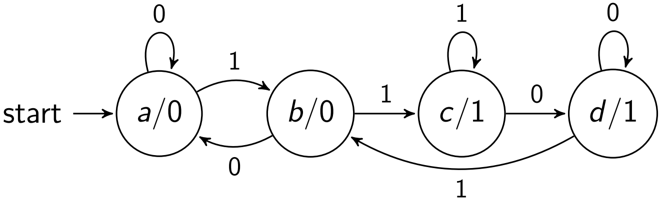
Numerical lower bounds for $p_{pf}^{\leq d}(L)$:

L	1	2	3	4	5	6	7	8	9
$p_{pf}^{\leq 1}$	2	4	8	16	24	32	44	52	64
$p_{pf}^{\leq 2}$	2	4	8	16	32	64	128	256	400

L	10	11	12	13	14	15	16	17	18
$p_{pf}^{\leq 1}$	76	86	96	106	116	124	132	140	148
$p_{pf}^{\leq 2}$	600	828	1102	1504	1952	2352	2784	3334	?

Numerical lower bounds

Paperfolding sequence:



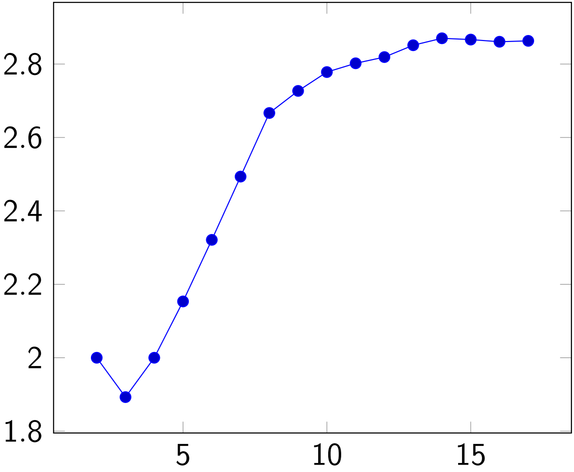
Numerical lower bounds for $p_{pf}^{\leq d}(L)$:

L	1	2	3	4	5	6	7	8	9
$p_{pf}^{\leq 1}$	2	4	8	16	24	32	44	52	64
$p_{pf}^{\leq 2}$	2	4	8	16	32	64	128	256	400

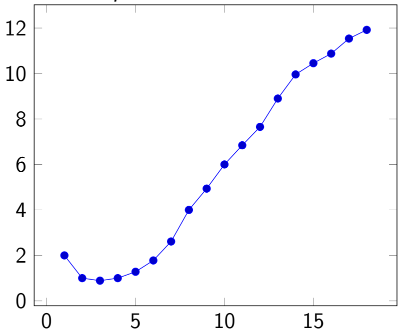
L	10	11	12	13	14	15	16	17	18
$p_{pf}^{\leq 1}$	76	86	96	106	116	124	132	140	148
$p_{pf}^{\leq 2}$	600	828	1102	1504	1952	2352	2784	3334	?

$p_{pf}^{\leq 2}$

Plot of $\log(p_{pf}^{\leq 2}(L))/\log(L)$.



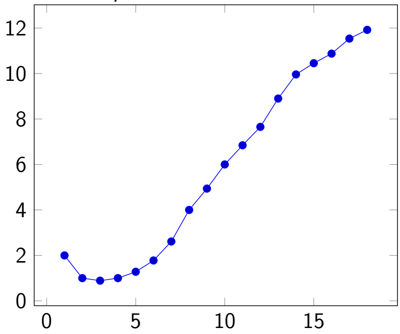
Plot of $p_{pf}^{\leq 2}(L))/L^2$.



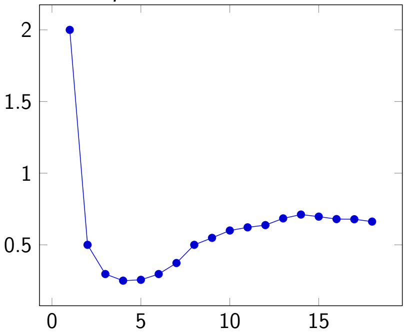
Plot of $p_{pf}^{\leq 2}(L))/L^3$.

$p_{pf}^{\le 2}$

Plot of $p_{pf}^{\le 2}(L))/L^2$.



Plot of $p_{pf}^{\le 2}(L))/L^3$.



Can we do better?

Why did we consider the paper-folding sequence?

The number of words of length 2^ℓ that are not synchronizing equals 2 for all $\ell \in \mathbb{N}$! (In general: $O(2^{\ell(1-\eta)})$.)

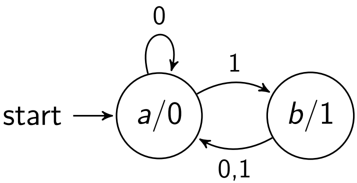
Consider the period-doubling sequence, where only 1 word $(1 \dots 1)$ is not synchronizing:

Can we do better?

Why did we consider the paper-folding sequence?
The number of words of length 2^ℓ that are not synchronizing equals 2 for all $\ell \in \mathbb{N}$! (In general: $O(2^{\ell(1-\eta)})$.)
Consider the period-doubling sequence, where only 1 word $(1 \ldots 1)$ is not synchronizing:

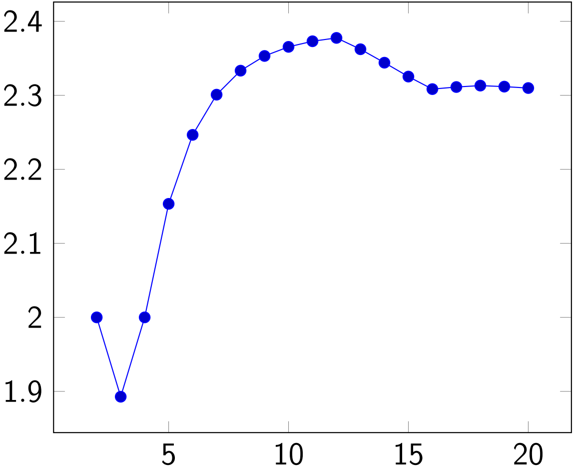
Can we do better?

Why did we consider the paper-folding sequence?
The number of words of length 2^ℓ that are not synchronizing equals 2 for all $\ell \in \mathbb{N}$! (In general: $O(2^{\ell(1-\eta)})$.)
Consider the period-doubling sequence, where only 1 word $(1 \ldots 1)$ is not synchronizing:



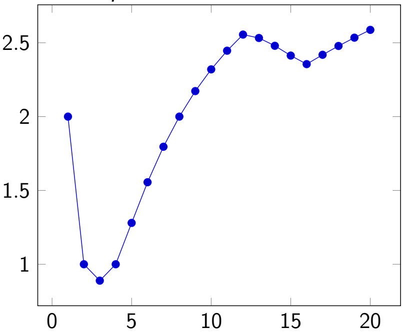
$p_{pd}^{\leq 2}$

Plot of $\log(p_{pd}^{\leq 2}(L))/\log(L)$.



$p_{pd}^{\leq 2}$

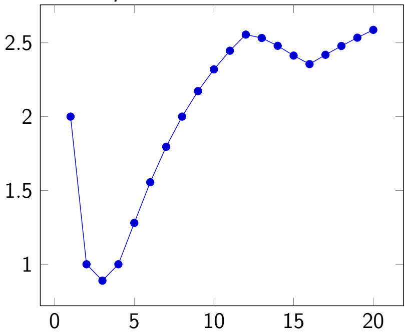
Plot of $p_{pd}^{\leq 2}(L))/L^2$.



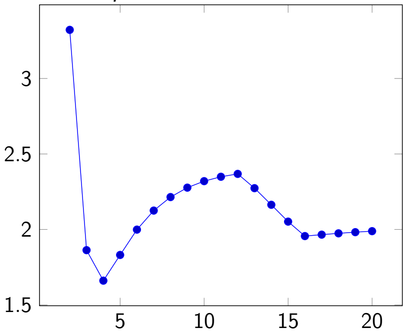
Plot of $p_{pd}^{\leq 2}(L))/(L^2 \log(L))$.

$p_{pd}^{\leq 2}$

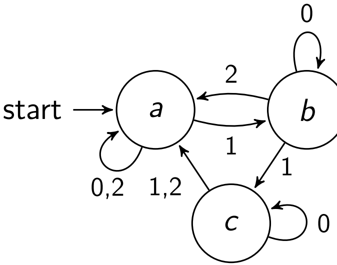
Plot of $p_{pd}^{\leq 2}(L))/L^2$.



Plot of $p_{pd}^{\leq 2}(L))/(L^2 \log(L))$.



Does $p_a^{\leq 1}$ always grow at most polynomially?

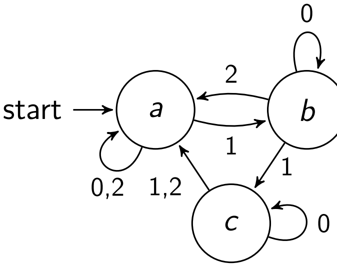


The words that are not synchronizing belong to $\{0,1\}^*$.

L	1	2	3	4	5	6	7	8	9	10
$p_a^{\leq 1}$	2	4	8	16	32	64	128	256	512	1 024

L	11	12	13	14	15	16	17	18	19	20
$p_a^{\leq 1}$	2 048	4 080	8 123	15 972	30 516	55 309	94 451	151 763	233 543	335 362

Does $p_a^{\leq 1}$ always grow at most polynomially?

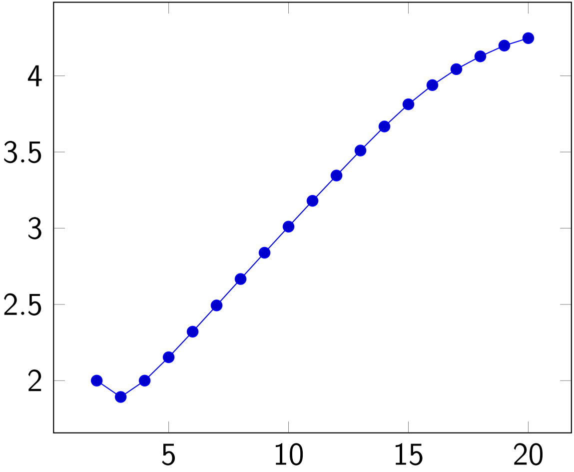


The words that are not synchronizing belong to $\{0, 1\}^*$.

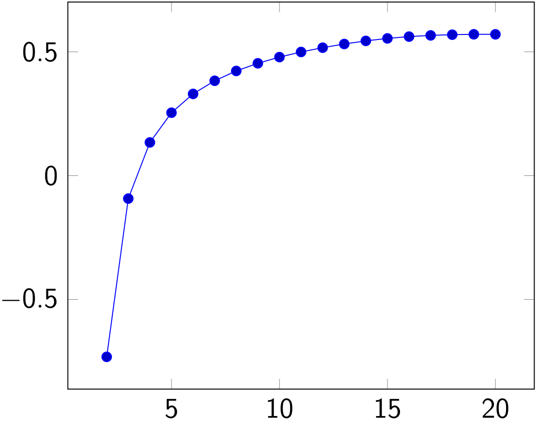
L	1	2	3	4	5	6	7	8	9	10
$p_a^{\leq 1}$	2	4	8	16	32	64	128	256	512	1 024

L	11	12	13	14	15	16	17	18	19	20
$p_a^{\leq 1}$	2 048	4 080	8 123	15 972	30 516	55 309	94 451	151 763	233 543	335 362

Plot of $\log(p_a^{\leq 1}(L))/\log(L)$.

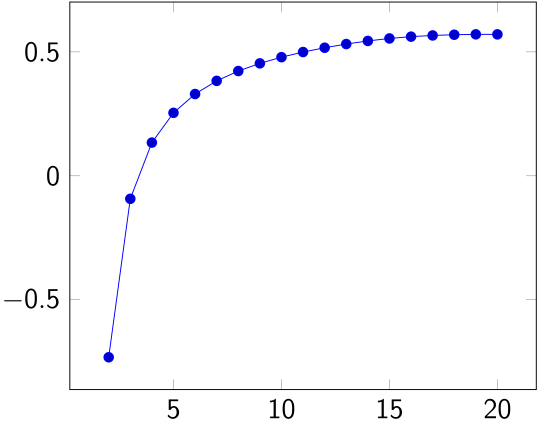


Plot of $\log(\log(p_a^{\leq 1}(L)))/\log(L)$.



This plot suggests that $p_a^{\leq 1}(L) \gg exp(L^{\eta})$ for some $\eta > 0.5$.

Plot of $\log(\log(p_a^{\leq 1}(L)))/\log(L)$.



This plot suggests that $p_a^{\leq 1}(L) \gg exp(L^\eta)$ for some $\eta > 0.5$.

Consequences of Theorem 1

Theorem 2 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $c > 0$ the subword complexity of $a(\lfloor n^c \rfloor)$ grows sub-exponentially $(\exp(o(L)))$.

Remark: The same result holds for any function f with "nice" derivatives.

Theorem 3 (Konieczny, M., 2024+)

Theorem 1 can be used to give upper bounds for $p_{\mathbf{a}}^{\leq d}$ for general automatic sequences $\mathbf{a}$.

Consequences of Theorem 1

Theorem 2 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $c > 0$ the subword complexity of $a(\lfloor n^c \rfloor)$ grows sub-exponentially $(\exp(o(L)))$.

Remark: The same result holds for any function f with "nice" derivatives.

Theorem 3 (Konieczny, M., 2024+)

Theorem 1 can be used to give upper bounds for $p_{a}^{\leq d}$ for general automatic sequences $\mathbf{a}$.

Consequences of Theorem 1

Theorem 2 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $c > 0$ the subword complexity of $a(\lfloor n^c \rfloor)$ grows sub-exponentially ($\exp(o(L))$).

Remark: The same result holds for any function f with "nice" derivatives.

Theorem 3 (Konieczny, M., 2024+)

Theorem 1 can be used to give upper bounds for $p_a^{\leq d}$ for general automatic sequences $\mathbf{a}$.

Background to Theorem 1 and 2

Theorem (Drmota, Mauduit, Rivat; 2019)

Let **t** denote the Thue-Morse sequence. Then

$$p_{t(n^2)}(L) = 2^L.$$

Actually, the Thue-Morse sequence is normal along the squares.

Theorem (M., 2018)

The same also holds for block-additive functions modulo m instead of the Thue-Morse sequence.

Theorem (M., Spiegelhofer, 2017)

Let **t** denote the Thue-Morse sequence and let $1 < c < \frac{3}{2}$. Then $(t(\lfloor n^c \rfloor))_{n\in\mathbb{N}}$ is normal.

Background to Theorem 1 and 2

Theorem (Drmota, Mauduit, Rivat; 2019)

Let **t** denote the Thue-Morse sequence. Then

$$p_{t(n^2)}(L) = 2^L.$$

Actually, the Thue-Morse sequence is normal along the squares.

Theorem (M., 2018)

The same also holds for block-additive functions modulo m instead of the Thue-Morse sequence.

Theorem (M., Spiegelhofer, 2017)

Let **t** denote the Thue-Morse sequence and let $1 < c < \frac{3}{2}$. Then $(t(\lfloor n^c \rfloor))_{n \in \mathbb{N}}$ is normal.

Background to Theorem 1 and 2

Theorem (Drmota, Mauduit, Rivat; 2019)

Let $\mathbf{t}$ denote the Thue-Morse sequence. Then

$$p_{t(n^2)}(L) = 2^L.$$

Actually, the Thue-Morse sequence is normal along the squares.

Theorem (M., 2018)

The same also holds for block-additive functions modulo m instead of the Thue-Morse sequence.

Theorem (M., Spiegelhofer, 2017)

Let $\mathbf{t}$ denote the Thue-Morse sequence and let $1 < c < \frac{3}{2}$. Then $(t(\lfloor n^c \rfloor))_{n \in \mathbb{N}}$ is normal.

Proof of Theorem 1

Naive approach

- Let f be a m-periodic function. Then $p_f^{\leq d}(L) \leq m^{d+1}$.
- Approximate $a(n)$ by a k^λ-periodic function $f(n)$.
- $a(n)$ and $f(n)$ agree on most residue classes modulo k^λ.
- Problem: P can hit the "bad" residue classes very often. (Trivial example: $P(x) = k^\lambda x + r$.)

Proof of Theorem 1

Naive approach

- Let f be a m-periodic function. Then $p_f^{\leq d}(L) \leq m^{d+1}$.
- Approximate $a(n)$ by a k^λ-periodic function $f(n)$.
- $a(n)$ and $f(n)$ agree on most residue classes modulo k^λ.
- Problem: P can hit the "bad" residue classes very often. (Trivial example: $P(x) = k^\lambda x + r$.)

Proof of Theorem 1

Naive approach

- Let f be a m-periodic function. Then $p_f^{\leq d}(L) \leq m^{d+1}$.
- Approximate $a(n)$ by a k^λ-periodic function $f(n)$.
- $a(n)$ and $f(n)$ agree on most residue classes modulo k^λ.
- Problem: P can hit the "bad" residue classes very often. (Trivial example: $P(x) = k^\lambda x + r$.)

Proof of Theorem 1

Naive approach

- Let f be a m-periodic function. Then $p_f^{\leq d}(L) \leq m^{d+1}$.
- Approximate $a(n)$ by a k^λ-periodic function $f(n)$.
- $a(n)$ and $f(n)$ agree on most residue classes modulo k^λ.
- Problem: P can hit the "bad" residue classes very often. (Trivial example: $P(x) = k^\lambda x + r$.)

Reductions

We study $(a(P(n)), a(P(n+1)), \dots, a(P(n+L-1)))$.

- Let $Q(\ell) = P(n+\ell)$. Study $(a(Q(0)), a(Q(1)), \dots, a(Q(L-1)))$.
- Avoid trivial problems:
  $Q(\ell) = k^{\lambda_0}\left(z'_d \ell^d + \dots + z'_1 \ell + z'_0\right) + r$.
  Using the kernel: $\exists b_i \in Ker_k(a)$ with $b_i(n) = a(nk^{\lambda_0} + r)$.
  $a(Q(\ell)) = b_i(z'_d \ell^d + \dots + z'_1 \ell + z'_0) = b_i(Q'(\ell))$.
- Remains to study $b_i(Q'(\ell))$ where some z'_i $(i \geq 1)$ is not divisible by k.

Reductions

We study $(a(P(n)), a(P(n+1)), \ldots, a(P(n+L-1)))$.

- Let $Q(\ell) = P(n+\ell)$. Study $(a(Q(0)), a(Q(1)), \ldots, a(Q(L-1)))$.
- Avoid trivial problems:
  $Q(\ell) = k^{\lambda_0}\left(z_d'\ell^d + \ldots + z_1'\ell + z_0'\right) + r$.
  Using the kernel: $\exists b_i \in Ker_k(a)$ with $b_i(n) = a(nk^{\lambda_0} + r)$.
  $a(Q(\ell)) = b_i(z_d'\ell^d + \ldots + z_1'\ell + z_0') = b_i(Q'(\ell))$.
- Remains to study $b_i(Q'(\ell))$ where some z_i' $(i \geq 1)$ is not divisible by k.

Reductions

We study $(a(P(n)), a(P(n+1)), \ldots, a(P(n+L-1)))$.

- Let $Q(\ell) = P(n+\ell)$. Study $(a(Q(0)), a(Q(1)), \ldots, a(Q(L-1)))$.
- Avoid trivial problems:
  $Q(\ell) = k^{\lambda_0}\left(z_d'\ell^d + \ldots + z_1'\ell + z_0'\right) + r.$
  Using the kernel: $\exists b_i \in Ker_k(a)$ with $b_i(n) = a(nk^{\lambda_0} + r)$.
  $a(Q(\ell)) = b_i(z_d'\ell^d + \ldots + z_1'\ell + z_0') = b_i(Q'(\ell))$.
- Remains to study $b_i(Q'(\ell))$ where some z_i' $(i \geq 1)$ is not divisible by k.

Reductions

We study $(a(P(n)), a(P(n+1)), \ldots, a(P(n+L-1)))$.

- Let $Q(\ell) = P(n+\ell)$. Study $(a(Q(0)), a(Q(1)), \ldots, a(Q(L-1)))$.
- Avoid trivial problems:
  $Q(\ell) = k^{\lambda_0}\left(z'_d \ell^d + \ldots + z'_1 \ell + z'_0\right) + r$.
  Using the kernel: $\exists b_i \in Ker_k(a)$ with $b_i(n) = a(nk^{\lambda_0} + r)$.
  $a(Q(\ell)) = b_i(z'_d \ell^d + \ldots + z'_1 \ell + z'_0) = b_i(Q'(\ell))$.
- Remains to study $b_i(Q'(\ell))$ where some z'_i $(i \geq 1)$ is not divisible by k.

Reductions

We study $(a(P(n)), a(P(n+1)), \ldots, a(P(n+L-1)))$.

- Let $Q(\ell) = P(n+\ell)$. Study $(a(Q(0)), a(Q(1)), \ldots, a(Q(L-1)))$.
- Avoid trivial problems:
  $Q(\ell) = k^{\lambda_0}\left(z'_d \ell^d + \ldots + z'_1 \ell + z'_0\right) + r$.
  Using the kernel: $\exists b_i \in Ker_k(a)$ with $b_i(n) = a(nk^{\lambda_0} + r)$.
  $a(Q(\ell)) = b_i(z'_d \ell^d + \ldots + z'_1 \ell + z'_0) = b_i(Q'(\ell))$.
- Remains to study $b_i(Q'(\ell))$ where some z'_i $(i \geq 1)$ is not divisible by k.

Example

Consider $Q'(\ell) = 5 \cdot 3^4 \cdot \ell$ modulo 6^4.

$$(Q'(0))_6 = 00000 \qquad (Q'(8))_6 = 23000$$
$$(Q'(1))_6 = 01513 \qquad (Q'(9))_6 = 24513$$
$$(Q'(2))_6 = 03430 \qquad (Q'(10))_6 = 30430$$
$$(Q'(3))_6 = 05343 \qquad (Q'(11))_6 = 32343$$
$$(Q'(4))_6 = 11300 \qquad (Q'(12))_6 = 34300$$
$$(Q'(5))_6 = 13213 \qquad (Q'(13))_6 = 40213$$
$$(Q'(6))_6 = 15130 \qquad (Q'(14))_6 = 42130$$
$$(Q'(7))_6 = 21043 \qquad (Q'(15))_6 = 44043$$

Example

Consider $Q'(\ell) = 5 \cdot 3^4 \cdot \ell$ modulo 6^4.

$$(Q'(0))_6 = 00000 \qquad (Q'(8))_6 = 23000$$
$$(Q'(1))_6 = 01513 \qquad (Q'(9))_6 = 24513$$
$$(Q'(2))_6 = 03430 \qquad (Q'(10))_6 = 30430$$
$$(Q'(3))_6 = 05343 \qquad (Q'(11))_6 = 32343$$
$$(Q'(4))_6 = 11300 \qquad (Q'(12))_6 = 34300$$
$$(Q'(5))_6 = 13213 \qquad (Q'(13))_6 = 40213$$
$$(Q'(6))_6 = 15130 \qquad (Q'(14))_6 = 42130$$
$$(Q'(7))_6 = 21043 \qquad (Q'(15))_6 = 44043$$

Example

Consider $Q'(\ell) = 5 \cdot 3^4 \cdot \ell$ modulo 6^4.

$$(Q'(0))_6 = 00000 \qquad (Q'(8))_6 = 23000$$
$$(Q'(1))_6 = 01513 \qquad (Q'(9))_6 = 24513$$
$$(Q'(2))_6 = 03430 \qquad (Q'(10))_6 = 30430$$
$$(Q'(3))_6 = 05343 \qquad (Q'(11))_6 = 32343$$
$$(Q'(4))_6 = 11300 \qquad (Q'(12))_6 = 34300$$
$$(Q'(5))_6 = 13213 \qquad (Q'(13))_6 = 40213$$
$$(Q'(6))_6 = 15130 \qquad (Q'(14))_6 = 42130$$
$$(Q'(7))_6 = 21043 \qquad (Q'(15))_6 = 44043$$

New Approach

- Problem: We still only hit few residue classes modulo k^λ. (E.g. $Q'(\ell) = 5 \cdot 3^\lambda \cdot \ell \bmod 6^\lambda$.)
- "low" digits: $Q'(\ell)$ might still not equidistribute $\bmod k^\lambda$.
- "high" digits work: $\exists \varepsilon(k) > 0$ such that for any $\mathbf{w} \in \mathcal{A}^{\varepsilon\lambda}$ we have $\#\{\ell < k^\lambda : (Q'(\ell) \bmod k^\lambda)_k \text{ starts with } \mathbf{w}\} \approx k^{\lambda(1-\varepsilon)}$.

New Approach

- Problem: We still only hit few residue classes modulo k^λ. (E.g. $Q'(\ell) = 5 \cdot 3^\lambda \cdot \ell \bmod 6^\lambda$.)
- "low" digits: $Q'(\ell)$ might still not equidistribute $\bmod k^\lambda$.
- "high" digits work: $\exists \varepsilon(k) > 0$ such that for any $\mathbf{w} \in \mathcal{A}^{\varepsilon\lambda}$ we have $\#\{\ell < k^\lambda : (Q'(\ell) \bmod k^\lambda)_k \text{ starts with } \mathbf{w}\} \approx k^{\lambda(1-\varepsilon)}$.

New Approach

- Problem: We still only hit few residue classes modulo k^λ. (E.g. $Q'(\ell) = 5 \cdot 3^\lambda \cdot \ell \bmod 6^\lambda$.)
- "low" digits: $Q'(\ell)$ might still not equidistribute $\bmod k^\lambda$.
- "high" digits work: $\exists \varepsilon(k) > 0$ such that for any $\mathbf{w} \in \mathcal{A}^{\varepsilon\lambda}$ we have $\#\{\ell < k^\lambda : (Q'(\ell) \bmod k^\lambda)_k \text{ starts with } \mathbf{w}\} \approx k^{\lambda(1-\varepsilon)}$.

Equidistribution of high digits

- Detection of digits: The digits of ℓ in base k between positions μ and λ coincide with the digits of $m < k^{\lambda-\mu}$ iff

$$\left\{\frac{\ell}{k^\lambda}\right\} \in \left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}}\right).$$

- Expand the indicator function into a Fourier series.

$$\sum_{\ell<k^\lambda} \mathbf{1}_{\left[\left\{\frac{Q'(\ell)}{k^\lambda}\right\}\in\left[\frac{m}{k^{\lambda-\mu}},\frac{m+1}{k^{\lambda-\mu}}\right)\right]} \approx \sum_{|h|<H} c_h \sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right).$$

- Use classical estimates for

$$\sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right),$$

depending on $\min_{1\le j\le d} \gcd(z_j', k^\lambda)$.

Equidistribution of high digits

- Detection of digits: The digits of ℓ in base k between positions μ and λ coincide with the digits of $m < k^{\lambda-\mu}$ iff

$$\left\{ \frac{\ell}{k^\lambda} \right\} \in \left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}} \right).$$

- Expand the indicator function into a Fourier series.

$$\sum_{\ell < k^\lambda} \mathbf{1}_{\left[\left\{\frac{Q'(\ell)}{k^\lambda}\right\} \in \left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}}\right)\right]} \approx \sum_{|h|<H} c_h \sum_{\ell < k^\lambda} \mathrm{e}\left(\frac{h \cdot Q'(\ell)}{k^\lambda}\right).$$

- Use classical estimates for

$$\sum_{\ell < k^\lambda} \mathrm{e}\left(\frac{h \cdot Q'(\ell)}{k^\lambda}\right),$$

depending on $\min_{1 \leq j \leq d} \gcd(z_j', k^\lambda)$.

Equidistribution of high digits

- Detection of digits: The digits of ℓ in base k between positions μ and λ coincide with the digits of $m < k^{\lambda-\mu}$ iff

$$\left\{\frac{\ell}{k^\lambda}\right\} \in \left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}}\right).$$

- Expand the indicator function into a Fourier series.

$$\sum_{\ell<k^\lambda} \mathbf{1}_{\left[\left\{\frac{Q'(\ell)}{k^\lambda}\right\}\in\left[\frac{m}{k^{\lambda-\mu}},\frac{m+1}{k^{\lambda-\mu}}\right)\right]} \approx \sum_{|h|<H} c_h \sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right).$$

- Use classical estimates for

$$\sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right),$$

depending on $\min_{1\leq j\leq d} \gcd(z_j', k^\lambda)$.

Equidistribution of high digits

- Detection of digits: The digits of ℓ in base k between positions μ and λ coincide with the digits of $m < k^{\lambda-\mu}$ iff

$$\left\{\frac{\ell}{k^\lambda}\right\} \in \left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}}\right).$$

- Expand the indicator function into a Fourier series.

$$\sum_{\ell<k^\lambda} \mathbf{1}_{\left[\left\{\frac{Q'(\ell)}{k^\lambda}\right\}\in\left[\frac{m}{k^{\lambda-\mu}}, \frac{m+1}{k^{\lambda-\mu}}\right)\right]} \approx \sum_{|h|<H} c_h \sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right).$$

- Use classical estimates for

$$\sum_{\ell<k^\lambda} \mathrm{e}\left(\frac{h\cdot Q'(\ell)}{k^\lambda}\right),$$

depending on $\min_{1\le j\le d} \gcd(z_j', k^\lambda)$.

Putting everything together

- Approximate $b_i(n)$ with a k^λ-periodic function $f(n)$.
- $b_i(Q'(\ell)) \neq f(Q'(\ell))$ only when $Q'(\ell) \bmod k^\lambda$ is not synchronizing.
- This happens rarely ($\ll Lk^{-\varepsilon\lambda}$).
- $p_f^{\leq d}(L) \leq \left(k^\lambda\right)^{d+1}$.
- (Optional: optimize λ as a function of L.)

Putting everything together

- Approximate $b_i(n)$ with a k^λ-periodic function $f(n)$.
- $b_i(Q'(\ell)) \neq f(Q'(\ell))$ only when $Q'(\ell)$ mod k^λ is not synchronizing.
- This happens rarely ($\ll Lk^{-\varepsilon\lambda}$).
- $p_f^{\leq d}(L) \leq (k^\lambda)^{d+1}$.
- (Optional: optimize λ as a function of L.)

Putting everything together

- Approximate $b_i(n)$ with a k^λ-periodic function $f(n)$.
- $b_i(Q'(\ell)) \neq f(Q'(\ell))$ only when $Q'(\ell) \bmod k^\lambda$ is not synchronizing.
- This happens rarely ($\ll Lk^{-\varepsilon\lambda}$).
- $p_f^{\leq d}(L) \leq \left(k^\lambda\right)^{d+1}$.
- (Optional: optimize λ as a function of L.)

Putting everything together

- Approximate $b_i(n)$ with a k^λ-periodic function $f(n)$.
- $b_i(Q'(\ell)) \neq f(Q'(\ell))$ only when $Q'(\ell) \bmod k^\lambda$ is not synchronizing.
- This happens rarely ($\ll Lk^{-\varepsilon\lambda}$).
- $p_f^{\leq d}(L) \leq \left(k^\lambda\right)^{d+1}$.
- (Optional: optimize λ as a function of L.)

Putting everything together

- Approximate $b_i(n)$ with a k^λ-periodic function $f(n)$.
- $b_i(Q'(\ell)) \neq f(Q'(\ell))$ only when $Q'(\ell)$ mod k^λ is not synchronizing.
- This happens rarely ($\ll Lk^{-\varepsilon\lambda}$).
- $p_f^{\leq d}(L) \leq \left(k^\lambda\right)^{d+1}$.
- (Optional: optimize λ as a function of L.)

Proof of Theorem 2

Theorem 2 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $c > 0$ the subword complexity of $a(\lfloor n^c \rfloor)$ grows sub-exponentially $(\exp(o(L)))$.

Connection to Theorem 1

We use Taylor expansion to write

$$(n+\ell)^c = \underbrace{\sum_{t=0}^{d} A_t^{(n)} \ell^t}_{P^{(n)}(\ell)} + g_\ell^{(n)}.$$

Proof of Theorem 2

Theorem 2 (Deshouillers, Drmota, M., Shubin, Spiegelhofer; 2024+)

Let $a(n)$ be a synchronizing automatic sequence. Then for any $c > 0$ the subword complexity of $a(\lfloor n^c \rfloor)$ grows sub-exponentially $(\exp(o(L)))$.

Connection to Theorem 1

We use Taylor expansion to write

$$(n+\ell)^c = \underbrace{\sum_{t=0}^{d} A_t^{(n)} \ell^t}_{P^{(n)}(\ell)} + g_\ell^{(n)}.$$

Periodic case (no error term)

$$\lfloor P^{(n)}(\ell) \rfloor \equiv u_\ell \bmod m \Leftrightarrow \left\{ \frac{P^{(n)}(\ell)}{m} \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \left\{ \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \exists z \leq (d+1)L^d :$$

$$\sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \in \left[z + \frac{u_\ell}{m}, z + \frac{u_\ell + 1}{m} \right).$$

Fix ℓ and treat $x_t = \left\{ \frac{A_t^{(n)}}{m} \right\}$ as variables.

Periodic case (no error term)

$$\lfloor P^{(n)}(\ell) \rfloor \equiv u_\ell \bmod m \Leftrightarrow \left\{ \frac{P^{(n)}(\ell)}{m} \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \left\{ \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \exists z \leq (d+1)L^d : \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \in \left[z + \frac{u_\ell}{m}, z + \frac{u_\ell + 1}{m} \right).$$

Fix ℓ and treat $x_t = \left\{ \frac{A_t^{(n)}}{m} \right\}$ as variables.

Periodic case (no error term)

$$\lfloor P^{(n)}(\ell) \rfloor \equiv u_\ell \bmod m \Leftrightarrow \left\{ \frac{P^{(n)}(\ell)}{m} \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \left\{ \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \exists z \leq (d+1)L^d :$$

$$\sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \in \left[z + \frac{u_\ell}{m}, z + \frac{u_\ell + 1}{m} \right).$$

Fix ℓ and treat $x_t = \left\{ \frac{A_t^{(n)}}{m} \right\}$ as variables.

Periodic case (no error term)

$$\lfloor P^{(n)}(\ell) \rfloor \equiv u_\ell \bmod m \Leftrightarrow \left\{ \frac{P^{(n)}(\ell)}{m} \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \left\{ \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$

$$\Leftrightarrow \exists z \leq (d+1)L^d :$$

$$\sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \in \left[z + \frac{u_\ell}{m}, z + \frac{u_\ell + 1}{m} \right).$$

Fix ℓ and treat $x_t = \left\{ \frac{A_t^{(n)}}{m} \right\}$ as variables.

Periodic case (no error term)

$$\lfloor P^{(n)}(\ell) \rfloor \equiv u_\ell \bmod m \Leftrightarrow \left\{ \frac{P^{(n)}(\ell)}{m} \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$
$$\Leftrightarrow \left\{ \sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \right\} \in \left[\frac{u_\ell}{m}, \frac{u_\ell + 1}{m} \right)$$
$$\Leftrightarrow \exists z \leq (d+1)L^d :$$
$$\sum_{t=0}^{d} \left\{ \frac{A_t^{(n)}}{m} \right\} \ell^t \in \left[z + \frac{u_\ell}{m}, z + \frac{u_\ell + 1}{m} \right).$$

Fix ℓ and treat $x_t = \left\{ \frac{A_t^{(n)}}{m} \right\}$ as variables.

Periodic case (no error term)

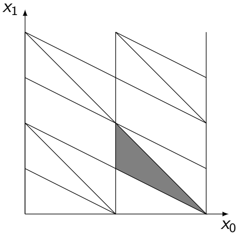
$\sum_{t=0}^{d} x_t \ell^t = z + \frac{u_\ell}{m}$ is a hyperplane.

Example:

The gray area corresponds to the intersection of the strips $0.5 < x_0 < 1$, $0.5 < x_0 + x_1 < 1$ and $1 < x_0 + 2x_1 < 1.5$.

Periodic case (no error term)

$\sum_{t=0}^{d} x_t \ell^t = z + \frac{u_\ell}{m}$ is a hyperplane.
Example:



The gray area corresponds to the intersection of the strips $0.5 < x_0 < 1$, $0.5 < x_0 + x_1 < 1$ and $1 < x_0 + 2x_1 < 1.5$.

Periodic case (no error term)

- There are at most mL^{d+2} hyperplanes.
- There are at most $\sum_{i=0}^{d+1} \binom{mL^{d+2}}{i} \ll_d m^{d+1} L^{(d+1)(d+2)}$ regions.
- We have uniformly in m and L,

$$\# \left\{ \left(\lfloor P^{(n)}(0) \rfloor \bmod m, \ldots, \lfloor P^{(n)}(L-1) \rfloor \bmod m \right) : n \geq 0 \right\} \ll_d m^{d+1} L^{(d+1)(d+2)}.$$

Periodic case (no error term)

- There are at most mL^{d+2} hyperplanes.
- There are at most $\sum_{i=0}^{d+1}\binom{mL^{d+2}}{i} \ll_d m^{d+1}L^{(d+1)(d+2)}$ regions.
- We have uniformly in m and L,

$$\#\left\{\left(\lfloor P^{(n)}(0)\rfloor \bmod m, \ldots, \lfloor P^{(n)}(L-1)\rfloor \bmod m\right) : n \geq 0\right\}$$
$$\ll_d m^{d+1}L^{(d+1)(d+2)}.$$

Periodic case (no error term)

- There are at most mL^{d+2} hyperplanes.
- There are at most $\sum_{i=0}^{d+1} \binom{mL^{d+2}}{i} \ll_d m^{d+1} L^{(d+1)(d+2)}$ regions.
- We have uniformly in m and L,

$$\# \left\{ \left(\lfloor P^{(n)}(0) \rfloor \bmod m, \ldots, \lfloor P^{(n)}(L-1) \rfloor \bmod m \right) : n \geq 0 \right\} \\ \ll_d m^{d+1} L^{(d+1)(d+2)}.$$

Conclusion

- Synchronizing automatic sequences are easier to treat than "uniform" automatic sequences. (We can treat higher degrees.)
- However, questions about subword complexity are still difficult!

Open Problem

Can we prove for some non-periodic automatic sequences that $p_a^{\leq d}(L)$ grows at most polynomially for all $d \in \mathbb{N}$?

Thank you for your attention!

Conclusion

- Synchronizing automatic sequences are easier to treat than "uniform" automatic sequences. (We can treat higher degrees.)
- However, questions about subword complexity are still difficult!

Open Problem

Can we prove for some non-periodic automatic sequences that $p_a^{\leq d}(L)$ grows at most polynomially for all $d \in \mathbb{N}$?

Thank you for your attention!

Conclusion

- Synchronizing automatic sequences are easier to treat than "uniform" automatic sequences. (We can treat higher degrees.)
- However, questions about subword complexity are still difficult!

Open Problem

Can we prove for some non-periodic automatic sequences that $p_a^{\leq d}(L)$ grows at most polynomially for all $d \in \mathbb{N}$?

Thank you for your attention!

Conclusion

- Synchronizing automatic sequences are easier to treat than "uniform" automatic sequences. (We can treat higher degrees.)
- However, questions about subword complexity are still difficult!

Open Problem

Can we prove for some non-periodic automatic sequences that $p_a^{\leq d}(L)$ grows at most polynomially for all $d \in \mathbb{N}$?

Thank you for your attention!