Combinatorics on Words: An Introduction

Jeffrey Shallit
School of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1
Canada
shallit@cs.uwaterloo.ca
http://www.cs.uwaterloo.ca/~shallit

The Main Themes

- periodicity

The Main Themes

- periodicity
- patterns and pattern avoidance

The Main Themes

- periodicity
- patterns and pattern avoidance
- equations in words

The Main Themes

- periodicity
- patterns and pattern avoidance
- equations in words
- infinite words and their properties

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

word - a finite or infinite list of symbols chosen from Σ

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

word - a finite or infinite list of symbols chosen from Σ

Σ^* - set of all finite words

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

word - a finite or infinite list of symbols chosen from Σ

Σ^* - set of all finite words

Σ^+ - set of all finite nonempty words

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

word - a finite or infinite list of symbols chosen from Σ

Σ^* - set of all finite words

Σ^+ - set of all finite nonempty words

Σ^ω - set of all (right-)infinite words

Some notation

Σ - a finite nonempty set of symbols - the *alphabet*

word - a finite or infinite list of symbols chosen from Σ

Σ^* - set of all finite words

Σ^+ - set of all finite nonempty words

Σ^ω - set of all (right-)infinite words

$\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$

More notation

the empty word: ϵ

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i, \;\; w[i..j] := a_i a_{i+1} \cdots a_j$

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i$, $w[i..j] := a_i a_{i+1} \cdots a_j$

$\mathbf{w} = a_0 a_1 a_2 \cdots$

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i, \;\; w[i..j] := a_i a_{i+1} \cdots a_j$

$\mathbf{w} = a_0 a_1 a_2 \cdots$

$x^\omega = xxx \cdots$

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i, \;\; w[i..j] := a_i a_{i+1} \cdots a_j$

$\mathbf{w} = a_0 a_1 a_2 \cdots$

$x^\omega = xxx \cdots$

ultimately periodic: $\mathbf{z} = xy^\omega$

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i, \;\; w[i..j] := a_i a_{i+1} \cdots a_j$

$\mathbf{w} = a_0 a_1 a_2 \cdots$

$x^\omega = xxx \cdots$

ultimately periodic: $\mathbf{z} = xy^\omega$

Operations: concatenation, raising to powers $x^n = \overbrace{xx \cdots x}^{n}$, $x^0 = \epsilon$, reversal x^R

More notation

the empty word: ϵ

$w = a_1 a_2 \cdots a_n$

$w[i] := a_i, \;\; w[i..j] := a_i a_{i+1} \cdots a_j$

$\mathbf{w} = a_0 a_1 a_2 \cdots$

$x^\omega = xxx \cdots$

ultimately periodic: $\mathbf{z} = xy^\omega$

Operations: concatenation, raising to powers $x^n = \overbrace{xx \cdots x}^{n}$, $x^0 = \epsilon$, reversal x^R

prefix, suffix, factor, subword

Algebraic framework

- semigroup: concatenation is multiplication, associative

Algebraic framework

- semigroup: concatenation is multiplication, associative
- monoid: semigroup + identity element (ϵ)

Algebraic framework

- semigroup: concatenation is multiplication, associative
- monoid: semigroup + identity element (ϵ)
- free monoid: no relations among elements

Algebraic framework

- semigroup: concatenation is multiplication, associative
- monoid: semigroup + identity element (ϵ)
- free monoid: no relations among elements
- group: add inverses of elements a^{-1}

Periodicity

words - fundamentally noncommutative

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

When do words commute?

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

When do words commute?
Here are two words that “almost” commute:

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

When do words commute?

Here are two words that "almost" commute:

$$w = 01010 \text{ and } x = 01011010$$

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

When do words commute?
Here are two words that "almost" commute:

$$w = 01010 \text{ and } x = 01011010$$

$$wx \quad = \quad 0101001011010$$

Periodicity

words - fundamentally noncommutative

$$\texttt{casebook} \neq \texttt{bookcase}$$

When do words commute?
Here are two words that "almost" commute:

$$w = 01010 \text{ and } x = 01011010$$

$$\begin{aligned} wx &= 0101001011010 \\ xw &= 0101101001010 \end{aligned}$$

By the way, this raises the question: can the Hamming distance between wx and xw be 1? It can't; there is a one-line proof.

What are the solutions to $x^2 = y^3$ in words?

- Over $\mathbb{N}$, $x^2 = y^3$ iff x is a cube and y is a square

What are the solutions to $x^2 = y^3$ in words?

- Over $\mathbb{N}$, $x^2 = y^3$ iff x is a cube and y is a square

- Suggests what solutions of $x^i = y^j$ will look like over words

What are the solutions to $x^2 = y^3$ in words?

- Over $\mathbb{N}$, $x^2 = y^3$ iff x is a cube and y is a square

- Suggests what solutions of $x^i = y^j$ will look like over words

First Theorem of Lyndon-Schützenberger

Theorem

Let $x, y \in \Sigma^+$. *Then the following three conditions are equivalent:*

First Theorem of Lyndon-Schützenberger

Theorem

Let $x, y \in \Sigma^+$. Then the following three conditions are equivalent:
(1) $xy = yx$;

First Theorem of Lyndon-Schützenberger

Theorem

Let $x, y \in \Sigma^+$. Then the following three conditions are equivalent:
(1) $xy = yx$;
(2) There exist $z \in \Sigma^+$ and integers $k, l > 0$ such that $x = z^k$ and $y = z^l$;

First Theorem of Lyndon-Schützenberger

Theorem

Let $x, y \in \Sigma^+$. Then the following three conditions are equivalent:
(1) $xy = yx$;
(2) There exist $z \in \Sigma^+$ and integers $k, l > 0$ such that $x = z^k$ and $y = z^l$;
(3) There exist integers $i, j > 0$ such that $x^i = y^j$.

Second Theorem of Lyndon-Schützenberger

Under what conditions can a string have a nontrivial proper prefix and suffix that are identical?

Second Theorem of Lyndon-Schützenberger

Under what conditions can a string have a nontrivial proper prefix and suffix that are identical?

Examples in English: `reader` — begins and ends with `r`

Second Theorem of Lyndon-Schützenberger

Under what conditions can a string have a nontrivial proper prefix and suffix that are identical?

Examples in English: `reader` — begins and ends with `r`

`alfalfa` — which begins and ends with `alfa`

Second Theorem of Lyndon-Schützenberger

Under what conditions can a string have a nontrivial proper prefix and suffix that are identical?

Examples in English: `reader` — begins and ends with `r`

`alfalfa` — which begins and ends with `alfa`

The answer is given by the following theorem.

Second Theorem of Lyndon-Schützenberger

Under what conditions can a string have a nontrivial proper prefix and suffix that are identical?

Examples in English: `reader` — begins and ends with `r`

`alfalfa` — which begins and ends with `alfa`

The answer is given by the following theorem.

Theorem

Let $x, y, z \in \Sigma^+$. Then $xy = yz$ if and only if there exist $u \in \Sigma^+$, $v \in \Sigma^*$, and an integer $e \geq 0$ such that $x = uv$, $z = vu$, and $y = (uv)^e u = u(vu)^e$.

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

A nonpower is called *primitive*.

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

A nonpower is called *primitive*.

Every nonempty word can be written uniquely in the form x^k where x is primitive and $k \geq 1$.

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

A nonpower is called *primitive*.

Every nonempty word can be written uniquely in the form x^k where x is primitive and $k \geq 1$.

Enumeration: there are exactly

$$\sum_{d \mid n} \mu(d) k^{n/d}$$

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

A nonpower is called *primitive*.

Every nonempty word can be written uniquely in the form x^k where x is primitive and $k \geq 1$.

Enumeration: there are exactly

$$\sum_{d \mid n} \mu(d) k^{n/d}$$

primitive words of length n over a k-letter alphabet. Here μ is the Möbius function and the sum is over the divisors of n.

Primitive words

We say a word x is a *power* if it can be expressed as $x = y^n$ for some $y \neq \epsilon$, $n \geq 2$.

A nonpower is called *primitive*.

Every nonempty word can be written uniquely in the form x^k where x is primitive and $k \geq 1$.

Enumeration: there are exactly

$$\sum_{d \mid n} \mu(d) k^{n/d}$$

primitive words of length n over a k-letter alphabet. Here μ is the Möbius function and the sum is over the divisors of n.

Open question: is the set of primitive binary words a CFL?

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

For example, the English word `enlist` is a conjugate of `listen`.

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

For example, the English word `enlist` is a conjugate of `listen`.

A conjugate of a k-th power is a k-th power of a conjugate.

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

For example, the English word `enlist` is a conjugate of `listen`.

A conjugate of a k-th power is a k-th power of a conjugate.

Every primitive word has an unbordered conjugate.

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

For example, the English word `enlist` is a conjugate of `listen`.

A conjugate of a k-th power is a k-th power of a conjugate.

Every primitive word has an unbordered conjugate.

Lyndon word: lexicographically least among all its conjugates

Conjugates

A word w is a *conjugate* of a word x if w can be obtained from x by cyclically shifting the letters.

For example, the English word `enlist` is a conjugate of `listen`.

A conjugate of a k-th power is a k-th power of a conjugate.

Every primitive word has an unbordered conjugate.

Lyndon word: lexicographically least among all its conjugates

Theorem: Every finite word has a unique factorization as the product of Lyndon words $w_1 w_2 \cdots w_n$, where $w_1 \geq w_2 \geq w_3 \cdots w_n$.

Fractional powers

We say a word w is a p/q power, for integers $p \geq q \geq 1$, if

$$w = x^{\lfloor p/q \rfloor} x'$$

for a prefix x' of x such that $|w|/|x| = p/q$.

Fractional powers

We say a word w is a p/q power, for integers $p \geq q \geq 1$, if

$$w = x^{\lfloor p/q \rfloor} x'$$

for a prefix x' of x such that $|w|/|x| = p/q$.

For example, the French word `entente` is a $7/3$-power, as it can be written as $(\texttt{ent})^2\texttt{e}$.

Fractional powers

We say a word w is a p/q power, for integers $p \geq q \geq 1$, if

$$w = x^{\lfloor p/q \rfloor} x'$$

for a prefix x' of x such that $|w|/|x| = p/q$.

For example, the French word `entente` is a $7/3$-power, as it can be written as $(\texttt{ent})^2\texttt{e}$.

The German word `schematische` is a $3/2$ power.

Fractional powers

We say a word w is a p/q power, for integers $p \geq q \geq 1$, if

$$w = x^{\lfloor p/q \rfloor} x'$$

for a prefix x' of x such that $|w|/|x| = p/q$.

For example, the French word `entente` is a $7/3$-power, as it can be written as $(\texttt{ent})^2\texttt{e}$.

The German word `schematische` is a $3/2$ power.

If $w = x^{\lfloor p/q \rfloor} x'$ is a p/q power, then we call x a *period* of w. Often the word *period* is used to refer to $|x|$.

Fractional powers

We say a word w is a p/q power, for integers $p \geq q \geq 1$, if

$$w = x^{\lfloor p/q \rfloor} x'$$

for a prefix x' of x such that $|w|/|x| = p/q$.

For example, the French word `entente` is a $7/3$-power, as it can be written as $(\texttt{ent})^2\texttt{e}$.

The German word `schematische` is a $3/2$ power.

If $w = x^{\lfloor p/q \rfloor} x'$ is a p/q power, then we call x a *period* of w. Often the word *period* is used to refer to $|x|$.

If a word w is a $p/q > 1$ power, then it begins and ends with some nonempty string. Such a string is also called *bordered*. Otherwise it is *unbordered*.

Unbordered words

The unbordered words play the same role for fractional powers as the primitive words do for ordinary powers.

Unbordered words

The unbordered words play the same role for fractional powers as the primitive words do for ordinary powers.

Enumeration of unbordered words is more challenging and there is no simple closed form.

Unbordered words

The unbordered words play the same role for fractional powers as the primitive words do for ordinary powers.

Enumeration of unbordered words is more challenging and there is no simple closed form.

However, there are asymptotically $c_k k^n$ such words, where c_k is a constant that tends to 1 as k tends to ∞.

Duval's conjecture

Let $\mu(w)$ be the length of the longest unbordered factor of w.

Duval's conjecture

Let $\mu(w)$ be the length of the longest unbordered factor of w.

Let $p(w)$ be the length of the longest period of w.

Duval's conjecture

Let $\mu(w)$ be the length of the longest unbordered factor of w.

Let $p(w)$ be the length of the longest period of w.

Duval conjectured that if $|w| \geq 3\mu(w)$, then $\mu(w) = p(w)$.

Duval's conjecture

Let $\mu(w)$ be the length of the longest unbordered factor of w.

Let $p(w)$ be the length of the longest period of w.

Duval conjectured that if $|w| \geq 3\mu(w)$, then $\mu(w) = p(w)$.

This was proved by Harju & Nowotka, and S. Holub. The result has been improved to $|w| \geq 3\mu(w) - 2 \implies \mu(w) = p(w)$.

The Fine-Wilf theorem

Theorem

Let w and x be nonempty words. Let $\mathbf{y} \in w\{w, x\}^\omega$ and $\mathbf{z} \in x\{w, x\}^\omega$. Then the following conditions are equivalent:

(a) $\mathbf{y}$ *and* $\mathbf{z}$ *agree on a prefix of length at least* $|w| + |x| - \gcd(|w|, |x|)$;

The Fine-Wilf theorem

Theorem

Let w and x be nonempty words. Let $\mathbf{y} \in w\{w, x\}^\omega$ and $\mathbf{z} \in x\{w, x\}^\omega$. Then the following conditions are equivalent:

(a) $\mathbf{y}$ *and* $\mathbf{z}$ *agree on a prefix of length at least* $|w| + |x| - \gcd(|w|, |x|)$;

(b) $wx = xw$;

The Fine-Wilf theorem

Theorem

Let w and x be nonempty words. Let $\mathbf{y} \in w\{w, x\}^\omega$ and $\mathbf{z} \in x\{w, x\}^\omega$. Then the following conditions are equivalent:

(a) $\mathbf{y}$ *and* $\mathbf{z}$ *agree on a prefix of length at least* $|w| + |x| - \gcd(|w|, |x|)$*;*

(b) $wx = xw$*;*

(c) $\mathbf{y} = \mathbf{z}$*.*

(c) $\implies$ (a): Trivial.

The Fine-Wilf theorem

Theorem

Let w and x be nonempty words. Let $\mathbf{y} \in w\{w, x\}^\omega$ and $\mathbf{z} \in x\{w, x\}^\omega$. Then the following conditions are equivalent:

(a) $\mathbf{y}$ *and* $\mathbf{z}$ *agree on a prefix of length at least* $|w| + |x| - \gcd(|w|, |x|)$;

(b) $wx = xw$;

(c) $\mathbf{y} = \mathbf{z}$.

(c) $\implies$ (a): Trivial.
We'll prove (a) $\implies$ (b) and (b) $\implies$ (c).

Fine-Wilf: The proof

Proof.
(a) **y** and **z** agree on a prefix of length at least $|w| + |x| - \gcd(|w|, |x|) \implies$ (b) $wx = xw$

Fine-Wilf: The proof

Proof.
(a) **y** and **z** agree on a prefix of length at least $|w| + |x| - \gcd(|w|, |x|) \Longrightarrow$ (b) $wx = xw$

We prove the contrapositive. Suppose $wx \neq xw$.

Fine-Wilf: The proof

Proof.
(a) **y** and **z** agree on a prefix of length at least $|w| + |x| - \gcd(|w|, |x|) \Longrightarrow$ (b) $wx = xw$

We prove the contrapositive. Suppose $wx \neq xw$.

Then we prove that **y** and **z** differ at a position $\leq |w| + |x| - \gcd(|w|, |x|)$.

Fine-Wilf: The proof

Proof.
(a) **y** and **z** agree on a prefix of length at least $|w| + |x| - \gcd(|w|, |x|) \Longrightarrow$ (b) $wx = xw$

We prove the contrapositive. Suppose $wx \neq xw$.

Then we prove that **y** and **z** differ at a position $\leq |w| + |x| - \gcd(|w|, |x|)$.

The proof is by induction on $|w| + |x|$.

Fine-Wilf: The proof

Proof.

(a) **y** and **z** agree on a prefix of length at least $|w| + |x| - \gcd(|w|, |x|) \Longrightarrow$ (b) $wx = xw$

We prove the contrapositive. Suppose $wx \neq xw$.

Then we prove that **y** and **z** differ at a position $\leq |w| + |x| - \gcd(|w|, |x|)$.

The proof is by induction on $|w| + |x|$.

The base case is $|w| + |x| = 2$. Then $|w| = |x| = 1$, and $|w| + |x| - \gcd(|w|, |x|) = 1$. Since $wx \neq xw$, we must have $w = a$, $x = b$ with $a \neq b$. Then **y** and **z** differ at the first position.

Now assume the result is true for $|w| + |x| < k$.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

If $|w| = |x|$ then **y** and **z** must disagree at the $|w|$'th position or earlier, for otherwise $w = x$ and $wx = xw$; since $|w| \leq |w| + |x| - \gcd(|w|, |x|) = |w|$, the result follows.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

If $|w| = |x|$ then **y** and **z** must disagree at the $|w|$'th position or earlier, for otherwise $w = x$ and $wx = xw$; since $|w| \leq |w| + |x| - \gcd(|w|, |x|) = |w|$, the result follows.

So, without loss of generality, assume $|w| < |x|$.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

If $|w| = |x|$ then **y** and **z** must disagree at the $|w|$'th position or earlier, for otherwise $w = x$ and $wx = xw$; since $|w| \leq |w| + |x| - \gcd(|w|, |x|) = |w|$, the result follows.

So, without loss of generality, assume $|w| < |x|$.

If w is not a prefix of x, then **y** and **z** disagree on the $|w|$'th position or earlier, and again $|w| \leq |w| + |x| - \gcd(|w|, |x|)$.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

If $|w| = |x|$ then **y** and **z** must disagree at the $|w|$'th position or earlier, for otherwise $w = x$ and $wx = xw$; since $|w| \leq |w| + |x| - \gcd(|w|, |x|) = |w|$, the result follows.

So, without loss of generality, assume $|w| < |x|$.

If w is not a prefix of x, then **y** and **z** disagree on the $|w|$'th position or earlier, and again $|w| \leq |w| + |x| - \gcd(|w|, |x|)$.

So w is a proper prefix of x.

Now assume the result is true for $|w| + |x| < k$.

We prove it for $|w| + |x| = k$.

If $|w| = |x|$ then **y** and **z** must disagree at the $|w|$'th position or earlier, for otherwise $w = x$ and $wx = xw$; since $|w| \leq |w| + |x| - \gcd(|w|, |x|) = |w|$, the result follows.

So, without loss of generality, assume $|w| < |x|$.

If w is not a prefix of x, then **y** and **z** disagree on the $|w|$'th position or earlier, and again $|w| \leq |w| + |x| - \gcd(|w|, |x|)$.

So w is a proper prefix of x.

Write $x = wt$ for some nonempty word t.

Now any common divisor of $|w|$ and $|x|$ must also divide $|x| - |w| = |t|$, and similarly any common divisor of both $|w|$ and $|t|$ must also divide $|w| + |t| = |x|$. So $\gcd(|w|, |x|) = \gcd(|w|, |t|)$.

Now $wt \neq tw$, for otherwise we have $wx = wwt = wtw = xw$, a contradiction.

Now $wt \neq tw$, for otherwise we have $wx = wwt = wtw = xw$, a contradiction.

Then $\mathbf{y} = ww\cdots$ and $\mathbf{z} = wt\cdots$. By induction (since $|w| + |t| < k$) $w^{-1}\mathbf{y}$ and $w^{-1}\mathbf{z}$ disagree at position $|w| + |t| - \gcd(|w|, |t|)$ or earlier.

Now $wt \neq tw$, for otherwise we have $wx = wwt = wtw = xw$, a contradiction.

Then $\mathbf{y} = ww \cdots$ and $\mathbf{z} = wt \cdots$. By induction (since $|w| + |t| < k$) $w^{-1}\mathbf{y}$ and $w^{-1}\mathbf{z}$ disagree at position $|w| + |t| - \gcd(|w|, |t|)$ or earlier.

Hence $\mathbf{y}$ and $\mathbf{z}$ disagree at position $2|w| + |t| - \gcd(|w|, |t|) = |w| + |x| - \gcd(|w|, |x|)$ or earlier.

(b) $\implies$ (c): If $wx = xw$, then by the theorem of Lyndon-Schützenberger, both w and x are in u^+ for some word u. Hence $\mathbf{y} = u^\omega = \mathbf{z}$. ■

Finite Sturmian words

The proof also implies a way to get words that optimally "almost commute", in the sense that xw and wx should agree on as long a segment as possible.

Finite Sturmian words

The proof also implies a way to get words that optimally "almost commute", in the sense that xw and wx should agree on as long a segment as possible.

Theorem

For each $m, n \geq 1$ there exist words x, w of length m, n, respectively, such that xw and wx agree on a prefix of length $m + n - \gcd(m, n) - 1$ but differ at position $m + n - \gcd(m, n)$.

Finite Sturmian words

The proof also implies a way to get words that optimally "almost commute", in the sense that xw and wx should agree on as long a segment as possible.

Theorem

For each $m, n \geq 1$ there exist words x, w of length m, n, respectively, such that xw and wx agree on a prefix of length $m + n - \gcd(m, n) - 1$ but differ at position $m + n - \gcd(m, n)$.

These are the finite Sturmian words.

Finite Sturmian words

The proof also implies a way to get words that optimally "almost commute", in the sense that xw and wx should agree on as long a segment as possible.

Theorem

For each $m, n \geq 1$ there exist words x, w of length m, n, respectively, such that xw and wx agree on a prefix of length $m + n - \gcd(m, n) - 1$ but differ at position $m + n - \gcd(m, n)$.

These are the finite Sturmian words.

Many authors have worked on generalizations to multiple periods: Castelli, Mignosi, & Restivo, Simpson & Tijdeman, Constantinescu & Ilie, ...

Patterns and pattern avoidance

The story begins with Axel Thue in 1906.

Patterns and pattern avoidance

The story begins with Axel Thue in 1906.

He noticed that over a 2-letter alphabet, every word of length ≥ 4 contains a square: either 0^2, 1^2, $(01)^2$ or $(10)^2$.

Patterns and pattern avoidance

The story begins with Axel Thue in 1906.

He noticed that over a 2-letter alphabet, every word of length ≥ 4 contains a square: either 0^2, 1^2, $(01)^2$ or $(10)^2$.

But over a 3-letter alphabet, it is possible to create arbitrarily long words (or — what is equivalent — an infinite word) with no square factors at all. Such a word is called *squarefree*.

Patterns and pattern avoidance

The story begins with Axel Thue in 1906.

He noticed that over a 2-letter alphabet, every word of length ≥ 4 contains a square: either 0^2, 1^2, $(01)^2$ or $(10)^2$.

But over a 3-letter alphabet, it is possible to create arbitrarily long words (or — what is equivalent — an infinite word) with no square factors at all. Such a word is called *squarefree*.

The easiest way to construct such a sequence was found by Thue in 1912 (and rediscovered many times).

Patterns and pattern avoidance

The story begins with Axel Thue in 1906.

He noticed that over a 2-letter alphabet, every word of length ≥ 4 contains a square: either 0^2, 1^2, $(01)^2$ or $(10)^2$.

But over a 3-letter alphabet, it is possible to create arbitrarily long words (or — what is equivalent — an infinite word) with no square factors at all. Such a word is called *squarefree*.

The easiest way to construct such a sequence was found by Thue in 1912 (and rediscovered many times).

It is based on the Thue-Morse sequence.

The Thue-Morse morphism

Morphism: a map h from Σ^* to Δ^* such that

$$h(xy) = h(x)h(y).$$

The Thue-Morse morphism

Morphism: a map h from Σ^* to Δ^* such that

$$h(xy) = h(x)h(y).$$

Thue-Morse morphism: $\mu(0) = 01$; $\mu(1) = 10$.

The Thue-Morse morphism

Morphism: a map h from Σ^* to Δ^* such that

$$h(xy) = h(x)h(y).$$

Thue-Morse morphism: $\mu(0) = 01$; $\mu(1) = 10$.

If $\Sigma = \Delta$ then we can iterate h.

The Thue-Morse morphism

Morphism: a map h from Σ^* to Δ^* such that

$$h(xy) = h(x)h(y).$$

Thue-Morse morphism: $\mu(0) = 01$; $\mu(1) = 10$.

If $\Sigma = \Delta$ then we can iterate h.

We write $h^i = \overbrace{h(h(h(\cdots)))}^{i}$.

Morphic words

If a nonerasing morphism has the property that $h(a) = ax$, then iterating h produces an infinite word

$$h^{\omega}(a) = a\,x\,h(x)\,h^{2}(x)\,h^{3}(x)\cdots .$$

Morphic words

If a nonerasing morphism has the property that $h(a) = ax$, then iterating h produces an infinite word

$$h^\omega(a) = a\,x\,h(x)\,h^2(x)\,h^3(x)\cdots.$$

If we do this with μ we get the *Thue-Morse* word:

Morphic words

If a nonerasing morphism has the property that $h(a) = ax$, then iterating h produces an infinite word

$$h^{\omega}(a) = a\,x\,h(x)\,h^2(x)\,h^3(x)\cdots .$$

If we do this with μ we get the *Thue-Morse* word:

$$\mathbf{t} = \mu^{\omega}(0) = 0110100110010110\cdots .$$

Morphic words

If a nonerasing morphism has the property that $h(a) = ax$, then iterating h produces an infinite word

$$h^\omega(a) = a\,x\,h(x)\,h^2(x)\,h^3(x)\cdots .$$

If we do this with μ we get the *Thue-Morse* word:

$$\mathbf{t} = \mu^\omega(0) = 0110100110010110\cdots .$$

Also rediscovered by Marston Morse, Max Euwe, Solomon Arshon, and the Danish composer Per Nørgård.

Properties of the Thue-Morse word

An *overlap* is a word of the form *axaxa*, where *a* is a single letter and *x* is a word.

Properties of the Thue-Morse word

An *overlap* is a word of the form $axaxa$, where a is a single letter and x is a word.

An example in English is `alfalfa` (take $x = \texttt{lf}$).

Properties of the Thue-Morse word

An *overlap* is a word of the form *axaxa*, where *a* is a single letter and *x* is a word.

An example in English is `alfalfa` (take $x = \texttt{lf}$).

Thus an overlap is just slightly more than a square.

Properties of the Thue-Morse word

An *overlap* is a word of the form $axaxa$, where a is a single letter and x is a word.

An example in English is `alfalfa` (take $x = \texttt{lf}$).

Thus an overlap is just slightly more than a square.

It is also called a 2^+ power.

Properties of the Thue-Morse word

An *overlap* is a word of the form *axaxa*, where *a* is a single letter and *x* is a word.

An example in English is `alfalfa` (take $x = \texttt{lf}$).

Thus an overlap is just slightly more than a square.

It is also called a 2^+ power.

Theorem

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

From overlap-free to squarefree

We can construct a squarefree word from **t**, as follows:

From overlap-free to squarefree

We can construct a squarefree word from **t**, as follows:

Count the number of 1's in **t** between consecutive 0's:

From overlap-free to squarefree

We can construct a squarefree word from **t**, as follows:

Count the number of 1's in **t** between consecutive 0's:

We get:

From overlap-free to squarefree

We can construct a squarefree word from **t**, as follows:

Count the number of 1's in **t** between consecutive 0's:

We get:

$$0\overbrace{11}^{2}0\overbrace{1}^{1}0\overbrace{}^{0}0\overbrace{11}^{2}0\overbrace{}^{0}0\overbrace{1}^{1}0\overbrace{11}^{2}0\overbrace{1}^{1}0\overbrace{}^{0}0\overbrace{1}^{1}0\overbrace{11}^{2}\cdots$$

From overlap-free to squarefree

We can construct a squarefree word from $\mathbf{t}$, as follows:

Count the number of 1's in $\mathbf{t}$ between consecutive 0's:

We get:

$$0\overbrace{11}^{2}0\overbrace{1}^{1}0\overbrace{}^{0}0\overbrace{11}^{2}0\overbrace{}^{0}0\overbrace{1}^{1}0\overbrace{11}^{2}0\overbrace{1}^{1}0\overbrace{}^{0}0\overbrace{1}^{1}0\overbrace{11}^{2}\cdots$$

This is squarefree, as a square in this word implies and overlap in the Thue-Morse word.

Enumeration of power-free words

How many squarefree words are there?

Enumeration of power-free words

How many squarefree words are there?

Infinite - countable or uncountable

Enumeration of power-free words

How many squarefree words are there?

Infinite - countable or uncountable

Finite - polynomially-many or exponentially-many of length n?

Enumeration of power-free words

How many squarefree words are there?

Infinite - countable or uncountable

Finite - polynomially-many or exponentially-many of length n?

Same question can be asked for the overlap-free words.

Enumeration of power-free words

How many squarefree words are there?

Infinite - countable or uncountable

Finite - polynomially-many or exponentially-many of length n?

Same question can be asked for the overlap-free words.

For overlap-free words over $\{0, 1\}$ there is a factorization theorem of Restivo and Salemi that implies only polynomially-many of length n.

Enumeration of power-free words

How many squarefree words are there?

Infinite - countable or uncountable

Finite - polynomially-many or exponentially-many of length n?

Same question can be asked for the overlap-free words.

For overlap-free words over $\{0, 1\}$ there is a factorization theorem of Restivo and Salemi that implies only polynomially-many of length n.

For squarefree words over $\{0, 1, 2\}$ there are exponentially many.

Dejean's Conjecture

Given an alphabet Σ of cardinality k, we can try to find the optimal (fractional) exponent α_k avoidable by infinite words over Σ.

Dejean's Conjecture

Given an alphabet Σ of cardinality k, we can try to find the optimal (fractional) exponent α_k avoidable by infinite words over Σ.

For $k = 2$ we have already seen that overlaps are avoidable and squares are not. So $\alpha_2 = 2$.

Dejean's Conjecture

Given an alphabet Σ of cardinality k, we can try to find the optimal (fractional) exponent α_k avoidable by infinite words over Σ.

For $k = 2$ we have already seen that overlaps are avoidable and squares are not. So $\alpha_2 = 2$.

Dejean (1972) showed that $\alpha_3 = 7/4$ and conjectured that $\alpha_4 = 7/5$ and $\alpha_k = k/(k-1)$ for $k \geq 5$.

Dejean's Conjecture

Given an alphabet Σ of cardinality k, we can try to find the optimal (fractional) exponent α_k avoidable by infinite words over Σ.

For $k = 2$ we have already seen that overlaps are avoidable and squares are not. So $\alpha_2 = 2$.

Dejean (1972) showed that $\alpha_3 = 7/4$ and conjectured that $\alpha_4 = 7/5$ and $\alpha_k = k/(k-1)$ for $k \geq 5$.

This conjecture has been proven by the combined efforts of Pansiot, Moulin-Ollagnier, Currie & Mohammad-Noori, Carpi, Currie & Rampersad, and Rao.

Dejean's Conjecture

Given an alphabet Σ of cardinality k, we can try to find the optimal (fractional) exponent α_k avoidable by infinite words over Σ.

For $k = 2$ we have already seen that overlaps are avoidable and squares are not. So $\alpha_2 = 2$.

Dejean (1972) showed that $\alpha_3 = 7/4$ and conjectured that $\alpha_4 = 7/5$ and $\alpha_k = k/(k-1)$ for $k \geq 5$.

This conjecture has been proven by the combined efforts of Pansiot, Moulin-Ollagnier, Currie & Mohammad-Noori, Carpi, Currie & Rampersad, and Rao.

Still open: many other variants of Dejean where the length of the period is also taken into account.

More general patterns

Instead of avoiding xx or $axaxa$, we can try to avoid more general patterns.

More general patterns

Instead of avoiding *xx* or *axaxa*, we can try to avoid more general patterns.

"Avoiding the pattern α" means constructing an infinite word $\mathbf{x}$ such that, for all non-erasing morphisms h, the word $h(\alpha)$ is not a factor of $\mathbf{x}$.

More general patterns

Instead of avoiding *xx* or *axaxa*, we can try to avoid more general patterns.

"Avoiding the pattern α" means constructing an infinite word **x** such that, for all non-erasing morphisms *h*, the word $h(\alpha)$ is not a factor of **x**.

Not all patterns are avoidable — even if the alphabet is arbitrarily large.

More general patterns

Instead of avoiding xx or $axaxa$, we can try to avoid more general patterns.

"Avoiding the pattern α" means constructing an infinite word $\mathbf{x}$ such that, for all non-erasing morphisms h, the word $h(\alpha)$ is not a factor of $\mathbf{x}$.

Not all patterns are avoidable — even if the alphabet is arbitrarily large.

For example - it is impossible to avoid xyx, since every sufficiently long string z will contain three occurrences of some letter a, say $z = rasatau$, and then we can let $x = a$, $y = sat$, both x and y are nonempty.

Avoiding general patterns

Given a pattern, it is decidable (via *Zimin's algorithm*) if it is avoidable over *some* alphabet.

Avoiding general patterns

Given a pattern, it is decidable (via *Zimin's algorithm*) if it is avoidable over *some* alphabet.

However, we do not have a general procedure to decide if a given pattern is avoidable over a fixed alphabet.

Abelian powers

Instead of avoiding xx, we can consider trying to avoid other kinds of patterns: the so-called *abelian* powers.

Abelian powers

Instead of avoiding xx, we can consider trying to avoid other kinds of patterns: the so-called *abelian* powers.

An abelian square is a nonempty word of the form xx', where x' is a permutation of x.

Abelian powers

Instead of avoiding xx, we can consider trying to avoid other kinds of patterns: the so-called *abelian* powers.

An abelian square is a nonempty word of the form xx', where x' is a permutation of x.

For example, `interessierten` is an abelian square in German, as `sierten` is a permutation of `interes`.

Abelian powers

Instead of avoiding xx, we can consider trying to avoid other kinds of patterns: the so-called *abelian* powers.

An abelian square is a nonempty word of the form xx', where x' is a permutation of x.

For example, `interessierten` is an abelian square in German, as `sierten` is a permutation of `interes`.

In a similar way, we can define abelian cubes as words of the form $xx'x''$ where both x' and x'' are permutations of x.

Abelian powers: summary of results

it is possible to avoid abelian squares over a 4-letter alphabet, and this is optimal;

Abelian powers: summary of results

it is possible to avoid abelian squares over a 4-letter alphabet, and this is optimal;

it is possible to avoid abelian cubes over a 3-letter alphabet, and this is optimal;

Abelian powers: summary of results

it is possible to avoid abelian squares over a 4-letter alphabet, and this is optimal;

it is possible to avoid abelian cubes over a 3-letter alphabet, and this is optimal;

it is possible to avoid abelian fourth powers over a 2-letter alphabet, and this is optimal.

Abelian powers: summary of results

it is possible to avoid abelian squares over a 4-letter alphabet, and this is optimal;

it is possible to avoid abelian cubes over a 3-letter alphabet, and this is optimal;

it is possible to avoid abelian fourth powers over a 2-letter alphabet, and this is optimal.

An open problem: is it possible to avoid, over a finite subset of $\mathbb{N}$, patterns of the form xx' where $|x| = |x'|$ and $\sum x = \sum x'$?

Abelian powers: summary of results

it is possible to avoid abelian squares over a 4-letter alphabet, and this is optimal;

it is possible to avoid abelian cubes over a 3-letter alphabet, and this is optimal;

it is possible to avoid abelian fourth powers over a 2-letter alphabet, and this is optimal.

An open problem: is it possible to avoid, over a finite subset of $\mathbb{N}$, patterns of the form xx' where $|x| = |x'|$ and $\sum x = \sum x'$?

Also still open: fractional version of abelian powers

A generalization of abelian powers

involution: $h^2(x) = x$ for all words x

A generalization of abelian powers

involution: $h^2(x) = x$ for all words x

involutions can be morphic ($h(xy) = h(x)h(y)$) or antimorphic ($h(xy) = h(y)h(x)$)

A generalization of abelian powers

involution: $h^2(x) = x$ for all words x

involutions can be morphic ($h(xy) = h(x)h(y)$) or antimorphic ($h(xy) = h(y)h(x)$)

One antimorphism of biological interest: reverse string and apply map $A \leftrightarrow T$, $G \leftrightarrow C$.

A generalization of abelian powers

involution: $h^2(x) = x$ for all words x

involutions can be morphic ($h(xy) = h(x)h(y)$) or antimorphic ($h(xy) = h(y)h(x)$)

One antimorphism of biological interest: reverse string and apply map $A \leftrightarrow T$, $G \leftrightarrow C$.

Can avoid some patterns involving involution, but not others

Equations in words

Example 1:

$$abX = Xba.$$

Equations in words

Example 1:

$$abX = Xba.$$

The only solutions are $X \in (ab)^*a$.

Equations in words

Example 1:

$$abX = Xba.$$

The only solutions are $X \in (ab)^* a$.

Example 2:

$$XaXbY = aXYbX$$

Equations in words

Example 1:

$$abX = Xba.$$

The only solutions are $X \in (ab)^*a$.

Example 2:

$$XaXbY = aXYbX$$

The solutions are $X = a^i$, $Y = (a^i b)^j a^i$ for $i \geq 0, j \geq 0$.

Equations in words

Example 3: Fermat's equation for words: $x^i y^j = z^k$

Equations in words

Example 3: Fermat's equation for words: $x^i y^j = z^k$

The only solutions for $i, j, k \geq 2$ are when x, y, z are all powers of a third word.

Equations in words

Example 3: Fermat's equation for words: $x^i y^j = z^k$

The only solutions for $i, j, k \geq 2$ are when x, y, z are all powers of a third word.

Example 4:

$XYZ = ZVX$: many solutions, but not expressible by formula with integer parameters.

Equations in words

More generally, given an equation in words and constants, there is an algorithm (Makanin's algorithm) that is guaranteed to find a solution if one exists.

Equations in words

More generally, given an equation in words and constants, there is an algorithm (Makanin's algorithm) that is guaranteed to find a solution if one exists.

Satisfiability in PSPACE: Plandowski (2004)

Equations in words

More generally, given an equation in words and constants, there is an algorithm (Makanin's algorithm) that is guaranteed to find a solution if one exists.

Satisfiability in PSPACE: Plandowski (2004)

Finiteness of solutions: Plandowski (2006) but complete proofs not yet published.

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

- the Thue-Morse word, obtained by iterating $0 \to 01$, $1 \to 10$

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

- the Thue-Morse word, obtained by iterating $0 \to 01$, $1 \to 10$
- the Fibonacci word, obtained by iterating $0 \to 01$, $1 \to 0$

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

- the Thue-Morse word, obtained by iterating $0 \to 01$, $1 \to 10$
- the Fibonacci word, obtained by iterating $0 \to 01$, $1 \to 0$

morphic: obtained by iterating a morphism, then applying a coding (letter-to-letter morphism)

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

- the Thue-Morse word, obtained by iterating $0 \to 01$, $1 \to 10$
- the Fibonacci word, obtained by iterating $0 \to 01$, $1 \to 0$

morphic: obtained by iterating a morphism, then applying a coding (letter-to-letter morphism)

fixed point of uniform morphism: like the Thue-Morse word

Kinds of infinite words

pure morphic: obtained by iterating a morphism

Examples:

- the Thue-Morse word, obtained by iterating $0 \to 01$, $1 \to 10$
- the Fibonacci word, obtained by iterating $0 \to 01$, $1 \to 0$

morphic: obtained by iterating a morphism, then applying a coding (letter-to-letter morphism)

fixed point of uniform morphism: like the Thue-Morse word

automatic: image, under a coding, of a fixed point of a uniform morphism

Other important infinite words

Sturmian words: exactly $n+1$ factors of length n

Other important infinite words

Sturmian words: exactly $n + 1$ factors of length n
Many other characterizations:

Other important infinite words

Sturmian words: exactly $n + 1$ factors of length n
Many other characterizations:

- balanced: any two words w, x of length n have $\delta(x, y) = 1$, where $\delta(x, y) = ||x|_1 - |y|_1|$

Other important infinite words

Sturmian words: exactly $n+1$ factors of length n
Many other characterizations:

- balanced: any two words w, x of length n have $\delta(x,y) = 1$, where $\delta(x,y) = ||x|_1 - |y|_1|$
- of the form $(\lfloor \alpha(n+1) + \gamma \rfloor - \lfloor \alpha n + \gamma \rfloor)_{n \geq 1}$ for real α, γ

Other important infinite words

Sturmian words: exactly $n+1$ factors of length n
Many other characterizations:

- balanced: any two words w, x of length n have $\delta(x, y) = 1$, where $\delta(x, y) = ||x|_1 - |y|_1|$
- of the form $(\lfloor \alpha(n+1) + \gamma \rfloor - \lfloor \alpha n + \gamma \rfloor)_{n \geq 1}$ for real α, γ

Episturmian words: natural generalization of Sturmian words to larger alphabets

More infinite words

Toeplitz words: generated by starting with a periodic word with "holes"; then inserting another periodic word with holes into that, etc.

More infinite words

Toeplitz words: generated by starting with a periodic word with "holes"; then inserting another periodic word with holes into that, etc.

Paperfolding words: generated by iterated folding of a piece of paper

More infinite words

Toeplitz words: generated by starting with a periodic word with "holes"; then inserting another periodic word with holes into that, etc.

Paperfolding words: generated by iterated folding of a piece of paper

- $X_{n+1} = X_n a \overline{X_n}^R$

More infinite words

Toeplitz words: generated by starting with a periodic word with "holes"; then inserting another periodic word with holes into that, etc.

Paperfolding words: generated by iterated folding of a piece of paper

- $X_{n+1} = X_n a \overline{X_n}^R$

Kolakoski's word: generated by applying a transducer iteratively

More infinite words

Toeplitz words: generated by starting with a periodic word with "holes"; then inserting another periodic word with holes into that, etc.

Paperfolding words: generated by iterated folding of a piece of paper

- $X_{n+1} = X_n a \overline{X_n}^R$

Kolakoski's word: generated by applying a transducer iteratively

- The sequence $1221121221\cdots$ that encodes its own sequence of run lengths

Properties of infinite words

recurrence - every factor that occurs, occurs infinitely often

Properties of infinite words

recurrence - every factor that occurs, occurs infinitely often

uniform recurrence - recurrent, plus distance between two consecutive occurrences of the same factor of length n is bounded, for all n

Subword complexity

"subword" complexity - given an infinite word **w**, count the number of distinct factors of length n in **w**

Subword complexity

“subword” complexity - given an infinite word **w**, count the number of distinct factors of length n in **w**

- $O(n)$ for automatic sequences

Subword complexity

"subword" complexity - given an infinite word $\mathbf{w}$, count the number of distinct factors of length n in $\mathbf{w}$

- $O(n)$ for automatic sequences
- $n + 1$ for Sturmian words

Subword complexity

"subword" complexity - given an infinite word **w**, count the number of distinct factors of length n in **w**

- $O(n)$ for automatic sequences
- $n+1$ for Sturmian words
- $O(n^2)$ for morphic words

Subword complexity

"subword" complexity - given an infinite word **w**, count the number of distinct factors of length n in **w**

- $O(n)$ for automatic sequences
- $n+1$ for Sturmian words
- $O(n^2)$ for morphic words
- A classification of possible growth rates exists

Automatic sequences

- A *deterministic finite automaton with output* (DFAO) is a 6-tuple: $(Q, \Sigma, \delta, q_0, \Delta, \tau)$, where Δ is the finite *output alphabet* and $\tau : Q \to \Delta$ is the *output mapping*.

Automatic sequences

- A *deterministic finite automaton with output* (DFAO) is a 6-tuple: $(Q, \Sigma, \delta, q_0, \Delta, \tau)$, where Δ is the finite *output alphabet* and $\tau : Q \to \Delta$ is the *output mapping*.
- Next, we decide on a integer base $k \geq 2$ and represent n as a string of symbols over the alphabet $\Sigma = \{0, 1, 2, \ldots, k-1\}$.

Automatic sequences

- A *deterministic finite automaton with output* (DFAO) is a 6-tuple: $(Q, \Sigma, \delta, q_0, \Delta, \tau)$, where Δ is the finite *output alphabet* and $\tau : Q \to \Delta$ is the *output mapping*.
- Next, we decide on a integer base $k \geq 2$ and represent n as a string of symbols over the alphabet $\Sigma = \{0, 1, 2, \ldots, k-1\}$.
- To compute f_n, given an automaton M, express n in base-k, say,

$$a_r a_{r-1} \cdots a_1 a_0,$$

and compute

$$f_n = \tau(\delta(q_0, a_r a_{r-1} \cdots a_1 a_0)).$$

Automatic sequences

- A *deterministic finite automaton with output* (DFAO) is a 6-tuple: $(Q, \Sigma, \delta, q_0, \Delta, \tau)$, where Δ is the finite *output alphabet* and $\tau : Q \to \Delta$ is the *output mapping*.
- Next, we decide on a integer base $k \geq 2$ and represent n as a string of symbols over the alphabet $\Sigma = \{0, 1, 2, \ldots, k-1\}$.
- To compute f_n, given an automaton M, express n in base-k, say,

$$a_r a_{r-1} \cdots a_1 a_0,$$

  and compute

$$f_n = \tau(\delta(q_0, a_r a_{r-1} \cdots a_1 a_0)).$$

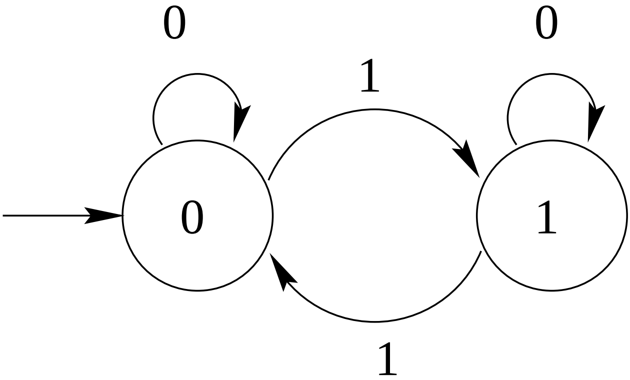
- Any sequence that can be computed in this way is said to be <u>k-automatic</u>.

The Thue-Morse automaton

- The word $\mathbf{t}$ is computed by the following DFAO:

The Thue-Morse automaton

- The word **t** is computed by the following DFAO:



Robustness

- the order in which the base-k digits are fed into the automaton in does not matter (provided it is fixed for all n);

Robustness

- the order in which the base-k digits are fed into the automaton in does not matter (provided it is fixed for all n);
- other representations also work (such as expansion in base-$(-k)$);

Robustness

- the order in which the base-k digits are fed into the automaton in does not matter (provided it is fixed for all n);
- other representations also work (such as expansion in base-$(-k)$);
- automatic sequences are closed under many operations, such as shift, periodic deletion, q-block compression, and q-block substitution.

Robustness

- the order in which the base-k digits are fed into the automaton in does not matter (provided it is fixed for all n);
- other representations also work (such as expansion in base-$(-k)$);
- automatic sequences are closed under many operations, such as shift, periodic deletion, q-block compression, and q-block substitution.
- if a symbol in an automatic sequence occurs with well-defined frequency r, then r is rational.

Christol's theorem

Theorem

(Christol [1980]). *Let* $(u_n)_{n\geq 0}$ *be a sequence over*

$$\Sigma = \{0, 1, \ldots, p-1\},$$

where p *is a prime. Then the formal power series* $U(X) = \sum_{n\geq 0} u_n X^n$ *is algebraic over* $GF(p)[X]$ *if and only if* $(u_n)_{n\geq 0}$ *is* p*-automatic.*

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.

Christol's theorem: example

Let $(t_n)_{n \geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

$$A(X) = \sum_{n\geq 0} t_{2n} X^{2n} + \sum_{n\geq 0} t_{2n+1} X^{2n+1}$$

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

$$\begin{aligned} A(X) &= \sum_{n\geq 0} t_{2n} X^{2n} + \sum_{n\geq 0} t_{2n+1} X^{2n+1} \\ &= \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} X^{2n} \end{aligned}$$

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

$$\begin{aligned} A(X) &= \sum_{n\geq 0} t_{2n} X^{2n} + \sum_{n\geq 0} t_{2n+1} X^{2n+1} \\ &= \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} X^{2n} \\ &= A(X^2) + XA(X^2) + X/(1 - X^2) \end{aligned}$$

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the THUE-MORSE sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

$$\begin{aligned}
A(X) &= \sum_{n\geq 0} t_{2n} X^{2n} + \sum_{n\geq 0} t_{2n+1} X^{2n+1} \\
&= \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} X^{2n} \\
&= A(X^2) + XA(X^2) + X/(1 - X^2) \\
&= A(X)^2(1 + X) + X/(1 + X)^2.
\end{aligned}$$

Christol's theorem: example

Let $(t_n)_{n\geq 0}$ denote the Thue-Morse sequence.
Then $t_n =$ sum of the bits in the binary expansion of n, mod 2.
Also $t_{2n} \equiv t_n$ and $t_{2n+1} \equiv t_n + 1$. If we set $A(X) = \sum_{n\geq 0} t_n X^n$, then

$$\begin{aligned} A(X) &= \sum_{n\geq 0} t_{2n} X^{2n} + \sum_{n\geq 0} t_{2n+1} X^{2n+1} \\ &= \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} t_n X^{2n} + X \sum_{n\geq 0} X^{2n} \\ &= A(X^2) + XA(X^2) + X/(1 - X^2) \\ &= A(X)^2(1+X) + X/(1+X)^2. \end{aligned}$$

Hence $(1+X)^3 A^2 + (1+X)^2 A + X = 0.$

Open Problems

- Is the set of primitive words over $\{0,1\}$ context-free? (Almost certainly not.)

Open Problems

- Is the set of primitive words over $\{0, 1\}$ context-free? (Almost certainly not.)
- What are the frequencies of letters in Kolakoski's word? Do they exist? Are they equal to $1/2$?

Open Problems

- Is the set of primitive words over $\{0, 1\}$ context-free? (Almost certainly not.)
- What are the frequencies of letters in Kolakoski's word? Do they exist? Are they equal to $1/2$?
- Nivat's conjecture: extension of periodicity to 2-dimensional arrays

Open Problems

- Is the set of primitive words over $\{0, 1\}$ context-free? (Almost certainly not.)
- What are the frequencies of letters in Kolakoski's word? Do they exist? Are they equal to $1/2$?
- Nivat's conjecture: extension of periodicity to 2-dimensional arrays
- Is there a word over a finite subset of $\mathbb{N}$ that avoids xx' with $|x| = |x'|$ and $\sum x = \sum x'$?

For Further Reading

- M. Lothaire, *Combinatorics on Words*, Cambridge, 1997 (reprint)
- M. Lothaire, *Algebraic Combinatorics on Words*, Cambridge, 2002
- M. Lothaire, *Applied Combinatorics on Words*, Cambridge, 2005
- V. Berthé, M. Rigo, *Combinatorics, Automata, and Number Theory*, Cambridge, 2010
- J.-P. Allouche and J. Shallit, *Automatic Sequences: Theory, Applications, Generalizations*, Cambridge, 2003.