Combinatorics on words: Factor complexity

Anna FRID

Aix-Marseille Université, September 2020

Definition

The *(factor) complexity* $p_{\mathbf{u}}(n)$ of an infinite word $\mathbf{u}$ is the number of its distinct factors of length n.

Definition

The *(factor) complexity* $p_{\mathbf{u}}(n)$ of an infinite word $\mathbf{u}$ is the number of its distinct factors of length n.

$$0110100110010110 0110\cdots$$

Definition

The *(factor) complexity* $p_{\mathbf{u}}(n)$ of an infinite word $\mathbf{u}$ is the number of its distinct factors of length n.

$$0110100110010110 0110\cdots$$

No factors 000, 111 $\Longrightarrow p_{\mathbf{u}}(3) = 6$.

Definition

The *(factor) complexity* $p_{\mathbf{u}}(n)$ of an infinite word $\mathbf{u}$ is the number of its distinct factors of length n.

$$01101001100101100110\cdots$$

No factors 000, 111 $\Longrightarrow p_{\mathbf{u}}(3) = 6$.

The factor complexity has almost nothing to do with the Kolmogorov complexity which is the "shortest possible description of the string".

Properties of factor complexity

Let $\mathbf{u}$ be an infinite word over k letters.

Properties of factor complexity

Let $\mathbf{u}$ be an infinite word over k letters.

- $1 \leq p_{\mathbf{u}}(n) \leq k^n$;

Properties of factor complexity

Let $\mathbf{u}$ be an infinite word over k letters.

- $1 \leq p_{\mathbf{u}}(n) \leq k^n$;
- $p_{\mathbf{u}}(n+1) \geq p_{\mathbf{u}}(n)$;

Properties of factor complexity

Let $\mathbf{u}$ be an infinite word over k letters.

- $1 \leq p_{\mathbf{u}}(n) \leq k^n$;
- $p_{\mathbf{u}}(n+1) \geq p_{\mathbf{u}}(n)$;
- If $p_{\mathbf{u}}(n+1) = p_{\mathbf{u}}(n)$, then $\mathbf{u}$ is ult. periodic.

Morse-Hedlund theorem

Theorem (Morse and Hedlund,1938)

An infinite word $\mathbf{u}$ *either is ultimately periodic, and then its complexity is ultimately constant, or satisfies* $p_{\mathbf{u}}(n) \geq n+1$.

Morse-Hedlund theorem

Theorem (Morse and Hedlund,1938)

An infinite word $\mathbf{u}$ *either is ultimately periodic, and then its complexity is ultimately constant, or satisfies* $p_{\mathbf{u}}(n) \geq n+1$.

A word $\mathbf{u}$ of complexity $p_{\mathbf{u}}(n) \geq n+1$ is called *Sturmian*.

Fibonacci word

Example (Fibonacci morphism)

$$\varphi(0) = 01, \varphi(1) = 0$$

$$0 \to 01 \to 01\,0 \to 010\,01 \to 01001\,010 \to 01001010\,01001 \to \cdots$$

Its fixed point is the *Fibonacci word*

$$\varphi^{\omega}(0) = 0100101001001010010100100101001001 \cdots$$

Fibonacci word

Example (Fibonacci morphism)

$$\varphi(0) = 01, \varphi(1) = 0$$

$$0 \to 01 \to 01\ 0 \to 010\ 01 \to 01001\ 010 \to 01001010\ 01001 \to \cdots$$

Its fixed point is the *Fibonacci word*

$$\varphi^\omega(0) = 0100101001001010010100100101001001 \cdots$$

Lemma

The Fibonacci word is Sturmian.

(The proof will follow.)

Special factors

Consider the set of factors $\mathrm{Fac}_{\mathbf{u}}(n)$ of an infinite word $\mathbf{u}$.

For a factor w of $\mathbf{u}$, denote by $L(w)$ ($R(w)$) the set of symbols a such that aw (wa) is also a factor of $\mathbf{u}$.
$\#L(w) = l(w)$,
$\#R(w) = r(w)$.

We say that w is a *left* (*right*) *special* factor of $\mathbf{u}$ if $l(w) \neq 1$ ($r(w) \neq 1$).

Special words

Denote by $RS_{\mathbf{u}}(n)$ the set of all right special factors of $\mathbf{u}$ of length n.

Special words

Denote by $RS_{\mathbf{u}}(n)$ the set of all right special factors of $\mathbf{u}$ of length n.

Then the first differences

$$d_{\mathbf{u}}(n) = p_{\mathbf{u}}(n+1) - p_{\mathbf{u}}(n) = \sum_{w \in \mathrm{Fac}_{\mathbf{u}}(n)} (r(w) - 1) = \sum_{w \in RS_{\mathbf{u}}(n)} (r(w) - 1).$$

Bispecial words

A word is *bispecial* if it is left and right special. The set of bispecial words $B_{\mathbf{u}}(n)$.

Bispecial words

A word is *bispecial* if it is left and right special. The set of bispecial words $B_{\mathbf{u}}(n)$.

Bispeciality degree:

$$b(v) = \#\{(a,b) | a,b \in \Sigma, avb \in F\} - l(v) - r(v) + 1$$

Bispecial words

A word is *bispecial* if it is left and right special. The set of bispecial words $B_{\mathbf{u}}(n)$.

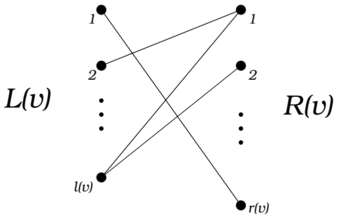
Bispeciality degree:

$$b(v) = \#\{(a,b) | a, b \in \Sigma, avb \in F\} - l(v) - r(v) + 1$$

Second differences

$$s_{\mathbf{u}}(n) = p_{\mathbf{u}}(n+2) - 2p_{\mathbf{u}}(n+1) + p_{\mathbf{u}}(n) = \sum_{v \in \mathrm{Fac}_{\mathbf{u}}(n)} b(v) = \sum_{v \in B_{\mathbf{u}}(n)} b(v).$$

Bispeciality graph



$$b(v) = \#\text{edges} - l(v) - r(v) + 1$$

Cassaigne, 1994

Fibonacci word is Sturmian

$$\varphi(0) = 01, \varphi(1) = 0$$

$$\mathbf{f} = \varphi^{\omega}(0) = 0\ 1\ 0\ 01\ 010\ 01001\ 01001010\ 0100101001001 \cdots$$

- It is not periodic since

$$\frac{|\varphi^n(0)|_0}{|\varphi^n(0)|} = \frac{F_{n+1}}{F_{n+2}} \to \frac{1}{\theta},$$

where $\theta = \frac{1+\sqrt{5}}{2}$ is the golden mean.

Fibonacci word is Sturmian

- $p_{\mathbf{f}}(1) = 2 \qquad (0, 1)$

  $p_{\mathbf{f}}(2) = 3 \qquad (00, 01, 10)$

  Suppose n is the shortest s.t. $d_{\mathbf{f}}(n+1) > 1$.

Fibonacci word is Sturmian

- $p_{\mathbf{f}}(1) = 2 \qquad (0, 1)$

  $p_{\mathbf{f}}(2) = 3 \qquad (00, 01, 10)$

  Suppose n is the shortest s.t. $d_{\mathbf{f}}(n+1) > 1$.

So, Sturmian words exist and the Fibonacci word is one of them.

Complexity of automatic words

Lemma

Let $\mathbf{u}$ *be a* k*-automatic word. Then for every* n *we have*

$$p_{\mathbf{u}}(kn+1) \leq kp_{\mathbf{u}}(n+1).$$

Complexity of automatic words

Lemma

Let $\mathbf{u}$ be a k-automatic word. Then for every n we have

$$p_{\mathbf{u}}(kn+1) \leq kp_{\mathbf{u}}(n+1).$$

Corollary

The complexity of a k-automatic word grows at most linearly.

Complexity of morphic words

- The complexity of a fixed point of a morphism can grow as $O(n^2)$, $O(n \log n)$, $O(n \log \log n)$, $O(n)$ or $O(1)$ [Pansiot 1984].

Complexity of morphic words

- The complexity of a fixed point of a morphism can grow as $O(n^2)$, $O(n \log n)$, $O(n \log \log n)$, $O(n)$ or $O(1)$ [Pansiot 1984].
- The complexity of a morphic word $\psi(\varphi^\omega(a))$ grows as $O(n^{1+1/k})$ for some k or at most as $O(n \log n)$ [Devyatov, 2008, preprint].

Further directions

- Characterizations of words of low complexity;

Further directions

- Characterizations of words of low complexity;
- Constructing word with given complexity growth;

Further directions

- Characterizations of words of low complexity;
- Constructing word with given complexity growth;
- Complexity of given words;

Further directions

- Characterizations of words of low complexity;
- Constructing word with given complexity growth;
- Complexity of given words;
- Complexity of languages (e. g. square-free words);

Further directions

- Characterizations of words of low complexity;
- Constructing word with given complexity growth;
- Complexity of given words;
- Complexity of languages (e. g. square-free words);
- General properties of the complexity function;

Further directions

- Characterizations of words of low complexity;
- Constructing word with given complexity growth;
- Complexity of given words;
- Complexity of languages (e. g. square-free words);
- General properties of the complexity function;
- Modifications of the definition.