Combinatorics of Finite Words and Suffix Automata

Gabriele Fici

Dipartimento di Informatica e Applicazioni
Università di Salerno (Italy)

CAI 2009 - Thessaloniki

20 May 2009

Combinatorics of Finite Words

A is a finite set of letters (the alphabet).

A finite word w is an element of A^*.

Its length $|w|$ is the number of its letters.

The empty word ε has length 0.

Let $w = a_1 a_2 \dots a_n$ be a word.

- $a_1 \dots a_i$, with $1 \leq i \leq n$, and ε are the prefixes of w.
- $a_j \dots a_n$, with $1 \leq j \leq n$, and ε are the suffixes of w.
- $a_j \dots a_i$, with $1 \leq i, j \leq n$, and ε are the factors of w.

Example

$A = \{a, n, b, c\}, \qquad w = banana$

$|banana| = 6$

ba is a prefix of *banana*

nana is a suffix of *banana*

a, *ba*, ε, *banana* are factors of *banana*

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.
- balanced words over two letters (say a and b): all the factors of the same length have the same number of a's (and of b's) up to 1. Ex. *abaababaabaab*.

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.
- balanced words over two letters (say *a* and *b*): all the factors of the same length have the same number of *a*'s (and of *b*'s) up to 1. Ex. *abaababaabaab*.
- differentiable words: words over $\{1, 2\}$ such that their Run Length Encoding is still a word over $\{1, 2\}$.
  Ex. 2211212212211

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.
- balanced words over two letters (say *a* and *b*): all the factors of the same length have the same number of *a*'s (and of *b*'s) up to 1. Ex. *abaababaabaab*.
- differentiable words: words over $\{1, 2\}$ such that their Run Length Encoding is still a word over $\{1, 2\}$.
  Ex. 2211212212211
- finite prefixes of (right) infinite words: Thue-Morse, Fibonacci, Kolakoski,...

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.
- balanced words over two letters (say *a* and *b*): all the factors of the same length have the same number of *a*'s (and of *b*'s) up to 1. Ex. *abaababaabaab*.
- differentiable words: words over $\{1, 2\}$ such that their Run Length Encoding is still a word over $\{1, 2\}$.
  Ex. 2211212212211
- finite prefixes of (right) infinite words: Thue-Morse, Fibonacci, Kolakoski,...
- many many others.

Combinatorics of Finite Words

Some famous classes of finite words:

- palindromes: $w^R = w$. Ex. *level*.
- balanced words over two letters (say *a* and *b*): all the factors of the same length have the same number of *a*'s (and of *b*'s) up to 1. Ex. *abaababaabaab*.
- differentiable words: words over $\{1, 2\}$ such that their Run Length Encoding is still a word over $\{1, 2\}$.
  Ex. 2211212212211
- finite prefixes of (right) infinite words: Thue-Morse, Fibonacci, Kolakoski,...
- many many others.

Intersections: 12112121121 is a balanced differentiable palindromic prefix of the Fibonacci word over $\{1, 2\}$...

What's the target?

Target

Classify the words through their combinatorial properties.

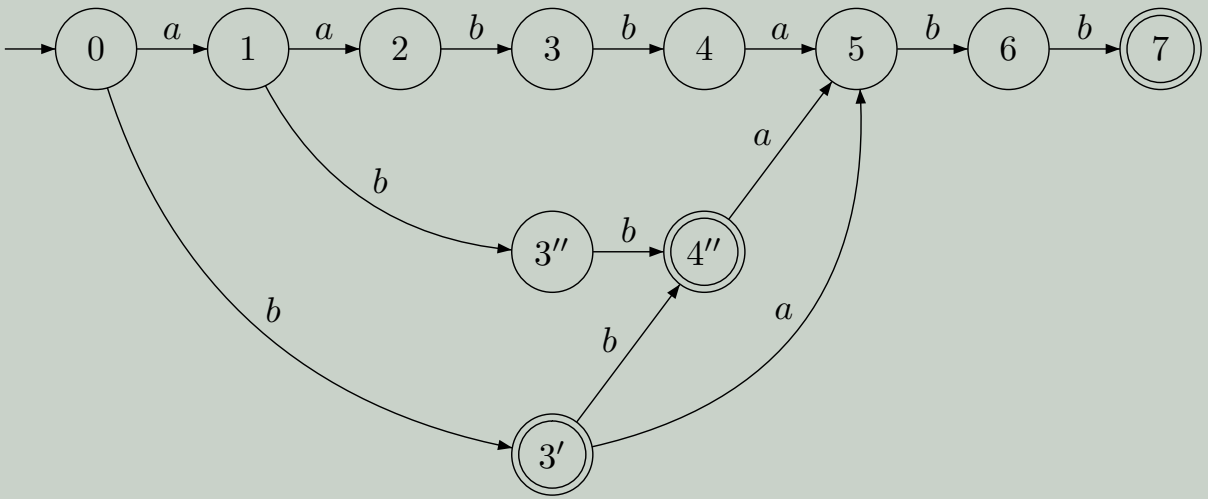
The suffix automaton

Definition (Blumer et al. 1985 - Crochemore 1986)

The suffix automaton of the word w is the minimal deterministic automaton recognizing the suffixes of w.

Example

The suffix automaton of *aabbabb*:



Algorithmically

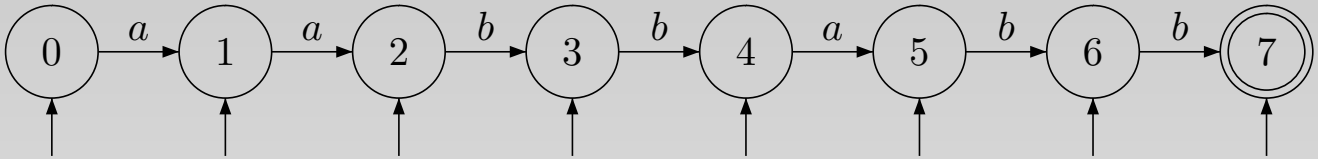
Theorem (Blumer et al. 1985 - Crochemore 1986)

The suffix automaton of a word w over a fixed alphabet A can be built in time and space $O(|w|)$.

One way to build the SA

Build a non-deterministic automaton:

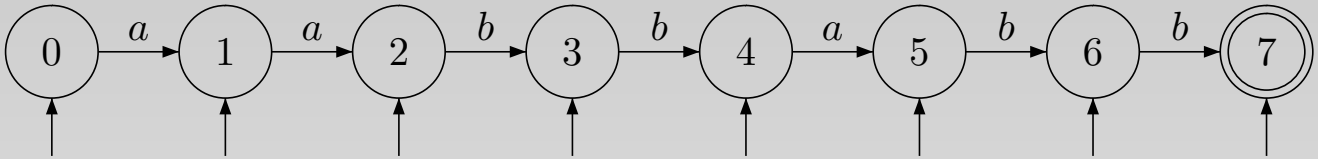
$$w = aabbabb$$



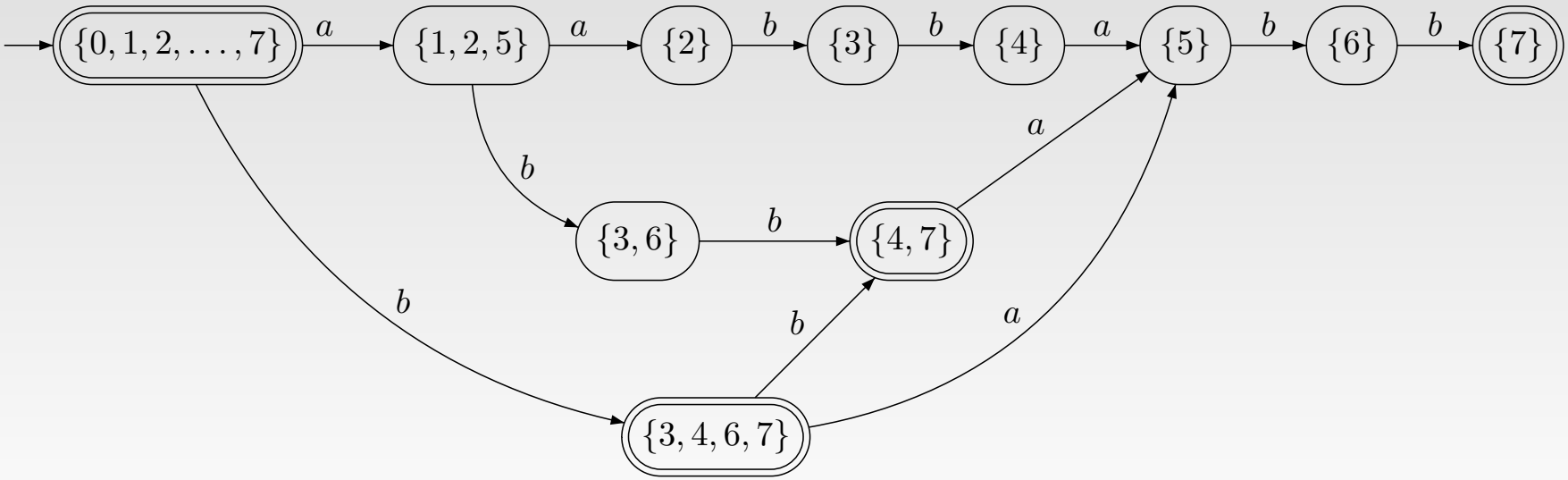
One way to build the SA

Build a non-deterministic automaton:

$$w = aabbabb$$



Determinize by subset construction:



Ending Positions

We associate to each factor v of w the set of ending positions of v in w.

Example

$$\begin{array}{c} w = a\ a\ b\ b\ a\ b\ b \\ 1\ 2\ 3\ 4\ 5\ 6\ 7 \end{array}$$

$Endset(b) = \{3, 4, 6, 7\}$, $Endset(abb) = Endset(bb) = \{4, 7\}$.

Ending Positions

We associate to each factor v of w the set of ending positions of v in w.

Example

$$\begin{array}{c} w = a\ a\ b\ b\ a\ b\ b \\ 1\ 2\ 3\ 4\ 5\ 6\ 7 \end{array}$$

$Endset(b) = \{3, 4, 6, 7\}$, $Endset(abb) = Endset(bb) = \{4, 7\}$.

We define on Fact(w) the equivalence:

$$u \sim v \Leftrightarrow Endset(u) = Endset(v)$$

Ending Positions

We associate to each factor v of w the set of ending positions of v in w.

Example

$$\begin{array}{c} w = a\ a\ b\ b\ a\ b\ b \\ 1\ 2\ 3\ 4\ 5\ 6\ 7 \end{array}$$

$Endset(b) = \{3, 4, 6, 7\}$, $Endset(abb) = Endset(bb) = \{4, 7\}$.

We define on Fact(w) the equivalence:

$$u \sim v \Leftrightarrow Endset(u) = Endset(v)$$

Then Fact(w)$/\sim$ is the set of states of the SA of w.

The number of states

The number of states (classes) of the SA is noted $|Q_w|$.

The bounds on $|Q_w|$ are well known:

$$|w| + 1 \leq |Q_w| \leq 2|w| - 1$$

The number of states

The number of states (classes) of the SA is noted $|Q_w|$.

The bounds on $|Q_w|$ are well known:

$$|w| + 1 \leq |Q_w| \leq 2|w| - 1$$

The upper bound is reached for $w = ab^{|w|-1}$, with $a \neq b$.

The number of states

The number of states (classes) of the SA is noted $|Q_w|$.

The bounds on $|Q_w|$ are well known:

$$|w| + 1 \leq |Q_w| \leq 2|w| - 1$$

The upper bound is reached for $w = ab^{|w|-1}$, with $a \neq b$.

And for the lower bound?

The number of states

The number of states (classes) of the SA is noted $|Q_w|$.

The bounds on $|Q_w|$ are well known:

$$|w| + 1 \leq |Q_w| \leq 2|w| - 1$$

The upper bound is reached for $w = ab^{|w|-1}$, with $a \neq b$.

And for the lower bound?

Problem

Characterize the class of words for which $|Q_w| = |w| + 1$.

Special Factors

Definition

- v is a left special factor of w if there exist $a \neq b$ such that av and bv are factors of w.
- v is a right special factor of w if there exist $a \neq b$ such that va and vb are factors of w.
- v is a bispecial factor of w if it is both left and right special.

Special Factors

Definition

- v is a left special factor of w if there exist $a \neq b$ such that av and bv are factors of w.
- v is a right special factor of w if there exist $a \neq b$ such that va and vb are factors of w.
- v is a bispecial factor of w if it is both left and right special.

Example ($w = aabbabb$)

$LS = \{\varepsilon, a, b, ab, abb\}$, $RS = \{\varepsilon, a, b\}$, $BIS = \{\varepsilon, a, b\}$

The number of states

Theorem (Sciortino, Zamboni 2007)

If $|A| = 2$ then the following conditions are equivalent for a word over A:

- $|Q_w| = |w| + 1$
- *Every left special factor of w is a prefix of w*
- *w is a prefix of a standard sturmian word.*

The number of states

Theorem (Sciortino, Zamboni 2007)

If $|A| = 2$ then the following conditions are equivalent for a word over A:

- $|Q_w| = |w| + 1$
- *Every left special factor of w is a prefix of w*
- *w is a prefix of a standard sturmian word.*

Without restriction on the cardinality of A we have the formula:

Lemma

$$|Q_w| = |w| + 1 + |D(w)|$$

where $D(w)$ is the set of left special factors which are not prefixes.

Property *LSP*

Definition

A word has property *LSP* if every left special factor is a prefix.

Property *LSP*

Definition

A word has property *LSP* if every left special factor is a prefix.

Corollary

$$|Q_w| = |w| + 1 \iff w \text{ has property LSP}$$

Property *LSP*

Definition

A word has property *LSP* if every left special factor is a prefix.

Corollary

$$|Q_w| = |w| + 1 \iff w \text{ has property LSP}$$

Problem

Characterize the class of words having the property LSP, over an arbitrary fixed alphabet A.

The binary case

For binary words we have the formula:

$$|Q_w| = 2|w| - H_w - P_w$$

H_w is the minimal length of a prefix of w occurring only once,
P_w is the maximal length of a left special prefix of w.

The binary case

For binary words we have the formula:

$$|Q_w| = 2|w| - H_w - P_w$$

H_w is the minimal length of a prefix of w occurring only once, P_w is the maximal length of a left special prefix of w.

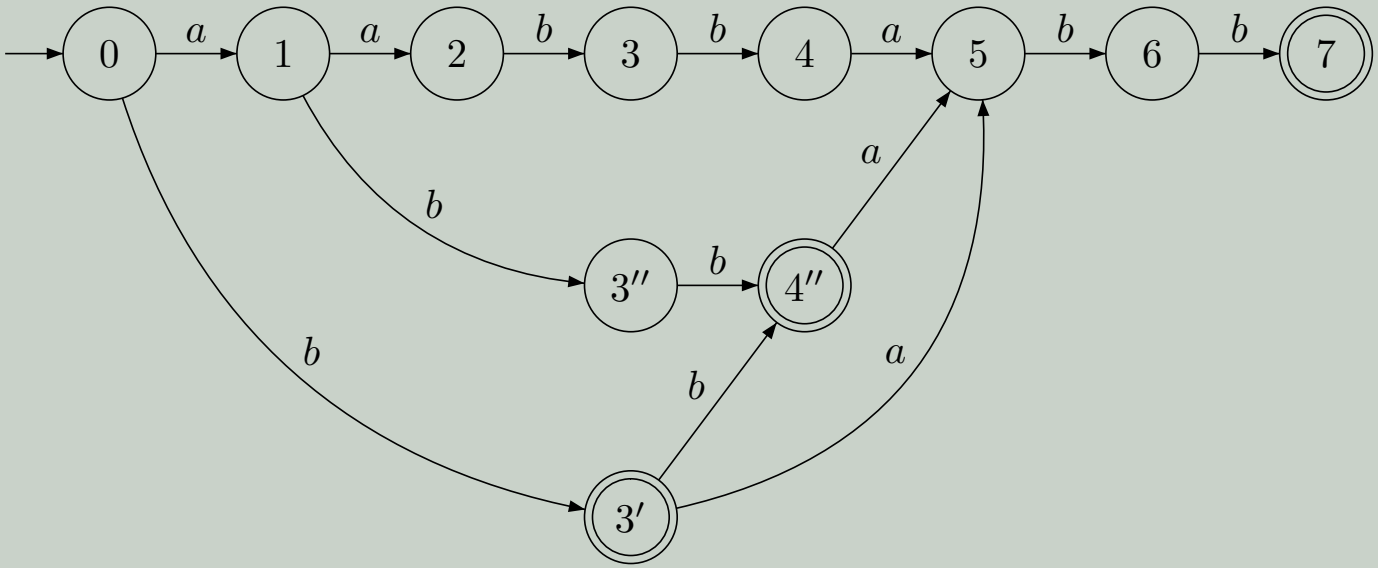
As a corollary we obtain a new characterization of standard sturmian words:

Corollary

w is a prefix of a stand. sturm. word $\Leftrightarrow |w| = H_w + P_w + 1$.

Example

Example ($w = aabbabb$)



$H_w = 2$ since aa occurs only once.
$P_w = 1$ since a is left special.

$$|Q_w| = 2 \cdot 7 - 2 - 1 = 11$$

The number of edges

What about the number of edges $\mathcal{E}_w$?

The number of edges

What about the number of edges $\mathcal{E}_w$?

The bounds on $\mathcal{E}_w$ are well known:

$$|w| \leq \mathcal{E}_w \leq 3|w| - 4$$

The number of edges

What about the number of edges $\mathcal{E}_w$?

The bounds on $\mathcal{E}_w$ are well known:

$$|w| \leq \mathcal{E}_w \leq 3|w| - 4$$

For binary words we give the formula:

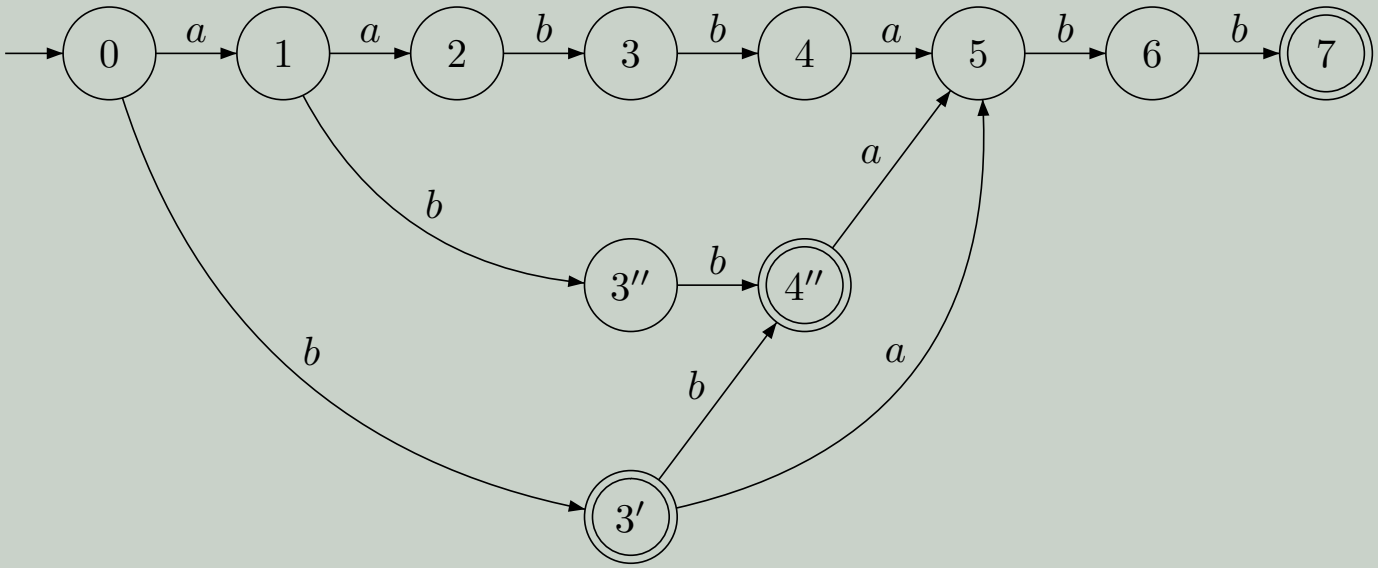
Lemma

$$\mathcal{E}_w = |Q_w| + |G(w)| - 1$$

$G(w)$ is the union of the sets of bispecial factors and right special prefixes of w.

Example

Example ($w = aabbabb$)



$$G(w) = BIS(w) \cup (Pref(w) \cap RS(w)) = \{\varepsilon, a, b\} \cup \{\varepsilon, a\}$$

$$|G(w)| = 3 \quad \Rightarrow \quad \mathcal{E}_w = 11 + 3 - 1 = 13.$$

Further Research

Problem

Does this approach can be applied to other data structures (factor oracle, suffix tree/trie, suffix array, etc.)?

Problem

Characterize the words having property LSP (i.e. every left special factor is a prefix).

Problem

Compute the average size of the SA for particular class of words.