UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

# Bubble-Flip – A New Generation Algorithm for Prefix Normal Words

*Ferdinando Cicalese, Zsuzsanna Lipták and* ***Massimiliano Rossi***

***Department of Computer Science – University of Verona***

*LATA 2018*

***9-11 Apr 2018, Bar-Ilan***

# Prefix Normal Words

**Definition**

**Prefix Normal Words**

binary words

no factor has more 1s than the prefix of the same length

*Fici, Lipták, "On prefix normal words". DLT* ***[2011]***

$w = \overbrace{\underbrace{11010}_{5}}^{3}00\overbrace{\underbrace{11011}_{5}}^{4}0$ **not** prefix normal

There is a factor of **length 5** with >3 **1s.**

**Definition**

## Prefix Normal Words

binary words

no factor has more 1s than the prefix of the same length

*Fici, Lipták, "On prefix normal words". DLT* ***[2011]***

#1s 3 4

$w=$ 1101000110110 **not** prefix normal

length 5 5

$u=$ 1101000100110 **prefix normal**

Is there a factor of **length** 1 with >1 **1s?** **no**
Is there a factor of **length** 2 with >2 **1s?** **no**
Is there a factor of **length** 3 with >2 **1s?** **no**
⋮
Is there a factor of **length** 7 with >3 **1s?** **no**
⋮

# Set of prefix normal words

$\mathcal{L}$ : set of all finite prefix normal words

$\mathcal{L}_n$ : set of all prefix normal words of length $n$

$\mathcal{L}_3$

*000*
*100*
*101*
*110*
*111*

$\mathcal{L}_4$

*0000*
*1000*
*1001*
*1010*
*1100*
*1101*
*1110*
*1111*

$\mathcal{L}_5$

*00000*
*10000*
*10001*
*10010*
*10100*
*10101*
*11000*
*11001*
*11010*
*11011*
*11100*
*11101*
*11110*
*11111*

# Basic Facts

Let $w \in \mathcal{L}$. Then

All prefixes of $w$ are in $\mathcal{L}$

$w0 \in \mathcal{L}$

$\mathcal{L}$: set of all finite prefix normal words
$\mathcal{L}_n$: set of all finite prefix normal words of length $n$

#1s

length

$$w' = \underbrace{\overbrace{w_1 w_2 \cdots w_j w_{j+1}}^{k}}_{j+1} \cdots \underbrace{\overbrace{w_{n-j+1} \cdots w_n 0}^{m \le k}}_{j+1}$$

$w1 \in \mathcal{L}$ if and only if for all $1 \le j < n$, we have

$$|w_1 \cdots w_{j+1}|_1 > |w_{n-j+1} \cdots w_n|_1$$

$$w'' = \underbrace{\overbrace{w_1 w_2 \cdots w_j w_{j+1}}^{k}}_{j+1} \cdots \underbrace{\overbrace{w_{n-j+1} \cdots w_n 1}^{> k}}_{j+1}$$

## A new generation algorithm

**Generation algorithm**

*i.e.*, an algorithm for visiting each object in $\mathcal{L}_n$

**Previous generation algorithm**

- Runs in amortized $O(n)$ time per word
- Conjectured run in amortized $O(\log n)$ time per word, but still open

*Burcsi, Fici, Lipták, Ruskey, Sawada, CPM* ***[2014]***

**New generation algorithm Bubble-Flip**

- Runs in $O(n)$ time per word
- Gives **new insights into properties** of prefix normal words
- Makes steps towards answering conjecture from ***CPM [2014]***

*Cicalese, Lipták, Rossi, LATA* ***[2018]***

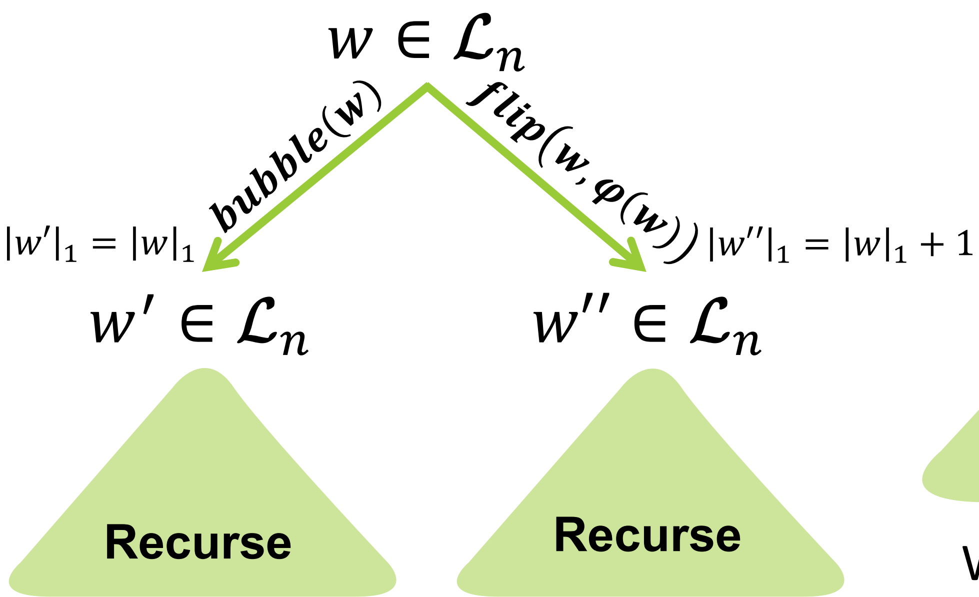
# The Bubble-Flip Algorithm

# Basic Idea

**Generation algorithm** visits each object in $\mathcal{L}_n$ exactly once

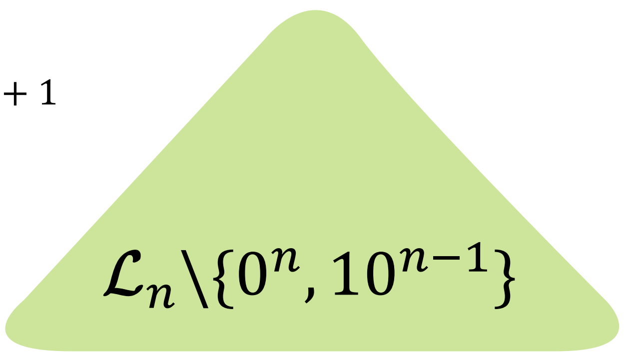
**Based on** two operations $\boldsymbol{bubble(w)}$ and $\boldsymbol{flip(w, \varphi(w))}$

**Recursive generation algorithm**




Starting from

$w$=$110^{n-2}$




We visit each object in $\mathcal{L}_n$

# Operation $bubble(w)$

$\boldsymbol{bubble(w)}$**:** move rightmost 1 by one to the right

$$w = 110100010011\color{red}{10}\color{black}{000} \xrightarrow{bubble(w)} 110100010011\color{red}{01}\color{black}{000}$$

**Lemma**

Let $w \in \mathcal{L}_n$. Then $bubble(w) \in \mathcal{L}_n$

$r$ Rightmost one

$$w = \overbrace{\underbrace{w_1 w_2 \cdots w_j}_{j}}^{k} \cdots \overbrace{\underbrace{w_{r-j-1} w_{r-j} \cdots w_{r-1} \color{red}{1}}_{j}}^{m \le k} \color{red}{0}\color{black}{00000}$$

$$\Big\downarrow bubble(w)$$

$$w' = \overbrace{\underbrace{w_1 w_2 \cdots w_j}_{j}}^{k} \cdots w_{r-j-1} \overbrace{\underbrace{w_{r-j} \cdots w_{r-1} \color{red}{01}}_{j}}^{\le m \le k} 00000$$

# Operation $flip(w, \varphi(w))$

**Phi $\boldsymbol{\varphi}(w)$:** Minimum position $j$ after the rightmost 1 such that $flip(w, j)$ is prefix normal (if such a $j$ exists)

Rightmost one $r$

$w$ = 1101000100110000 $\xrightarrow{flip(w, r+1)}$ 1101000100111000

$w$ = 1101000100110000 $\xrightarrow{flip(w, r+2)}$ 1101000100110100

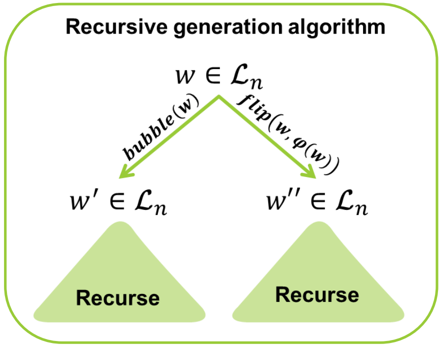
$w$ = 1101000100110000 $\xrightarrow{flip(w, r+3)}$ 1101000100110010

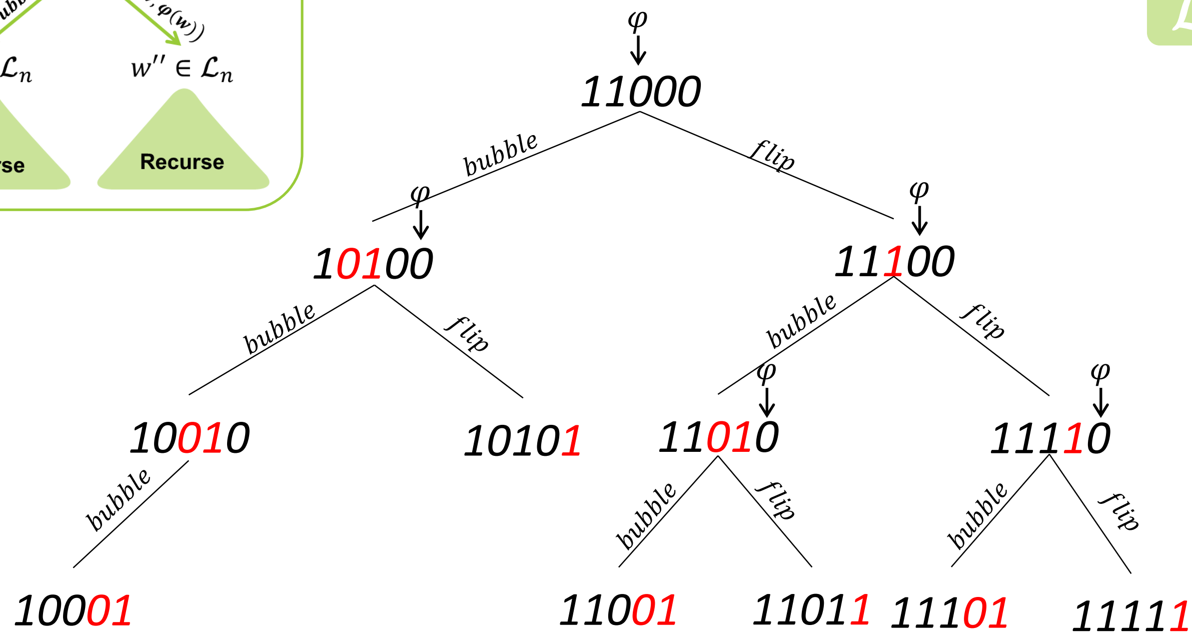
$$\varphi(w) = r + 3$$

**Fact**

Let $w \in \mathcal{L}_n$. Then $flip(w, \varphi(w)) \in \mathcal{L}_n$ $(by\ definition\ of\ \varphi(w))$

# Bubble-Flip algorithm




In-order traversal

$\mathcal{L}_5$

00000
10000
10001
10010
10100
10101
11000
11001
11010
11011
11100
11101
11110
11111

The in-order traversal visits the words in lexicographic order

The post-order traversal generates a **Gray code**

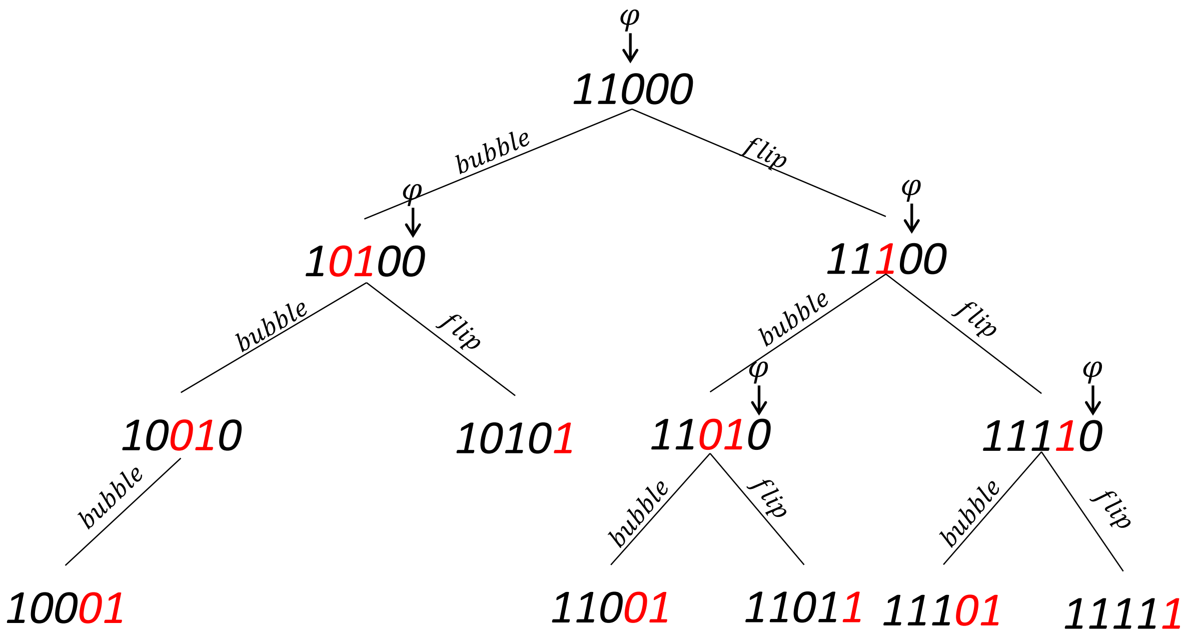
# Computation of $\varphi$

**Phi $\boldsymbol{\varphi(w)}$:** Minimum position $j$ after the rightmost 1 such that $flip(w, j)$ is prefix normal (if such a $j$ exists)

- **Easy** in quadratic time
- Can be computed in **linear** time:

$$w' = \overbrace{\underbrace{w_1 w_2 \cdots w_j}_{j}\underbrace{0 \cdots 0}_{k}}^{m} \cdots \overbrace{\underbrace{w_{r-j+1} \cdots w_r}_{j}\underbrace{0 \cdots 0}_{< k}\cancel{0}}^{m+1} 000000$$

# Analysis of Bubble-Flip algorithm




**Computation cost** (without outputting) **per word:**

- $bubble(w)$: $O(1)$
- $\varphi(w)$ : $\boldsymbol{O(n)}$
- $flip\big(w,\varphi(w)\big)$: $O(1)$

# Infinite prefix normal words

# Infinite prefix normal words and operation flipext

**Definition**

**Infinite Prefix Normal Words**

infinite binary words

no factor has more 1s than the prefix of the same length

Let $w \in \mathcal{L}$. Then $w0^{\omega}$ is an **infinite prefix normal word**

**Operation** $flipext(w)$ — *extension version of* $flip(w, \varphi(w))$

Let $w \in \mathcal{L}$. $flipext(w)$ is the finite word $w0^k1$, where $k = \min\{j \mid w0^j1 \in \mathcal{L}\}$.

The infinite word $v = flipext^{\omega}(w) = \lim_{i \to \infty} flipext^{(i)}(w)$

$w$= 110100010011 $\xrightarrow{flipext^{(1)}(w)}$ 110100010011001

110100010011001001001100100100110010010011001001 $\cdots$

# Minimum density

Let $w \in \{0,1\}^* \cup \{0,1\}^\omega$ . Then

**Density**

$$D(i) = \frac{|w_1 \cdots w_i|_1}{i}$$

**Minimum density**

$$\rho(w) = inf\{D(i) | 1 \leq i\}$$

**Iota** *(minimum density prefix)*

$$\iota(w) = \min\{j \mid \forall i : D(j) \leq D(i)\}$$ if it exists

$$\kappa(w) = 4$$

$w = \boxed{1101000100}11$

$$\iota(w) = 10$$

Minimum density

$$\rho(w) = \frac{4}{10}$$

$v$

10101010⋯ $\iota(v) = 2$ $\rho(v) = \frac{1}{2}$

$u$

110101010⋯ $\iota(u)$ is undefined $\rho(u) = \frac{1}{2}$

# Results on infinite words generated by $flipext^{\omega}(w)$

Let $w \in \mathcal{L}$ and $v = flipext^{\omega}(w)$. Then

$\boldsymbol{\rho}(\boldsymbol{v}) = \boldsymbol{\rho}(\boldsymbol{w})$, and as consequence $\boldsymbol{\iota}(\boldsymbol{v}) = \boldsymbol{\iota}(\boldsymbol{w})$ and $\boldsymbol{\kappa}(\boldsymbol{v}) = \boldsymbol{\kappa}(\boldsymbol{w})$

$v$ is the **densest** prefix normal word with $\boldsymbol{w}$ as prefix

$v$ is **ultimately periodic**. In particular,

- $v$ can be written as $\boldsymbol{v} = \boldsymbol{u}\boldsymbol{x}^{\boldsymbol{\omega}}$

where $|x| = \iota(w)$ , $|x|_1 = \kappa(w)$ and $x$ is prefix normal.

Moreover if $x$ not a suffix of $u$

- $|u| \leq \left(\binom{\iota}{\kappa} - 1\right) m\, \iota$

where $\iota = \iota(w)$, $\kappa = \kappa(w)$ and $m = \left\lfloor \frac{|w|}{\iota} \right\rfloor$

# Other results on Infinite Prefix Normal Words

- There exists an infinite prefix normal words $\boldsymbol{v}$ with $\boldsymbol{\rho}(\boldsymbol{v}) = \boldsymbol{\rho}$ for every $\boldsymbol{\rho} \in \mathbb{R}$
- If $\boldsymbol{v}$ is **ultimately periodic** then $\boldsymbol{\rho}(\boldsymbol{v})$ is **rational**
- If $\boldsymbol{\rho}(\boldsymbol{v})$ is **irrational** then $\boldsymbol{v}$ is **aperiodic**
- Connection with Sturmian words.

**(Submitted)**

# Open Problems

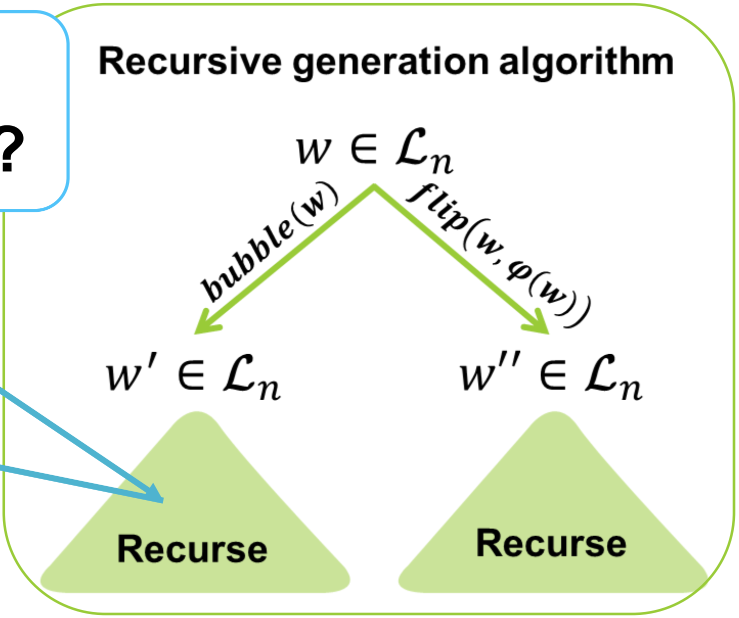
# Open problems – Counting Prefix Normal Words

**1. Counting Prefix Normal Words:** $|\mathcal{L}_n|$ ? (OEIS seq.no. A194850)

*TCS* ***[2017]***

**a. How many nodes** are there**?**

How many prefix normal words of length $n$ with a given prefix are there**?**




**b. How many** prefix normal words with **critical prefix** $1^s 0^t$ are there?

# Open problems – **Generation Algorithm**

**2. Generation Algorithm**

a. Sublinear time per word?
b. CAT (Constant Amortized Time per word)?
c. Testing for $w \in \mathcal{L}$ if $w1 \in \mathcal{L}$ in sublinear time?

**3. Can we test $w \in \mathcal{L}$ in $\boldsymbol{O\left(n^{1.5}\right)}$?**

# THANK YOU!

# Iota-factorization

**Iota-factorization** of $w$

Is the factorization of $w$ into $\iota$-length factors.

$w = u_1 u_2 \cdots u_r v$ and where $r = \lfloor |w|/\iota \rfloor$, $|u_i| = \iota$ for $i = 1, \dots, r$, and $|v| < \iota$, for $w$ finite,

$w = u_1 u_2 \cdots$ and where $|u_i| = \iota$ for all $i$, for $w$ infinite.

$$\overbrace{1101000100}^{\iota(w)=10}\overbrace{1100100100}^{\iota(w)}\overbrace{1100100100}^{\iota(w)}\overbrace{1100100100}^{\iota(w)}\overbrace{1100100100}^{\iota(w)}$$

$$\underbrace{1101000100}_{\kappa(w)=4}\underbrace{1100100100}_{\kappa(w)}\underbrace{1100100100}_{\kappa(w)}\underbrace{1100100100}_{\kappa(w)}\underbrace{1100100100}_{\kappa(w)}$$

## Lemma

Let $w$ be a finite or infinite prefix normal word, such that $\iota = \iota(w)$ exists. Let $w = u_1 u_2 \cdots$ be the Iota-factorization of $w$. **Then for all $\boldsymbol{i}$, $\boldsymbol{u_i \geq_{lex} u_{i+1}}$.**

# Iota-factorization properties

## Lemma

Let $w$ be a finite or infinite prefix normal word, such that $\iota = \iota(w)$ exists. Let $w = u_1 u_2 \cdots$ be the Iota-factorization of $w$. **Then for all $\boldsymbol{i}$, $\boldsymbol{u_i \geq_{lex} u_{i+1}}$.**

$\leq \kappa$

$$w = \quad \cdots \; \underbrace{a_1 a_2 \cdots a_j a_{j+1} \cdots a_\iota}_{u_i} \underbrace{b_1 b_2 \cdots b_j b_{j+1} \cdots b_\iota}_{u_{i+1}} \cdots$$

Let $\mathrm{h} = \min\{\mathrm{j} \; || a_{j+1} \cdots a_\iota b_1 b_2 \cdots b_j |_1 < \kappa\}$ then it holds that $a_h > b_h$

Since two consecutive factors have the same prefix, the first difference must be a 1 that turns into a 0. Then $\boldsymbol{u_i \geq_{lex} u_{i+1}}$.