Prefix normal words, binary jumbled pattern matching, and bubble languages

Péter Burcsi, Gabriele Fici, Zsuzsanna Lipták, Frank Ruskey, and Joe Sawada

LSD/LAW 2014
London, 6-7 Feb. 2014

How marrying two topics can lead to an explosion of results

Outline

- def 1: prefix normal words
- motivation: binary jumbled pattern matching
- def 2: bubble languages
- the marriage
- $\rightsquigarrow$ generation algorithm, enumeration results, testing algorithm, experimental results, new insights, and, and, and ...

Prefix Normal Words

Prefix normal words

Fici, L. (DLT 2011)

Definition

A binary word w is prefix normal (w.r.t. 1) if $\forall\ 1 \leq k \leq |w|$, no substring of length k has more 1s than the prefix of length k.

Example

$$w = 1011100100111110010$$

$$w' = 1110100101100101001 0$$

Prefix normal words

Fici, L. (DLT 2011)

Definition

A binary word w is prefix normal (w.r.t. 1) if $\forall\ 1 \leq k \leq |w|$, no substring of length k has more 1s than the prefix of length k.

Example

$$w = 1011100100111110010 \quad NO$$

$$w' = 1110100101100101001 0 \quad YES$$

Prefix normal words

Fici, L. (DLT 2011)

Definition

A binary word w is prefix normal (w.r.t. 1) if $\forall\ 1 \leq k \leq |w|$, no substring of length k has more 1s than the prefix of length k.

Example

$$w = 1011100100111110010 \quad NO$$

$$w' = 1110100101100101010 \quad YES$$

$\mathcal{L}_{\mathrm{PN}}$ = all prefix normal words.
Exists canonical prefix normal form of w: $\mathrm{PNF}_1(w)$.

Binary Jumbled Pattern Matching (BJPM)

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones? (Online: easy. Indexed: ?)

Binary Jumbled Pattern Matching (BJPM)

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones?

(PSC'09, LSD/LAW, FUN 2010, IPL 2010, JDA 2012, ToCS 2012, IJFCS 2012, CPM'12, IPL 2013 x 2, SPIRE'12, ESA'13 x 2, SPIRE'13, arxiv 2014 x 3)

Binary Jumbled Pattern Matching (BJPM)

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones?

Interval property $\rightsquigarrow$ linear size index:

Fix length k of substrings: no. 1s builds an interval.

Ex: $k = 4 : 1, 2, 3$ ones.

For each k, store max and min no. of 1s.

Binary Jumbled Pattern Matching (BJPM)

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones?

Interval property $\rightsquigarrow$ linear size index:

Fix length k of substrings: no. 1s builds an interval.

Ex: $k = 4 : 1, 2, 3$ ones.

For each k, store max and min no. of 1s.

k	1	2	3	4	5	...	11
max	1	2	3	3	4	...	...
min	0	0	0	1	2	...	...

Binary Jumbled Pattern Matching (BJPM)

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones?

Interval property $\rightsquigarrow$ linear size index:

Fix length k of substrings: no. 1s builds an interval.
Ex: $k = 4 : 1, 2, 3$ ones.
For each k, store max and min no. of 1s.

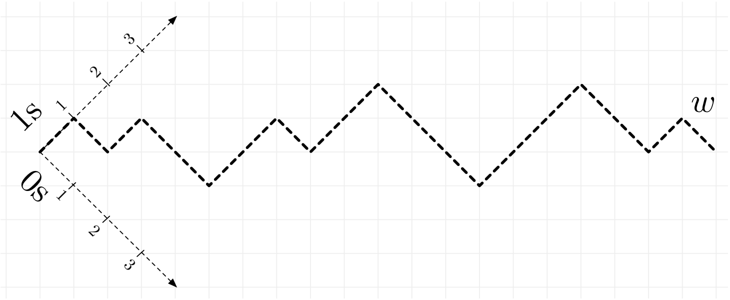
k	1	2	3	4	5	...	11
max	1	2	3	3	4	...	...
min	0	0	0	1	2	...	...

Research problem:

Compute this index efficiently.

BJPM with prefix normal forms

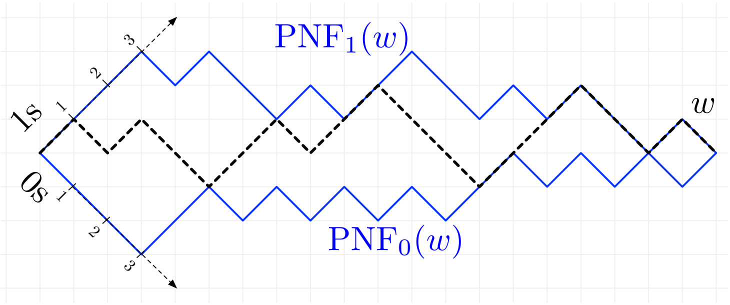
Does $\mathbf{w} = \mathbf{1010011011000111 0010}$ have a substring of length 11 containing exactly 5 ones?



$\diagup = 1$, $\diagdown = 0$,

BJPM with prefix normal forms

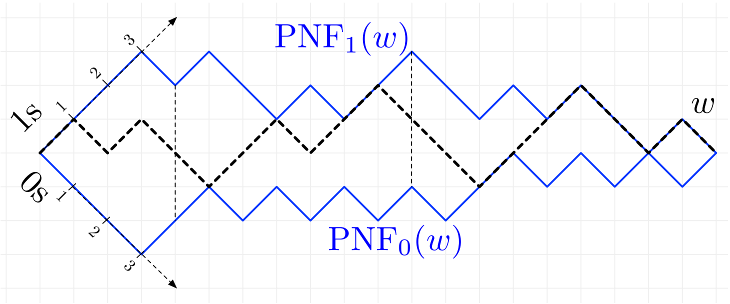
Does **w** = **10100110110001110010** have a substring of length 11 containing exactly 5 ones?



╱ = 1, ╲ = 0, Blue: prefix normal forms of w

BJPM with prefix normal forms

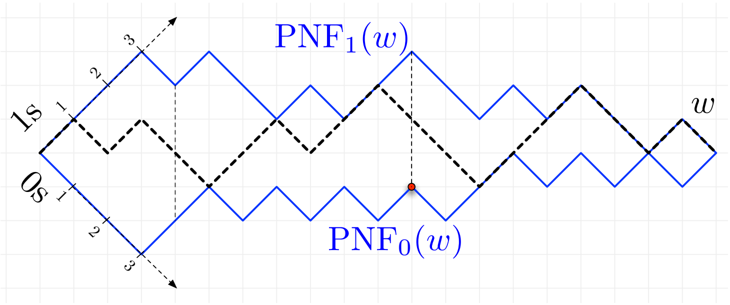
Does **w** = **101001101100011100010** have a substring of length 11 containing exactly 5 ones?



╱ = 1, ╲ = 0, Blue: prefix normal forms of w
verticals: fixed length substrings $k = 4, 11$.

BJPM with prefix normal forms

Does **w** = **10100110110001110010** have a substring of length 11 containing exactly 5 ones? YES



╱ = 1, ╲ = 0, Blue: prefix normal forms of *w*

verticals: fixed length substrings $k = 4, 11$.

BJPM with prefix normal forms

Does $\mathbf{w} = \mathbf{10100110110001110010}$ have a substring of length 11 containing exactly 5 ones? YES

no. 1s in pref$(\mathrm{PNF}_1(w), 11) \geq 5 \geq$ no. 1s in pref$(\mathrm{PNF}_0(w), 11)$

Thus, fast computation of PNFs yields fast solution to BJPM.

Bubble Languages

Bubble languages

Ruskey, Sawada, Williams (JCombTh.A, 2012)
Sawada, Williams (ElJComb. 2012)

Definition

A binary language $\mathcal{L}$ is called bubble if, for all $w \in \mathcal{L}$, exchanging the first 01 with 10, results in another word in $\mathcal{L}$.

Example

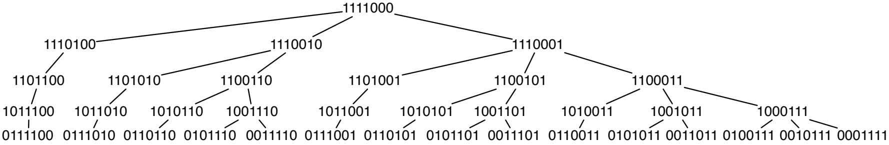
- $\{1001, 1010, 1100, 1000, 0000\}$ – YES
- $\{1001, 1010\}$ – NO

Theorem

$\mathcal{L}_{PN}$ is a bubble language.

An alternative characterization of bubble languages

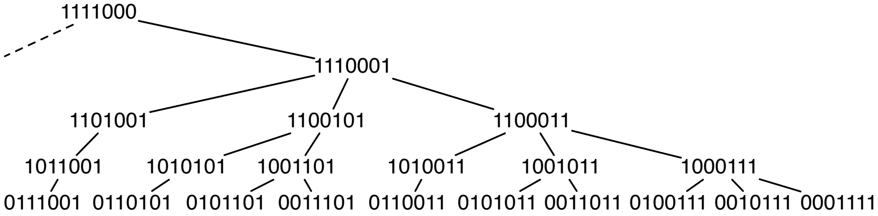
The bubble tree T_d^n on all strings of length n with d ones:



$v = 1^{s}0^{t}\gamma$, children of v: $1^{s-1}0^{i}10^{t-i}\gamma$, for $i = 1, \ldots, t$.

An alternative characterization of bubble languages

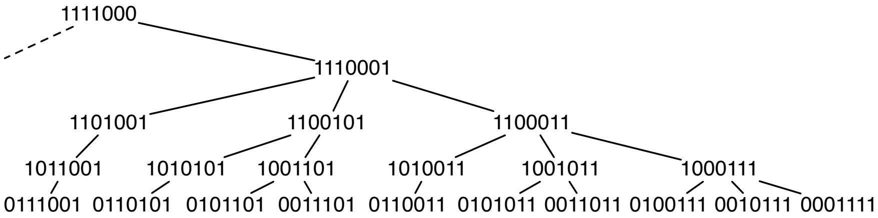
The bubble tree T_d^n on all strings of length n with d ones:



$v = 1^s 0^t \gamma$, children of v: $1^{s-1} 0^i 1 0^{t-i} \gamma$, for $i = 1, \ldots, t$.

An alternative characterization of bubble languages

The bubble tree T_d^n on all strings of length n with d ones:



$v = 1^s 0^t \gamma$, children of v: $1^{s-1} 0^i 1 0^{t-i} \gamma$, for $i = 1, \ldots, t$.

Observation

A language is bubble iff it is left- and up-closed in T_d^n, for all n, d.

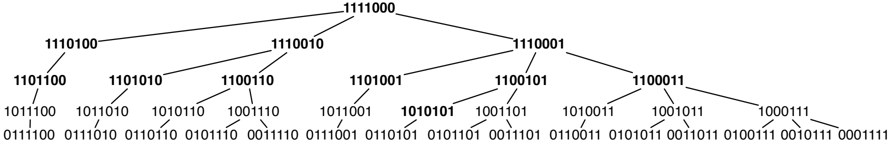
Bubble miracles

Let $\mathcal{L}$ be a bubble language.

- (Fixed-density subsets of) $\mathcal{L}$ are subtrees in the T_d^n's
- Traversal of these subtrees = generation algorithm for $\mathcal{L}$. (enumeration, listing)
- post-order yields a Gray code for $\mathcal{L}$ (cool-lex order)
- Need only: For $w \in \mathcal{L}$, which is the rightmost child still in $\mathcal{L}$? (Oracle for $\mathcal{L}$)
- If Oracle in time $O(f(n) \cdot k)$, where k = rightmost child, then generation algorithm in $O(f(n))$ amortized time per word.

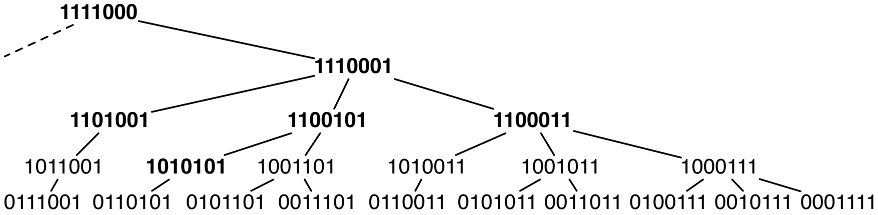
Prefix Normal Words and Bubble Languages

$\mathcal{L}_{\mathrm{PN}}$ in the bubble tree



For every node in $\mathcal{L}_{\mathrm{PN}}$, we need to decide which is rightmost child in $\mathcal{L}_{\mathrm{PN}}$.

$\mathcal{L}_{\mathrm{PN}}$ in the bubble tree



For every node in $\mathcal{L}_{\mathrm{PN}}$, we need to decide which is rightmost child in $\mathcal{L}_{\mathrm{PN}}$.

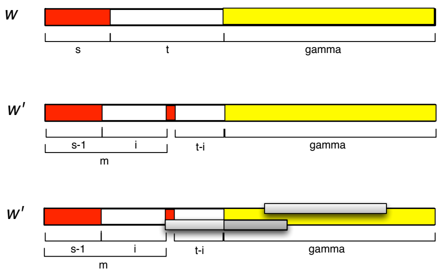
Oracle for $\mathcal{L}_{\text{PN}}$

Theorem

Let $w \in \mathcal{L}_{PN}$ and w' one of its children. Then it can be decided in linear time whether $w' \in \mathcal{L}_{PN}$.

(using some additional data structure, linear time+space)

Proof



Bubble miracles for prefix normal words

- Efficient generation algorithm for $\mathcal{L}_{\mathrm{PN}}$: amortized linear time per word conjectured $O(\log n)$
- Best previous: $O(2^n n^2)$ time; very substantial improvement (no. pn-words grows much slower than 2^n)
- Gray code for $\mathcal{L}_{\mathrm{PN}}$
- enumeration results (experiments)—not possible before!
- many new insights from the bubble property, the generation algorithm, the new representation of prefix normal words
- and, and, and ...

THANK YOU!

http://arxiv.org/abs/1401.6346

zsuzsanna.liptak@univr.it