

Local Repetitions in Strings

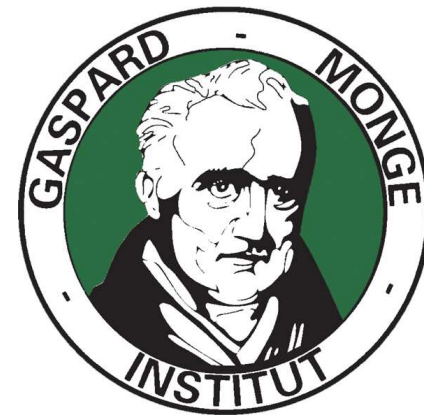
MAXIME CROCHEMORE

King's College London

Université Paris-Est

&

KING'S
College
LONDON



Repetitions and repeats

- ★ String = text = word = sequence of symbols
- ★ Repetition = periodic string = power: exponent ≥ 2

← length = 17 →
a b a a b a b a a b a b a a b a b
← period = 5 →

$$\text{exponent} = \frac{\text{length}}{\text{period}} = \frac{17}{5} = 3.4$$

Repetitions and repeats

- ★ String = text = word = sequence of symbols
- ★ Repetition = periodic string = power: exponent ≥ 2

length = 17
a b a a b a b a a b a b a a b a b
period = 5

$$\text{exponent} = \frac{\text{length}}{\text{period}} = \frac{17}{5} = 3.4$$

- ★ Repeat: $1 \leq \text{exponent} < 2$

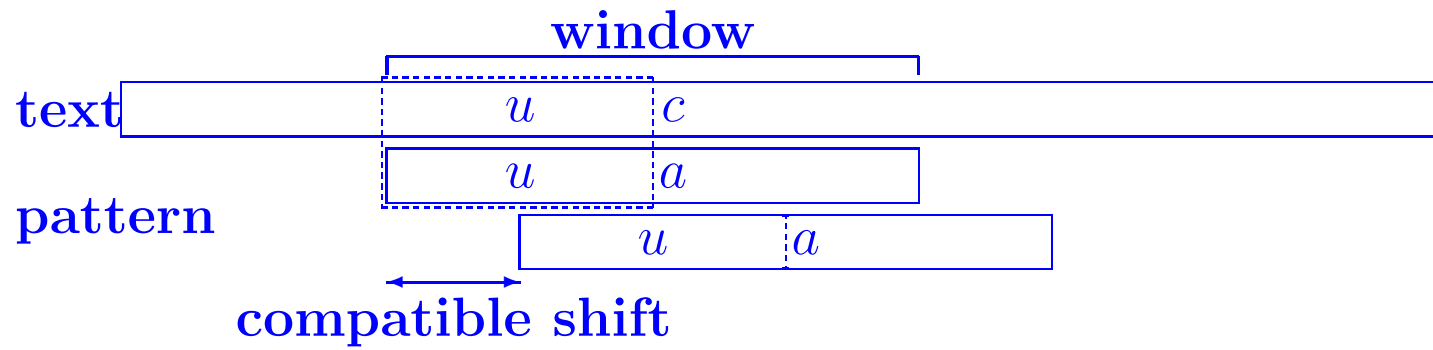
length = 15
a b a a b c c c c c a b a a b
period = 10

$$\text{exponent} = \frac{\text{length}}{\text{period}} = \frac{15}{10} = 1.5$$

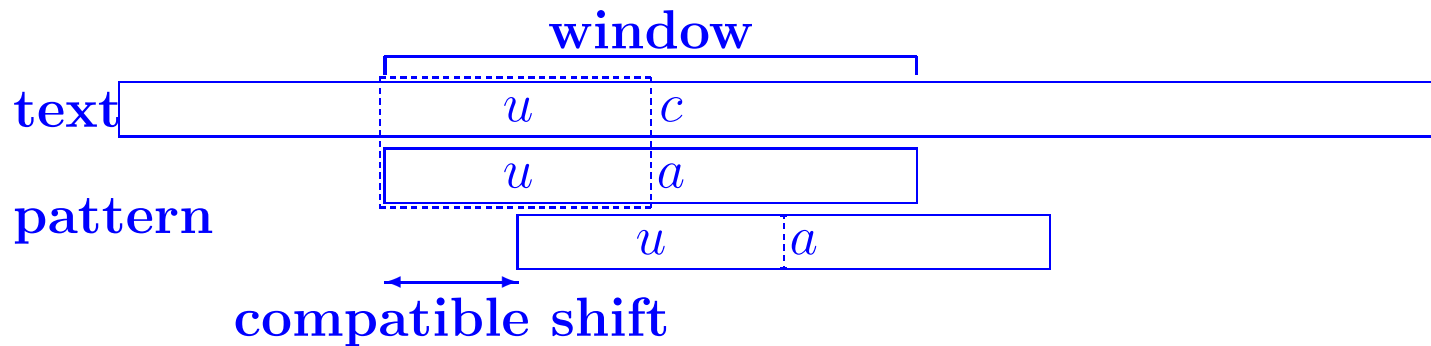
Local periodicities in strings

- ★ **Pattern matching algorithms**
 - String Matching: borders and periods
 - Time-space optimal String Matching: local and global periods
- ★ **Analysis of biological molecular sequences**
 - Intensive study of satellites, Simple Sequence Repeats, or Tandem Repeats in DNA sequences
- ★ **Text Compression**
 - Generalised run-length encoding
 - Dictionary-based compression
- ★ **Combinatorics on words**
 - Avoidability of repetitions
 - Interaction between periods
 - Counting repetitions

On-line String Matching



On-line String Matching



- ★ $\text{shift} = \text{period}(u)$
[Morris, Pratt, 1969]

text . . a b a a b a c
 pattern a b a a b a a
 a b a a b a a

- ★ not incompatible with c
[Knuth et al., 1977]

text . . a b a a b a c
 pattern a b a a b a a
 a b a a b a a

- ★ $\text{best shift} = \text{period}(uc)$
[Hancart, Simon, 1993]
[Breslauer et al., 1993]
alpha-size independent

text . . a b a a b a c
 pattern a b a a b a a
 a b a a b a a

Two-way string matching

Searching for pattern $x = x[0..m-1]$ in text $y = y[0..n-1]$

- ★ **Critical factorisation**

$x = uv$ and $|u| < \text{loc-per}(u, v) = \text{period}(x)$

- ★ **Searching**

shift length = **scan length** or **period of x**

Two-way string matching

Searching for pattern $x = x[0..m-1]$ in text $y = y[0..n-1]$

- ★ **Critical factorisation**

$x = uv$ and $|u| < \text{loc-per}(u, v) = \text{period}(x)$

- ★ **Searching**

shift length = **scan length** or **period of x**

- ★ **Preprocessing**

factorisation (u, v) and (smallest) period of x

- ★ **Theorem 1 ([C., Perrin, 1991])**

Two-way string matching runs in time $O(n)$.

Preprocessing time $O(m)$. Both with constant extra space.

- ★ **Other solutions**

[Galil, Seiferas, 1983], [C., 1992]

[C., Rytter, 1994], [Rytter, 2002]

real-time solution [Breslauer, Grossi, Mignosi, 2011]

Local periodicities in strings

- ★ Pattern matching algorithms
 - String Matching: borders and periods
 - Time-space optimal String Matching: local and global periods
- ★ Analysis of biological molecular sequences
 - Intensive study of satellites, Simple Sequence Repeats, or Tandem Repeats in DNA sequences
- ★ Text Compression
 - Generalised run-length encoding
 - Dictionary-based compression
- ★ Combinatorics on words
 - Avoidability of repetitions
 - Interaction between periods
 - Counting repetitions

Huntington's disease mRNA in EMBL

ID L12392; SV 1; linear; mRNA; STD; HUM; 10348 BP.
...
DE Homo sapiens Huntington's Disease (HD) mRNA, complete cds.
XX
KW trinucleotide repeat.
XX
OS Homo sapiens (human)
OC Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia;
OC Eutheria; Euarchontoglires; Primates; Haplorrhini; Catarrhini; Hominidae;
OC Homo.
XX
RN [1]
RP 1-10348
RX PUBMED; 8458085.
RA MacDonald M., Ambrose C.M.;
RT "A novel gene containing a trinucleotide repeat that is expanded and
RT unstable on Huntington's disease chromosomes. The Huntington's Disease
RT Collaborative Research Group [see comments]";
RL Cell 72(6):971-983(1993).
...

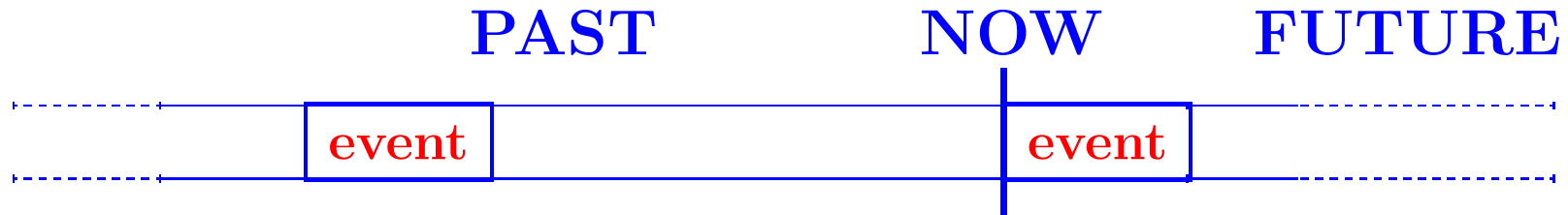
Polyglutamine repetition

```
...
FT   CDS                316..9750
...
FT                               /protein_id="AAB38240.1"
FT                               /translation="MATLEKLMKAFESLKSFQQQQQQQQQQQQQQQQQQQQQQQQQPPPPP
FT                               PPPPPQLPQPPPQAQPLLQPQPPPPPPPPPPGPAVAEEPLHRPKKELSATKKDRVNH
...
FT                               ...
FT                               FQSVLEVVAAPGSPYHRLTCLRNVHKVTTC"
FT   polyA_site         10348
FT                               /gene="HD"
XX
SQ   Sequence 10348 BP; 2408 A; 2807 C; 2744 G; 2389 T; 0 other;
      ttgctgtgtg aggcagaacc tgcgggggca ggggcgggct ggttccttg  ccagccattg          60
      gcagagtccg caggctaggg ctgtcaatca tgctggccgg cgtggccccg cctccgccgg          120
      cgcggccccg cctccgccgg cgcacgtctg ggacgcaagg cgccgtgggg gctgccggga          180
      cgggtccaag atggacggcc gctcaggttc tgcttttacc tgcggcccag agccccattc          240
      attgccccgg tgctgagcgg cgccgcgagt cggcccaggg cctccgggga ctgccgtgcc          300
      gggcgggaga ccgccATGgc gaccctggaa aagctgatga aggccttca gtccctcaag          360
      tccttcCAGC AGCAGCAGCA GCAGCAGCAG CAGCAGCAGC AGCAGCAGCA GCAGCAGCAG          420
      CAGCAGCAGC AACAGccgcc accgccgccg ccgccgccgc cgcctcctca gttcctcag          480
      ccgccgccgc aggcacagcc gctgctgcct cagccgcagc cgccccgccg gccgccccg          540
...
      ...
      atatcagtaa agagattaat tttaacgt                                10348
//
```

Local periodicities in strings

- ★ Pattern matching algorithms
 - String Matching: borders and periods
 - Time-space optimal String Matching: local and global periods
- ★ Analysis of biological molecular sequences
 - Intensive study of satellites, Simple Sequence Repeats, or Tandem Repeats in DNA sequences
- ★ Text Compression
 - Generalised run-length encoding
 - Dictionary-based compression
- ★ Combinatorics on words
 - Avoidability of repetitions
 - Interaction between periods
 - Counting repetitions

Remembering the Past



“Who so neglects learning in his youth, loses the past and is dead for the future.”

Euripides (484 BC - 406 BC)

Ziv-Lempel factorisation

★ Phrase = longest factor occurring before (LPF)

★ Example of $y = \text{abaabababaaababb}$

a b a a b a b a b a a a b a b b

a b a a b a b a b a a a b a b b

★ **Encoding:** phrases are carefully encoded as

(previous position, length)

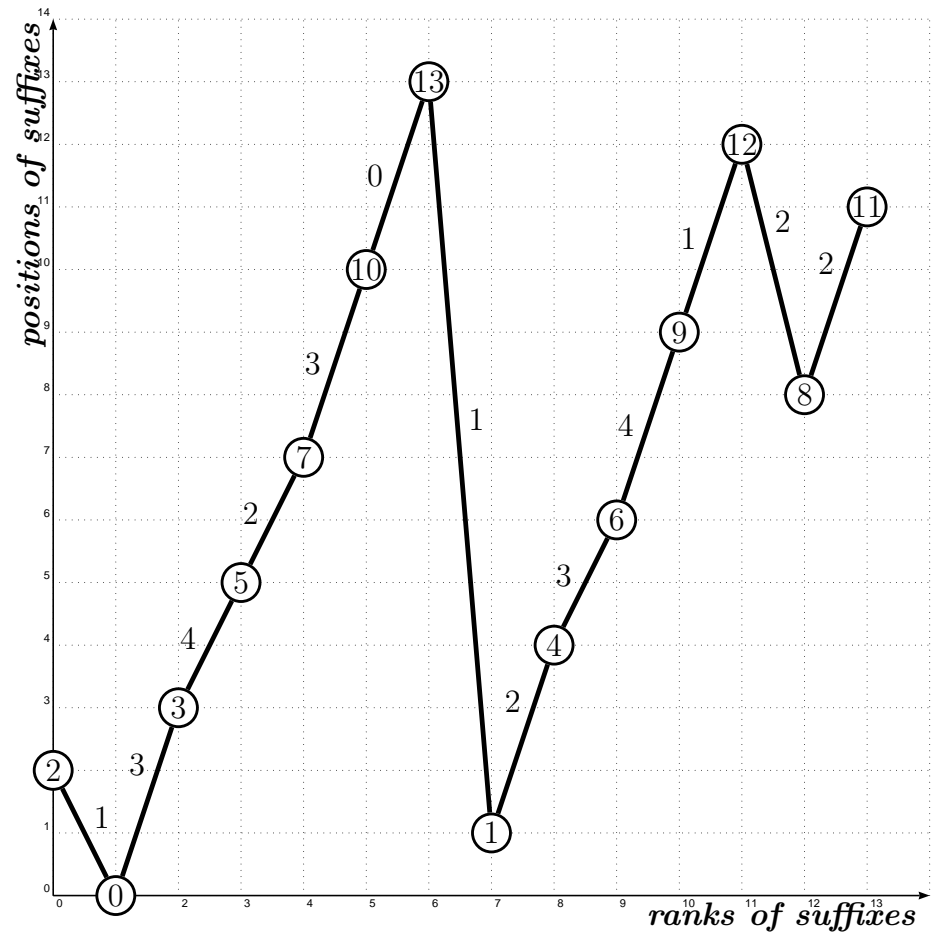
★ **Very efficient:** many variants implemented in compress, gzip, PKzip, rzm, lzturbo, etc.

★ **[Ziv, Lempel, 1977]**

★ **Computation** in time $O(n \log a)$ ($a =$ alphabet size)

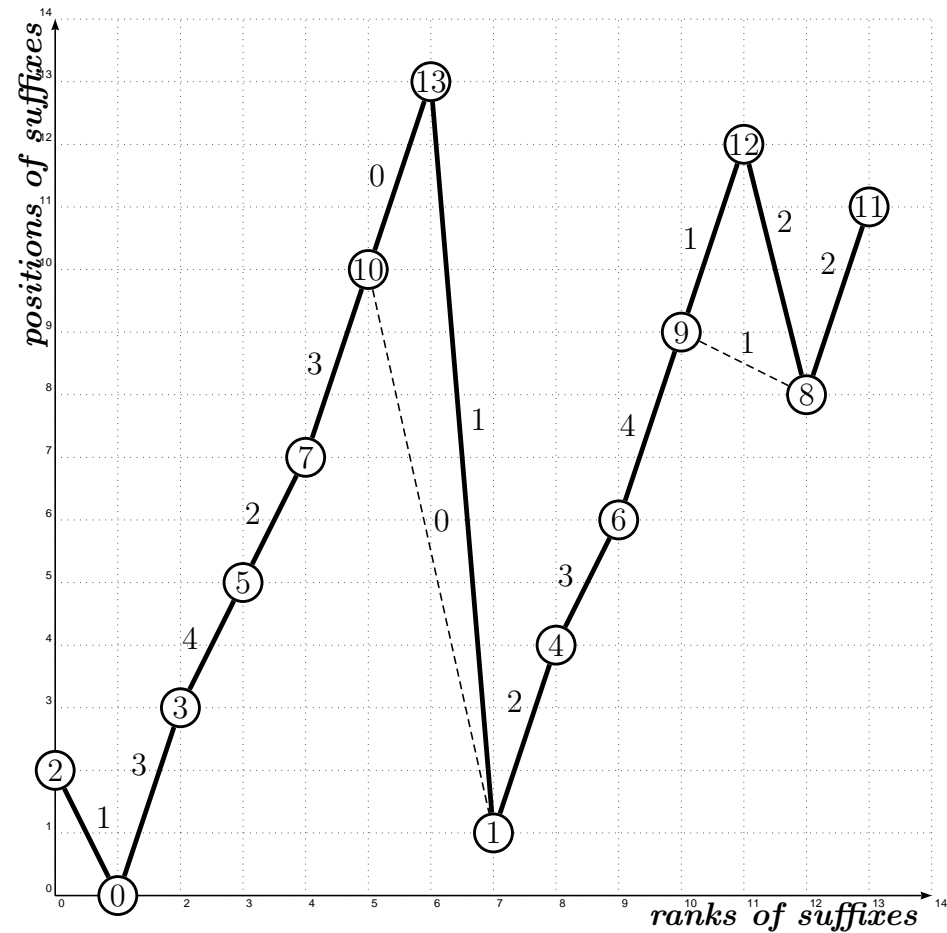
Graphic Representation of a Suffix Array

String	a	b	a	a	b	a	b	a	b	b	a	b	b	b
rank r	0	1	2	3	4	5	6	7	8	9	10	11	12	13
SUF[r]	2	0	3	5	7	10	13	1	4	6	9	12	8	11
LCP[r]	0	1	3	4	2	3	0	1	2	3	4	1	2	2



Basics for computing LPF

rank r	0	1	2	3	4	5	6	7	8	9	10	11	12	13
SUF[r]	2	0	3	5	7	10	13	1	4	6	9	12	8	11
LCP[r]	0	1	3	4	2	3	0	1	2	3	4	1	2	2
LPF[i]	1	0	3	4	2	3	1	0	2	3	4	2	1	2



LPF from Suffix Array

- ★ **Integer alphabet:** sorting letters can be done in linear time
- ★ **Suffix Array construction:** suffix sorting + LCP
 - Linear-time suffix sorting by
[Kärkkäinen, Sanders, 2003], [Ko, Aluru, 2003]
[Kim, Sim, Park, Park, 2003], [Nong, Zhang, Chan, 2009]
 - Linear-time computation of LCP table by
[Kasai, Lee, Arimura, Arikawa, Park, 2001]
- ★ **Computation of LPF table**
 - total linear time + constant space
 - fast and space economical
 - several variants
[C., Ilie, 2007], [C., Ilie, Iliopoulos, Kubica, Rytter, Waleń, 2009], [C., Tischler, 2009], [C., Iliopoulos, Kubica, Rytter, Waleń, 2009], [Chairungsee, C., 2009]

Local periodicities in strings

- ★ Pattern matching algorithms
 - String Matching: borders and periods
 - Time-space optimal String Matching: local and global periods
- ★ Analysis of biological molecular sequences
 - Intensive study of satellites, Simple Sequence Repeats, or Tandem Repeats in DNA sequences
- ★ Text Compression
 - Generalised run-length encoding
 - Dictionary-based compression
- ★ Combinatorics on words
 - Avoidability of repetitions
 - Interaction between periods
 - Counting repetitions

Avoiding repetitions

★ **Theorem 2 ([Thue, 1906, 1912])**

There are infinite binary strings with no overlap (that is, no repetition of exponent > 2).

There are infinite ternary strings with no square.

★ **Iterated morphisms**

– no overlap in t :

$$\begin{cases} t(0) = 01, \\ t(1) = 10. \end{cases}$$

$$t = t^\infty(0) = 011010011001011010100101\dots$$

– no square in f :

$$\begin{cases} f(a) = abc, \\ f(b) = ac, \\ f(c) = b. \end{cases}$$

$$f = f^\infty(a) = abcacbabcbacabcacbacbcb\dots$$

How few squares in a word?

★ Proposition 1 ([Fraenkel, Simpson, 1995])

There is an infinite binary word containing only 3 squares, 2 cubes, and no other repetition of exponent ≥ 2 .

★ Morphism h_0 :

$$\begin{cases} h_0(\mathbf{a}) = 01001110001101, \\ h_0(\mathbf{b}) = 0011, \\ h_0(\mathbf{c}) = 000111. \end{cases}$$

$h_0 = h_0(f^\infty(\mathbf{a}))$ contains:

- the 3 squares 00, 11, 1010
- the 2 cubes 000 and 111
- no other repetition of exponent ≥ 2

[Badkobeh, C., 2010]

How few squares in a repetition-constrained word?

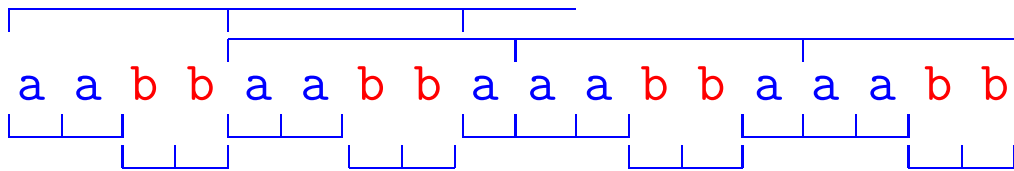
- ★ Theorem 3 ([Karhumäki, Shallit, 2004], [Shallit, 2008])
There is an infinite binary word avoiding $7/3^+$ -powers with a finitely many squares.
- ★ Theorem 4 ([Badkobeh, C., 2010])
... with 12 squares, the fewest possible.

$$\left\{ \begin{array}{l} g(a) = abac, \\ g(b) = babd, \\ g(c) = eabdf, \\ g(d) = fbace, \\ g(e) = bace, \\ g(f) = abdf. \end{array} \right. \quad \left\{ \begin{array}{l} h(a) = 10011, \\ h(b) = 01100, \\ h(c) = 01001, \\ h(d) = 10110, \\ h(e) = 0110, \\ h(f) = 1001. \end{array} \right.$$

- ★ $h = h(g^\infty(a))$ contains:
 - 12 squares, 2 $7/3$ -powers (0110110 and 1001001)
 - no other repetition of exponent ≥ 2

How many runs in a string?

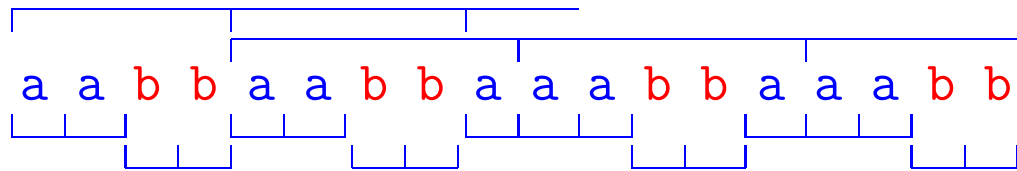
- ★ Useful for any algorithm dealing with repetitions in string
- ★ Word of length 18 with 10 runs



- ★ Theorem 5 ([Kolpakov, Kucherov, 1999])
There is no more than a linear number of runs in a string.

How many runs in a string?

- ★ Useful for any algorithm dealing with repetitions in string
- ★ Word of length 18 with 10 runs



- ★ **Theorem 6** ([Kolpakov, Kucherov, 1999])
There is no more than a linear number of runs in a string.
- ★ **Conjecture:** $\text{runs}(n) < n$
 $\text{runs}(n) =$ maximal number of runs in a string of length n
- ★ **Runs in binary strings:**

n	5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
$\text{runs}(n)$	2 3 4 5 5 6 7 8 8 10 10 11 12 13 14
n	20 21 22 23 24 25 26 27 28 29 30 31
$\text{runs}(n)$	15 15 16 17 18 19 20 21 22 23 24 25

Known bounds

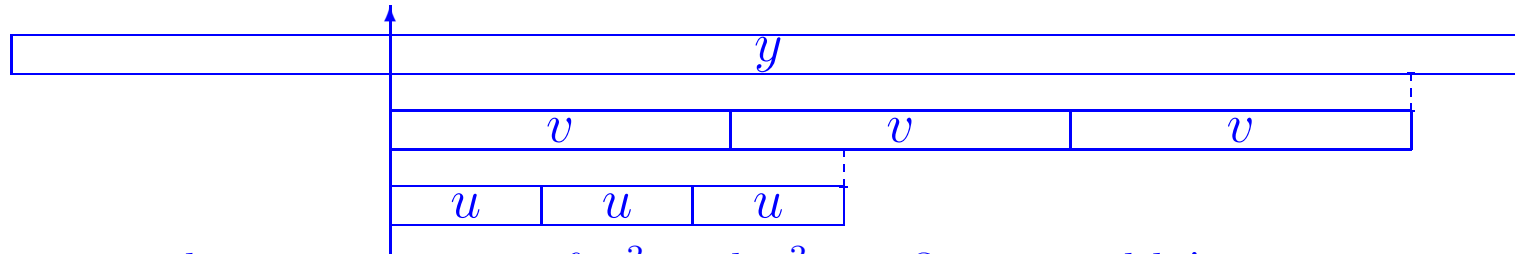
★ Upper bounds

- $5n$ [Rytter, 2006]
- $3.44n$ [Rytter, 2007][Puglisi, Simpson, Smyth, 2007]
- $1.6n$ [C., Ilie, 2007]
- $1.49n$ [Giraud, 2008]
- $1.029n$ [C., Ilie, Tinta, 2008]

★ Lower bounds

- $\frac{3}{1+\sqrt{5}}n \approx 0.927n$ [Franek, Simpson, Smyth, 2003]
- $0.94457564n$ [Kusano et al., 2008]
- $0.944575712n$ [Simpson, 2009]

How many cubes in a word?



largest position of u^3 and v^3 in y ? impossible!

Proposition 2

No more than $n - 2$ primitively-rooted cubes.

Valid for exponents $\geq 1 + \phi = \frac{3+\sqrt{5}}{2} \approx 2.62$

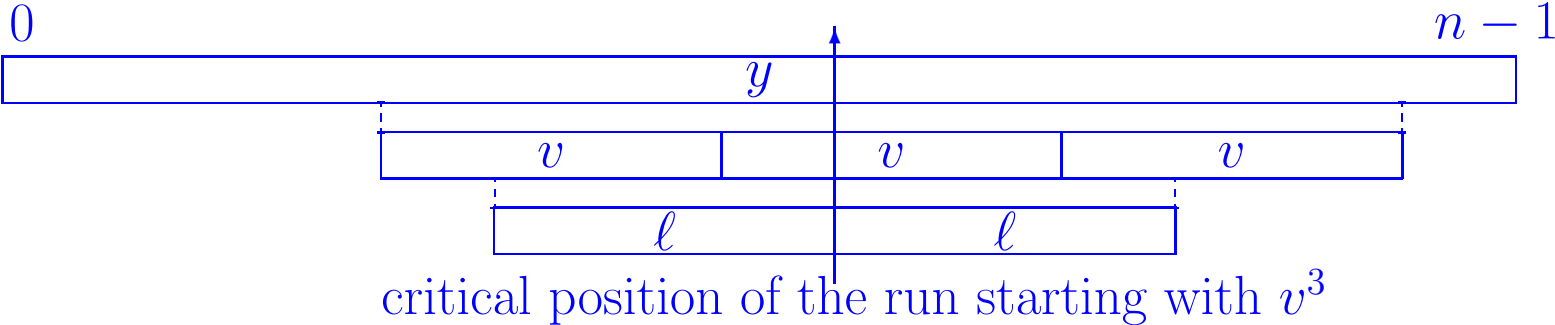
How many cubic runs in a string?

b a b a a a b a a a b a a a b b a a b b b b a

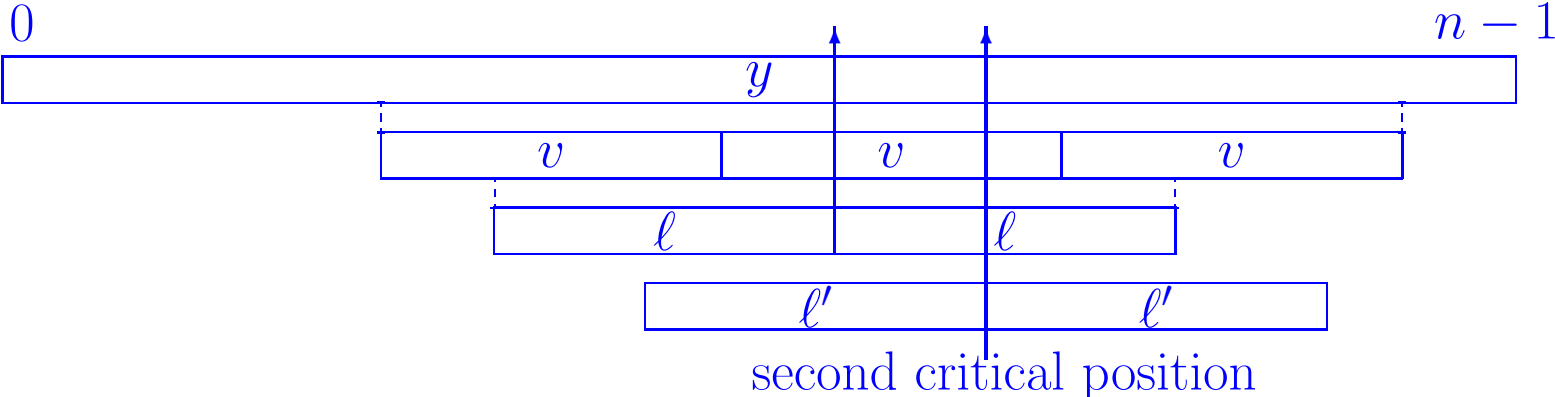
- ★ No more than $0.5n$ runs with exponent ≥ 3
- ★ There may be more than $0.406n$ runs with exponent ≥ 3
- ★ However the number of occurrences of primitively-rooted cubes can be $\Omega(n \log n)$
- ★ No obvious relation with the number of (distinct) cubes:

<p>b a a a c a a a d a a a e a a a f ..</p>	{	<p>1 cube $n/4$ runs</p>
<p>a b b a a b b a a b b a a b b a</p>	{	<p>$n/4$ cubes 1 run</p>

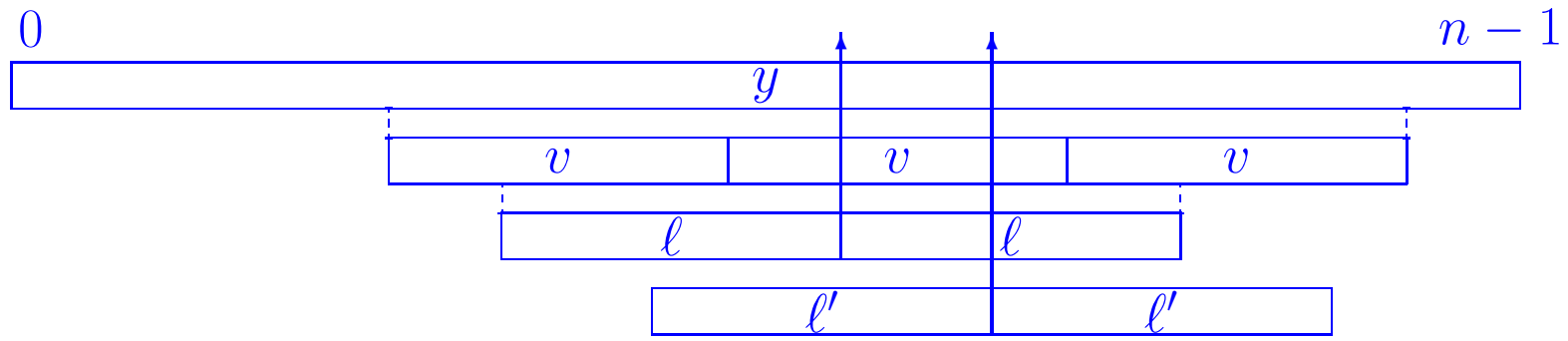
Upper bound



Upper bound



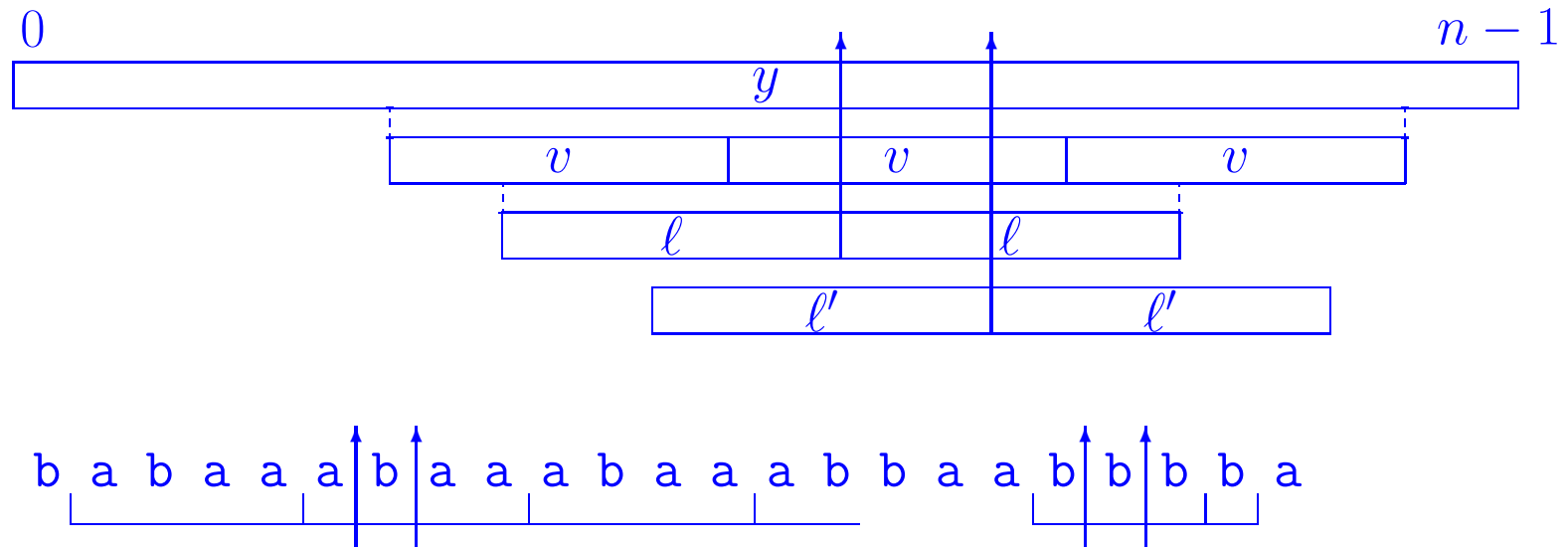
Upper bound



b a b a a a b a a a b a a b b a a b b b b a

Vertical arrows point upwards from the 6th, 7th, 14th, and 15th characters to the top of the main sequence bar in the diagram above.

Upper bound



- ★ the two (inter-)positions are associated with only one run
- ★ thus: no more than $(n - 1)/2$ runs with exponent ≥ 3

Stringology: the adventure goes on!

- ★ **Pattern matching**
approximate indexing, indeterminate strings, simple algorithms for repetitions, etc.
- ★ **Algorithms for bioinformatics**
alignments, matching for New Generation Sequencing, etc.
- ★ **Text compression**
compression/decompression speed, direct efficient factorisation, etc.
- ★ **Combinatorics**
solutions to conjectures: runs, squares, etc.

Collaborators on presented works

★ Abroad

- **Lucian Ilie**, University of Western Ontario
- **Marcin Kubica**, Warsaw University
- **Jakub Radoszewski**, Warsaw University
- **Wojciech Rytter**, Warsaw University
- **Liviu Tinta**, University of Western Ontario
- **Tomasz Waleń**, Warsaw University

★ KCL

- **Golnaz Badkobeh**
- **Supaporn Chairungsee**
- **Costas Iliopoulos**
- **Solon Pissis**
- **German Tischler**