Word and clump statistics

and an application to DNA evolution

P. Nicodème

CNRS - Team Calin - LIPN - University Paris13

00100

00100
111100

00100
111100
10111111100

00100
111100
10111111100
01100

00100
111100
10111111100
01100
0100

00100
111100
10111111100
01100
0100
11100

00100
111100
10111111100
01100
0100
11100
11100

00100
111100
10111111100
01100
0100
11100
11100
010100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010000111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111
111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111
111
00110001011

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111
111
001100010111
0100010010010100000100111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111
111
001100010111
0100010010010100000100111
00111

00100
111100
10111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000011010111
0111
111
001100010111
010001001001010000100111
00111
000101100101010000111

00100
111100
101111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000110101111
0111
111
001100010111
010001001001010000100111
00111
000101100101010000111
1010101010001010110010101000000100000110111

00100
111100
101111111100
01100
0100
11100
11100
010100
0110100
00100
1011111101011111100
00010100
0101100

100010100111
1100010100111
1000010111
0011000011000000111
0101010101010001111
1101101100110110101001011001100000110101111
0111
111
001100010111
010001001001010000100111
00111
000101100101010000111
1010101010001010110010101000000100000110111

1 1 1000011100

1	1	1000011100
2	2	0001101000010110

1	1	1000011100
2	2	0001101000010110
3	3	1001110101

1	1	1000011100
2	2	0001101000010110
3	3	1001110101
4		111011110

1	1	1000011100
2	2	0001101000010110
3	3	1001110101
4		1110111110
5		1110111111

1	1	1000011100
2	2	0001101000010110
3	3	1001110101
4		1110111110
5		1110111111
6	4	0010000101010

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010 10
7	5	11001011011

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010 10
7	5	11001011011
8	6	10011100010

1	1	10000011100
2	2	00011010000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010 10
7	5	11001011011
8	6	10011100010
9		10111010011

1	1	1000011100
2	2	0001101000010110
3	3	1001110101
4		1110111110
5		1110111111
6	4	0010000101010
7	5	1100101011011
8	6	1001100010
9		1011110100111
10	7	0011000001010

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000

1	1	1000011100
2	2	0001101000010110
3	3	1001110101
4		1110111110
5		1110111111
6	4	0010000101010
7	5	1100101011011
8	6	1001100010
9		1011110100011
10	7	0011000001010
11		0111111110010
12		1101011111100010
13		1110100011
14	8	1010010110000
15	9	0010000011011

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100000011011
16	10	00100000011110

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100000011011
16	10	00100000011110
17	11	00001100000111111

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100000011011
16	10	00100000011110
17	11	00001100000111111
18	12	10010100011

1	1	10000011100
2	2	00011010000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100000011011
16	10	00100000011110
17	11	00001100000111111
18	12	10010100011
19	13	00010101001110001

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100000011011
16	10	00100000011110
17	11	00001100000111111
18	12	10010100011
19	13	0001010100111000001
20		11111010011
21	14	10101001001000

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111
26		01101110000011

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111
26		01101110000011
27	17	10000000000

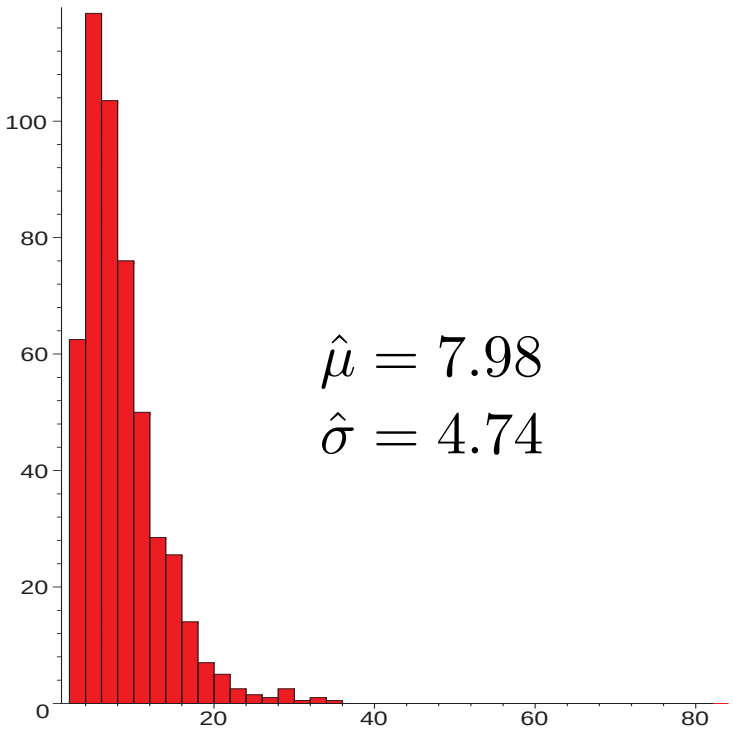
1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111
26		01101110000011
27	17	10000000000
28		00111111011

1	1	10000011100
2	2	00011010000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	00010101001110001
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	00010010101110
25		11111101111
26		01101110000011
27	17	10000000000
28		00111111011
29	18	11000011001

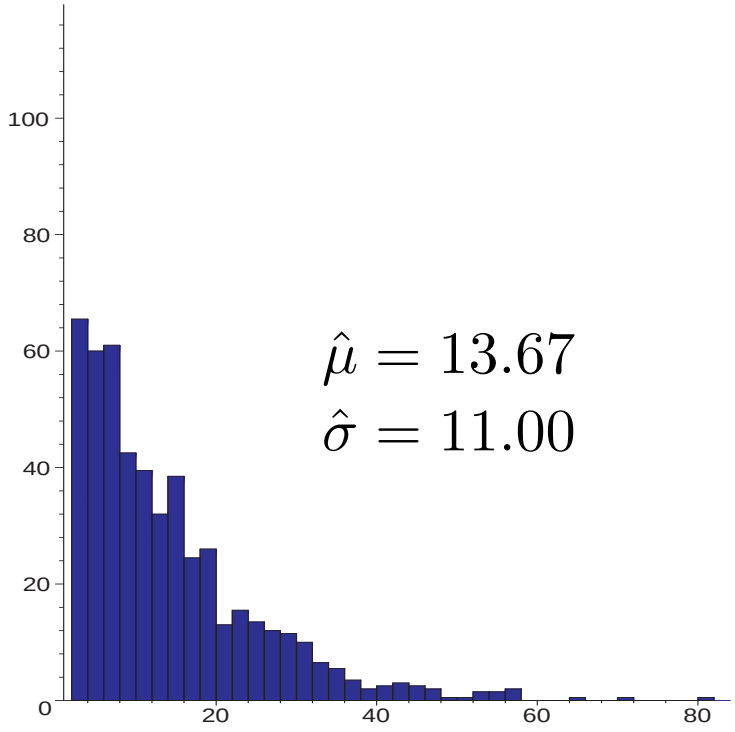
1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111
26		01101110000011
27	17	10000000000
28		00111111011
29	18	11000011001
30		01110010000

1	1	10000011100
2	2	0001101000010110
3	3	10011110101
4		11101111110
5		11101111111
6	4	00100001010
7	5	11001011011
8	6	10011100010
9		10111010011
10	7	00110000001010
11		01111110010
12		11010111100010
13		11101010011
14	8	10100101011000
15	9	00100001011
16	10	00100001110
17	11	00001100111111
18	12	10010100011
19	13	0001010100111000
20		11110101011
21	14	10101001001000
22	15	10100111111100
23		01110101111
24	16	0001001010110
25		11111101111
26		01101110000011
27	17	10000000000
28		00111111011
29	18	11000011001
30		01110010000

Waiting time (1000 random texts)

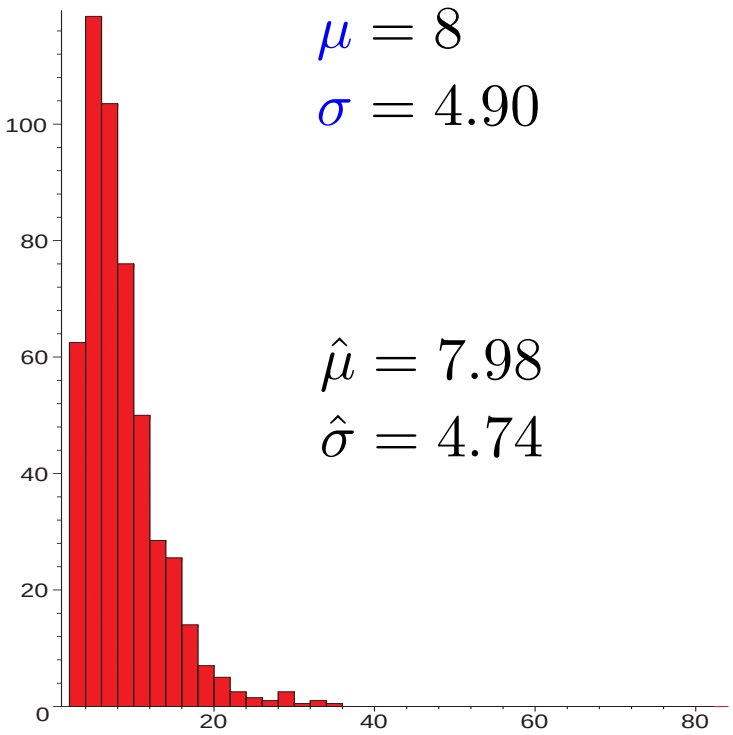


100



111

Waiting time (1000 random texts)



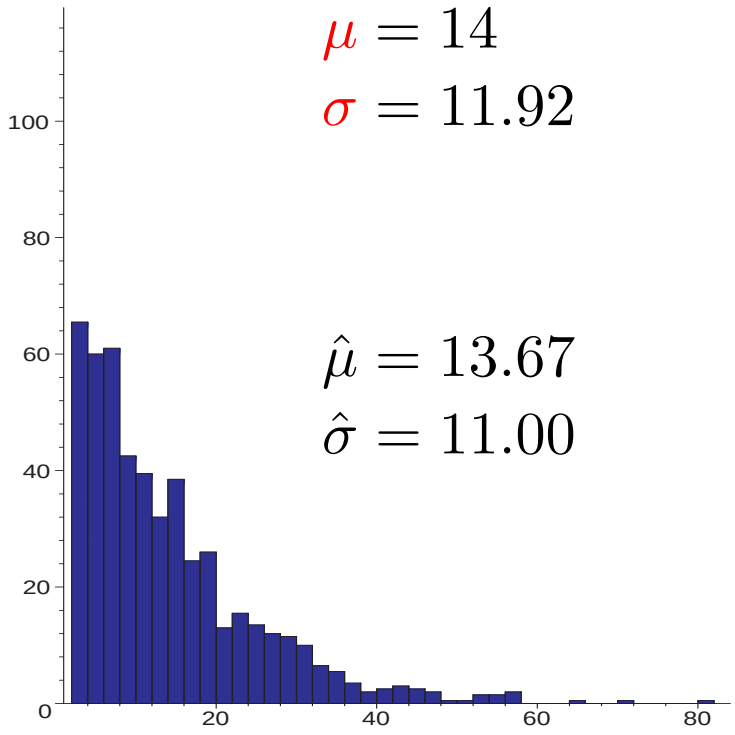
$\mu = 8$

$\sigma = 4.90$

$\hat{\mu} = 7.98$

$\hat{\sigma} = 4.74$

100



$\mu = 14$

$\sigma = 11.92$

$\hat{\mu} = 13.67$

$\hat{\sigma} = 11.00$

111

What is going on?

- Probability of appearance at a given position

$$\mathbf{P}(100) = \frac{1}{8}$$

$$\mathbf{P}(111) = \frac{1}{8}$$

What is going on?

- Probability of appearance at a given position

$$\mathbf{P}(100) = \frac{1}{8}$$

$$\mathbf{P}(111) = \frac{1}{8}$$

- BUT the 111 occur often by CLUMPS

$$\cdots 0111110 \qquad\qquad \cdots 011110$$
$$111 \qquad\qquad 111\cdots$$
$$111\cdots$$

- while the 100 NEVER OVERLAP

What is going on?

- Probability of appearance at a given position

$$\mathbf{P}(100) = \frac{1}{8}$$

$$\mathbf{P}(111) = \frac{1}{8}$$

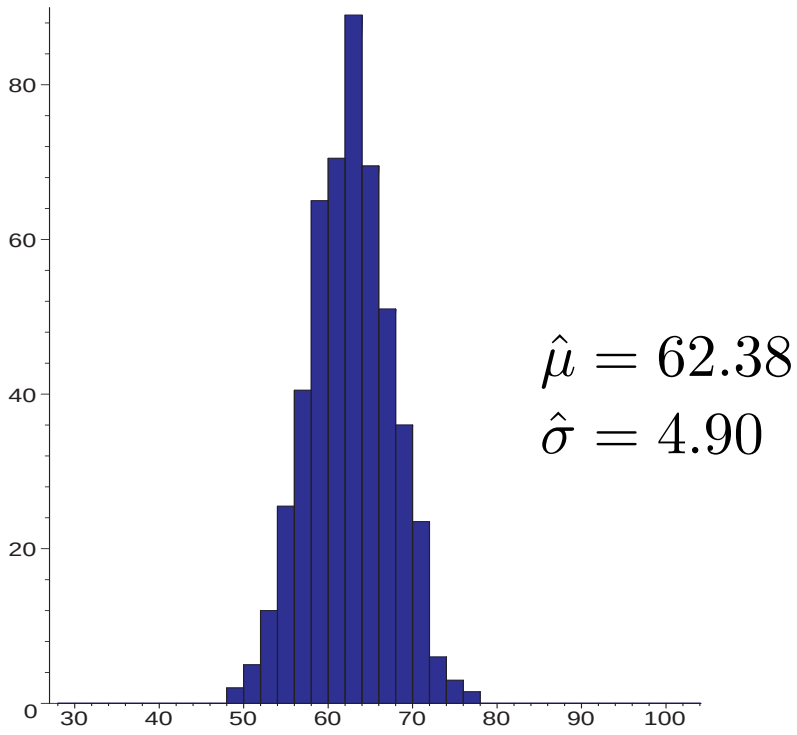
- BUT the 111 occur often by CLUMPS

$$\cdots 0111110 \qquad\qquad \cdots 011110$$
$$111 \qquad\qquad 111\cdots$$
$$111\cdots$$

- while the 100 NEVER OVERLAP

Expected waiting time: $\quad 111 - 14 \qquad 100 - 7$

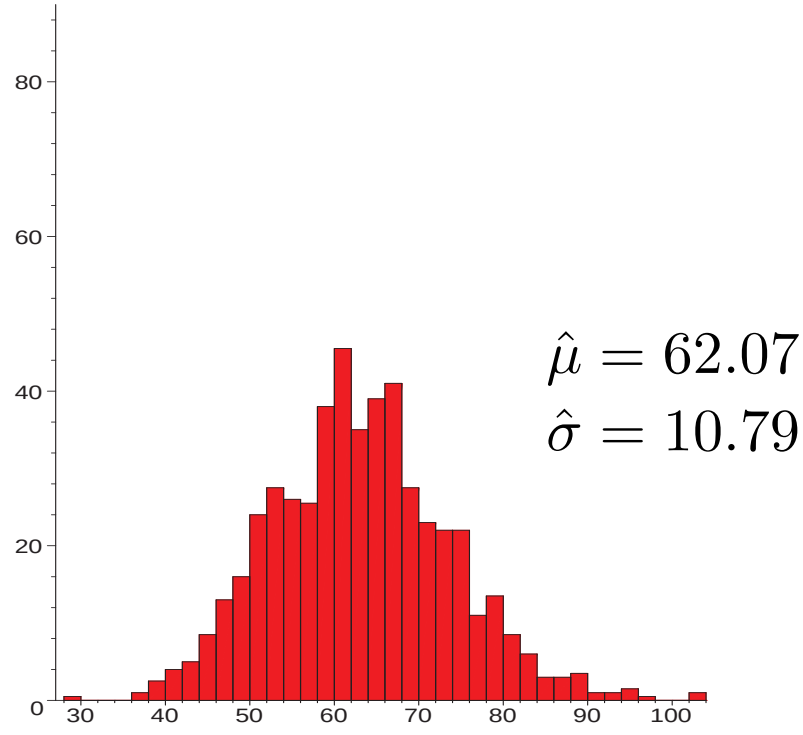
Number of occurrences 1000 texts of size 500



$\hat{\mu} = 62.38$

$\hat{\sigma} = 4.90$

100



$\hat{\mu} = 62.07$

$\hat{\sigma} = 10.79$

111

Part I

Statistics of reduced patterns

1 word - Bernoulli model

$\mathcal{A}$ alphabet, w considered word

Polynomial of autocorrelation

$\mathcal{C}_w = \{\ h, \quad w.h = u.w \quad \text{and} \quad |u| < |w|\ \}$

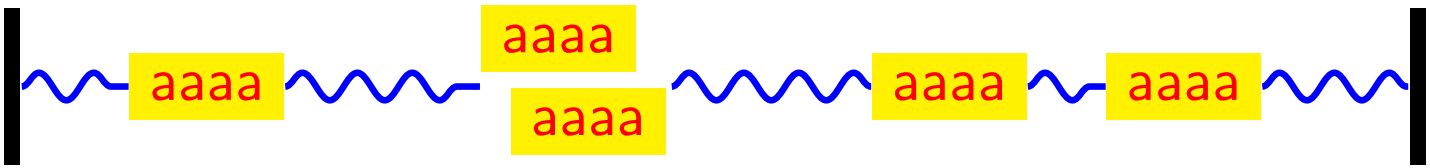
$w = ababa$

$$ababa$$
$$ababa|$$
$$\quad ababa$$
$$\quad\ ababa$$

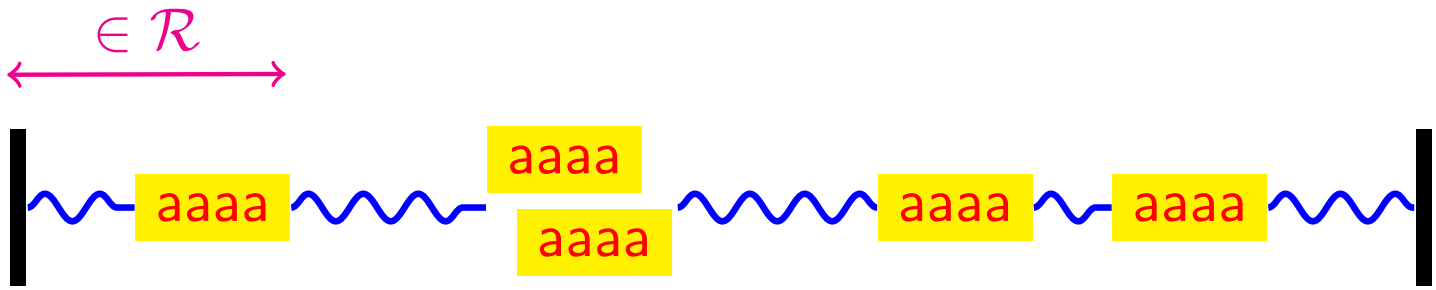
$\mathcal{C}_{ababa} = \{\epsilon, ba, baba\}$

$$C_{ababa}(z) = \sum_{v \in \mathcal{C}_{ababa}} \mathbf{P}(v) z^{|v|} = 1 + \omega_a \omega_b z^2 + \omega_a^2 \omega_b^2 z^4$$

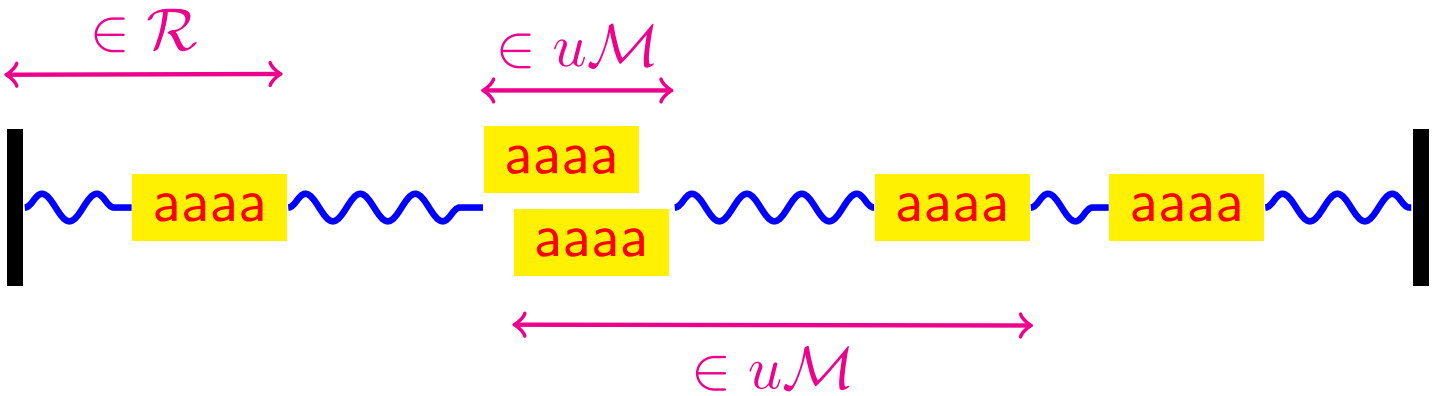
Guibas-Odlyzko decomposition for a word u



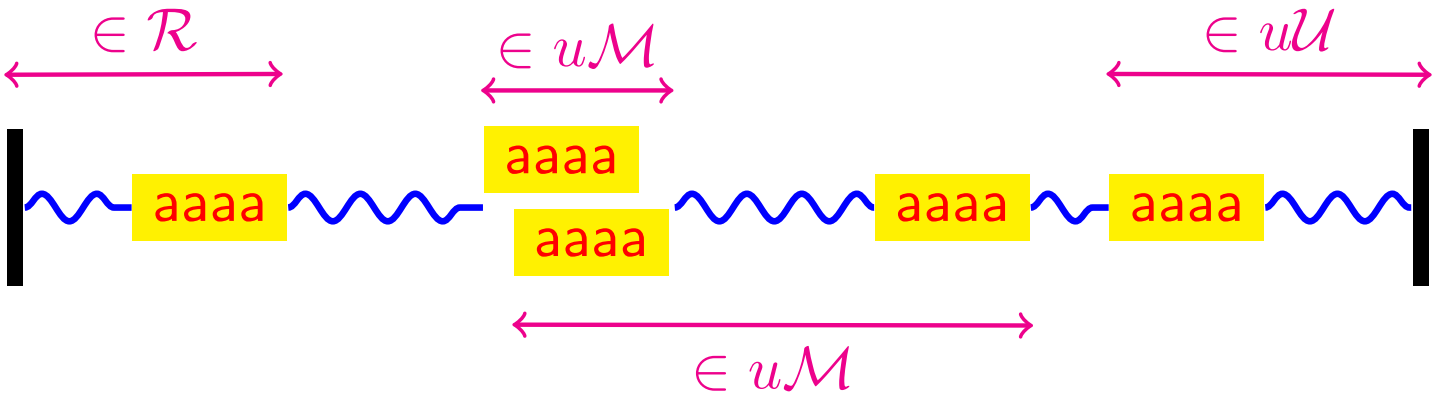
Guibas-Odlyzko decomposition for a word u



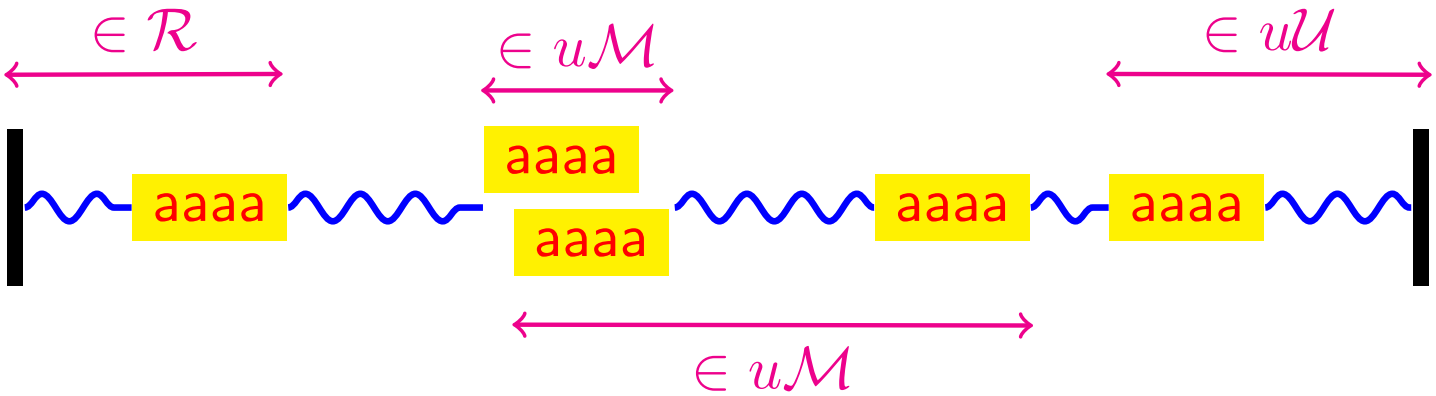
Guibas-Odlyzko decomposition for a word u



Guibas-Odlyzko decomposition for a word u



Guibas-Odlyzko decomposition for a word u



$$\mathcal{A}^\star = \mathcal{N} \cup \mathcal{R}.\mathcal{M}^\star.\mathcal{U} \quad \mathcal{L}_\bullet = \mathcal{N} \cup \mathcal{R}\bullet.(\mathcal{M}\bullet)^\star.\mathcal{U}$$

need to compute $\mathcal{N}, \mathcal{R}, \mathcal{M}, \mathcal{U}$

The languages $\mathcal{R}$, $\mathcal{M}$ and $\mathcal{U}$

First $\mathcal{R} = \{\ t = u.w \quad \text{and} \quad \nexists r, s, \quad t = r.w.s\ \}$

$aaaaaababa \in \mathcal{R}, \quad bbbbbbabababa \notin \mathcal{R}$

The languages $\mathcal{R}$, $\mathcal{M}$ and $\mathcal{U}$

First $\mathcal{R} = \{\ t = u.w \quad \text{and} \quad \nexists r, s, \quad t = r.w.s\ \}$

$aaaaaababa \in \mathcal{R}, \quad bbbbbababababa \notin \mathcal{R}$

Minimal $\mathcal{M} = \{\ t, \quad w.t = u.w \ \text{ and } \ \nexists r, s, \quad w.t = r.w.s\ \}$

$ababa_{aaaaababa} \in \mathcal{M} \quad ababa_{babbbbbbbbababa} \notin \mathcal{M} \quad ababa_{baba} \in \mathcal{M}$

The languages $\mathcal{R}$, $\mathcal{M}$ and $\mathcal{U}$

First $\mathcal{R} = \{\ t = u.w \quad \text{and} \quad \nexists r, s, \quad t = r.w.s\ \}$

$aaaaaababa \in \mathcal{R}, \quad bbbbbababababa \notin \mathcal{R}$

Minimal $\mathcal{M} = \{\ t, \quad w.t = u.w \ \text{ and } \ \nexists r, s, \quad w.t = r.w.s\ \}$

$ababa_{aaaaababa} \in \mathcal{M} \qquad ababa_{babbbbbbbbababa} \notin \mathcal{M} \qquad ababa_{ba} \in \mathcal{M}$

Ultimate $\mathcal{U} = \{t, \quad \nexists r, s, \quad w.t = r.w.s\}$

$ababa_{aabbbabbbbbbb} \in \mathcal{U} \qquad ababa_{babbbbbbbbbb} \notin \mathcal{U}$

The languages $\mathcal{R}$, $\mathcal{M}$ and $\mathcal{U}$

First $\mathcal{R} = \{\ t = u.w \quad \text{and} \quad \nexists r, s, \quad t = r.w.s\ \}$

$aaaaaababa \in \mathcal{R}, \quad bbbbbababababa \notin \mathcal{R}$

Minimal $\mathcal{M} = \{\ t, \quad w.t = u.w \ \text{ and } \ \nexists r, s, \quad w.t = r.w.s\ \}$

$ababa_{aaaaababa} \in \mathcal{M} \quad ababa_{babbbbbbbbababa} \notin \mathcal{M} \quad ababa_{ba} \in \mathcal{M}$

Ultimate $\mathcal{U} = \{t, \quad \nexists r, s, \quad w.t = r.w.s\}$

$ababa_{aabbbabbbbbbb} \in \mathcal{U} \qquad ababa_{babbbbbbbbbb} \notin \mathcal{U}$

Not $\mathcal{N} = \overline{\mathcal{A}^\star.w.\mathcal{A}^\star} = \{t, \ \nexists r, s, \quad t = r.w.s\}$

Régnier-Szpankowski - Equations over the langages

- (I) $\mathcal{A}^\star = \mathcal{U} + \mathcal{M}\mathcal{A}^\star \quad \Leftrightarrow \quad w.\mathcal{A}^\star = w.\mathcal{U} + w.\mathcal{M}\mathcal{A}^\star$
  – for any word beginning with w
  1. 1 single occurrence of $w \quad \Rightarrow \quad w.\mathcal{U}$
  2. several occurrences of $w \quad \Rightarrow \quad w.\mathcal{M}.\mathcal{A}^\star$

Régnier-Szpankowski - Equations over the langages

- (I) $\mathcal{A}^\star = \mathcal{U} + \mathcal{M}\mathcal{A}^\star \quad \Leftrightarrow \quad w.\mathcal{A}^\star = w.\mathcal{U} + w.\mathcal{M}\mathcal{A}^\star$
  – for any word beginning with w
  1. 1 single occurrence of $w \quad \Rightarrow \quad w.\mathcal{U}$
  2. several occurrences of $w \quad \Rightarrow \quad w.\mathcal{M}.\mathcal{A}^\star$

- (II) $\mathcal{A}^\star w = \mathcal{R}.\mathcal{C} + \mathcal{R}.\mathcal{A}^\star.w$ (remark $\epsilon \subset \mathcal{C}$)
  – for any word finishing by w
  1. first occurrence w overlaps last one
     or single occurrence of $w \quad \Rightarrow \mathcal{R}.\mathcal{C}$
  2. first occurrence of w does not overlap last $\quad \Rightarrow \mathcal{R}.\mathcal{A}^\star.w$

Equations over the languages (continued)

- (III) $\begin{cases} & \mathcal{M}^+ = \quad \mathcal{A}^\star.w + \quad \mathcal{C} - \epsilon \\ \Leftrightarrow & w.\mathcal{M}^+ = w.\mathcal{A}^\star.w + w.(\mathcal{C} - \epsilon) \end{cases}$
  1. $w.\mathcal{M}^+$, $\Rightarrow$ words beginning and finishing with w
  2. $w.\mathcal{A}^\star.w$, $\Rightarrow$ first and last occurrences do not overlap
  3. $w.(\mathcal{C} - \epsilon)$ $\Rightarrow$ first and last occurrences overlap

Equations over the languages (continued)

- (III) $\begin{cases} & \mathcal{M}^+ = \quad \mathcal{A}^\star.w + \quad \mathcal{C} - \epsilon \\ \Leftrightarrow & w.\mathcal{M}^+ = w.\mathcal{A}^\star.w + w.(\mathcal{C} - \epsilon) \end{cases}$
  1. $w.\mathcal{M}^+$, $\Rightarrow$ words beginning and finishing with w
  2. $w.\mathcal{A}^\star.w$, $\Rightarrow$ first and last occurrences do not overlap
  3. $w.(\mathcal{C} - \epsilon)$ $\Rightarrow$ first and last occurrences overlap

- (IV) $\mathcal{N}.\mathcal{A} = \mathcal{R} + \mathcal{N} - \epsilon$

  – concatenate a letter to a word of N $\Rightarrow$
  1. creates a match $\Rightarrow \mathcal{R}$
  2. does not create a match $\Rightarrow \mathcal{N}$
  3. (empty word ϵ forbidden)

From Languages to Generating Functions

(I) $\mathcal{A}^\star = \mathcal{U} + \mathcal{M}\mathcal{A}^\star \quad \rightsquigarrow \quad \frac{1}{1-z} = U(z) + \frac{M(z)}{1-z}$

(II) $\mathcal{A}^\star w = \mathcal{R}.\mathcal{A} + \mathcal{R}.\mathcal{A}^\star.w \quad \rightsquigarrow \quad \frac{\omega_w z^{|w|}}{1-z} = F(z)\left(C(z) + \frac{\omega_w z^{|w|}}{1-z}\right)$

(III) $\mathcal{M}^+ = \mathcal{A}^\star.w + \mathcal{A} - \epsilon \quad \rightsquigarrow \quad \frac{M(z)}{1-M(z)} = \frac{\omega_w z^{|w|}}{1-z} + C(z) - 1$

(IV) $\mathcal{N}.\mathcal{A} = \mathcal{R} + \mathcal{N} - \epsilon \quad \rightsquigarrow \quad zN(z) = F(z) + N(z) - 1$

Solving the system

$$R(z) = \frac{\omega_w z^{|w|}}{\omega_w z^{|w|} + (1-z)C(z)} \qquad U(z) = \frac{1}{\omega_w z^{|w|} + (1-z)C(z)}$$

$$N(z) = \frac{C(z)}{\omega_w z^{|w|} + (1-z)C(z)} \qquad M(z) = 1 + \frac{z-1}{\omega_w z^{|w|} + (1-z)C(z)}$$

$$\mathcal{L}_\bullet = \mathcal{N} \cup \mathcal{R}\bullet.(\mathcal{M}\bullet)^\star.\mathcal{U}$$

$$L(z,u) = N(z) + \frac{R(z)uU(z)}{1-uM(z)} = \frac{1}{1 - z - \dfrac{(u-1)\omega_w z^{|w|}}{1-(u-1)(C(z)-1)}}$$

Moments

X_n **random variable counting the number of occurrences of the word** w **in a random text of length** $\mathbf{n}$

- $$\mathbf{E}(X_n) = [z^n] \left. \frac{\partial L(z,u)}{\partial u} \right|_{u=1} = (\mathbf{n} - |w| + 1)\omega_w$$

- $$\begin{aligned} \mathbf{Var}(X_n) &= [z^n] \left. \frac{\partial}{\partial u} u \frac{\partial L(z,u)}{\partial u} \right|_{u=1} - \mathbf{E}^2(X_n) \\ &= \mathbf{n} \times \omega_w \big(2\mathcal{C}(1) - 1 - (2|w| - 1)\omega_w\big) + O(1) \end{aligned}$$

Number of occurrences

	100	111
$L(z,u)$	$\dfrac{1}{1-z+(1-u)\dfrac{z^3}{8}}$	$\dfrac{1+\dfrac{1-u}{8}(4z+2z^2)}{1-z-\dfrac{1-u}{8}(4z-2z^2-z^3)}$
$\mu(z)$	$\dfrac{z^3}{8(1-z)^2}$	$\dfrac{z^3}{8(1-z)^2}$
$m_{(2)}(z)$	$\dfrac{z^3}{8(1-z)^2}-\dfrac{z^3}{32(1-z)^3}$	$\dfrac{8z^3-4z^5-2z^6}{32(1-z)^3}$
μ_n	$(n-2)/8$	$(n-2)/8$
σ_n	$\sqrt{3n}/8$	$\sqrt{15n-40}/8$
μ_{500}	$249/4 = 62.25$	$249/4 = 62.25$
σ_{500}	$\sqrt{1500}/8 \approx 4.84$	$\sqrt{7460}/8 \approx 10.80$

Reduced compound patterns

$W = \{w_1, w_2\}$ and w_1 (resp. w_2) is **not factor** of w_2 (resp. w_1)

Correlation of words

$$\mathbb{C}(z) = (\mathcal{C}_{i,j}(z)) \qquad \mathcal{C}_{i,j} = \{h,\ w_i.h = u.w_j\}$$

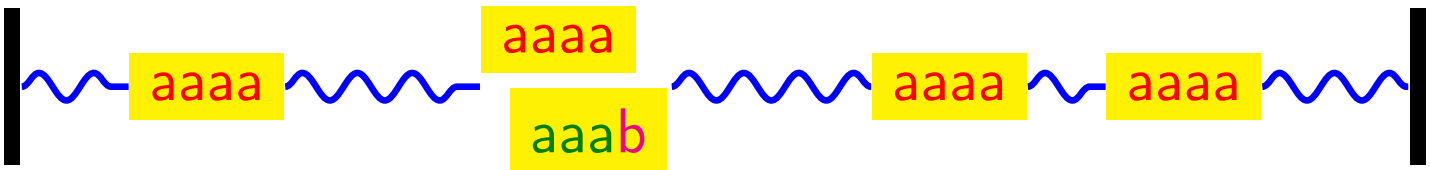
Example $W = \{aab, abaa\}$

$$\mathbb{C}(z) = \begin{pmatrix} 1 & \omega_a^2 z^2 \\ \omega_b z & 1 + \omega_a^2 \omega_b z^3 \end{pmatrix}$$

Languages **Right**, **Minimal**, **Ultimate** $\mathcal{R}_i, \mathcal{M}_{i,j}, \mathcal{U}_i$

Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

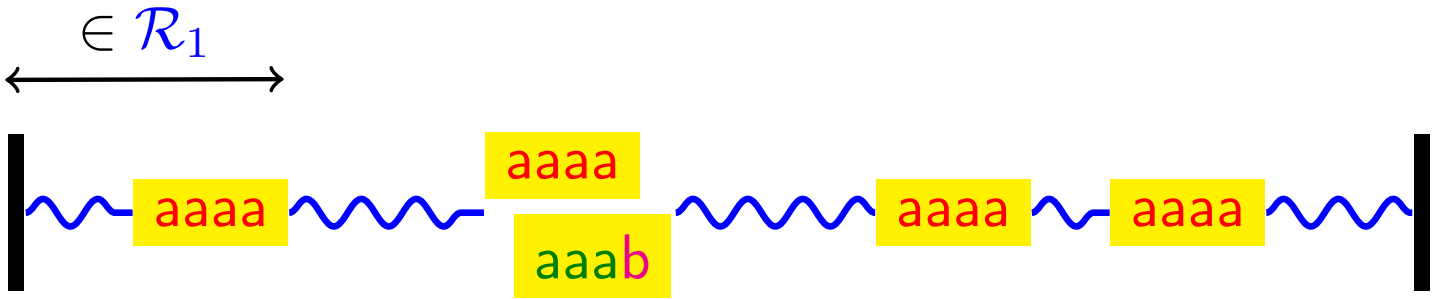
$$U = (\texttt{aaaa}, \texttt{aaab}) \qquad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$$



Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

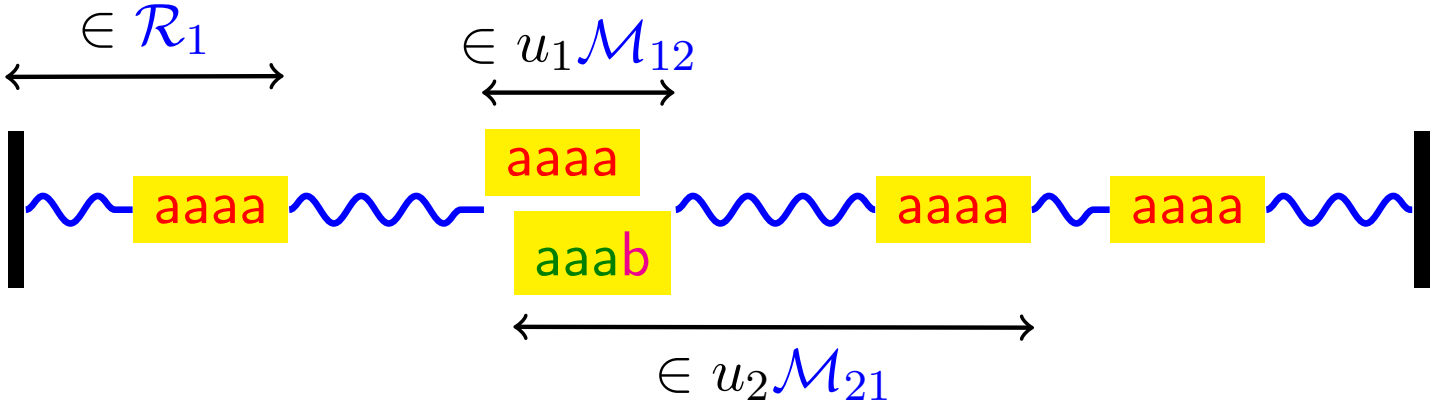
$U = (\texttt{aaaa}, \texttt{aaab})$

$$\begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$$



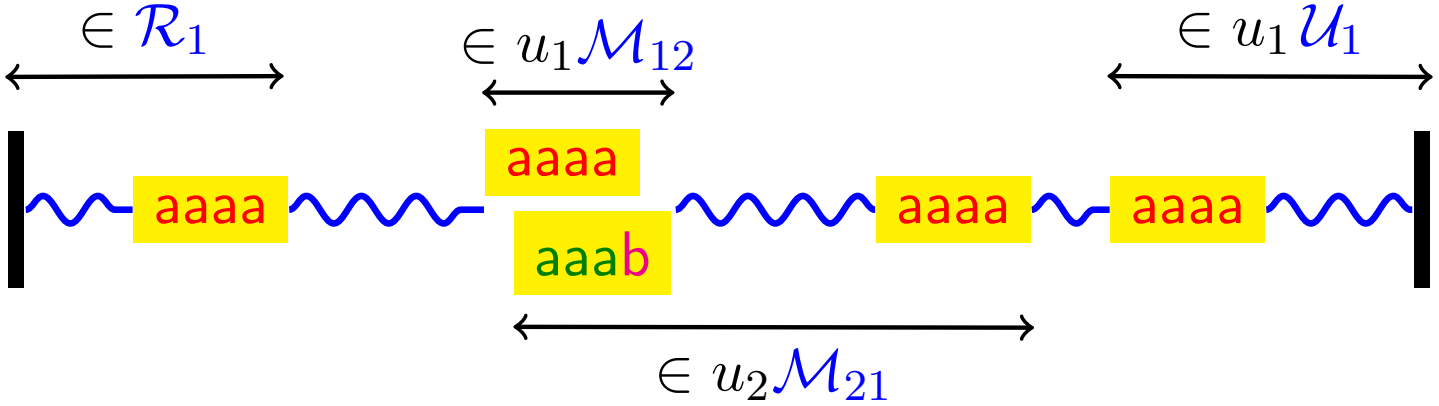
Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

$U = (\texttt{aaaa}, \texttt{aaab})$ $\quad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$



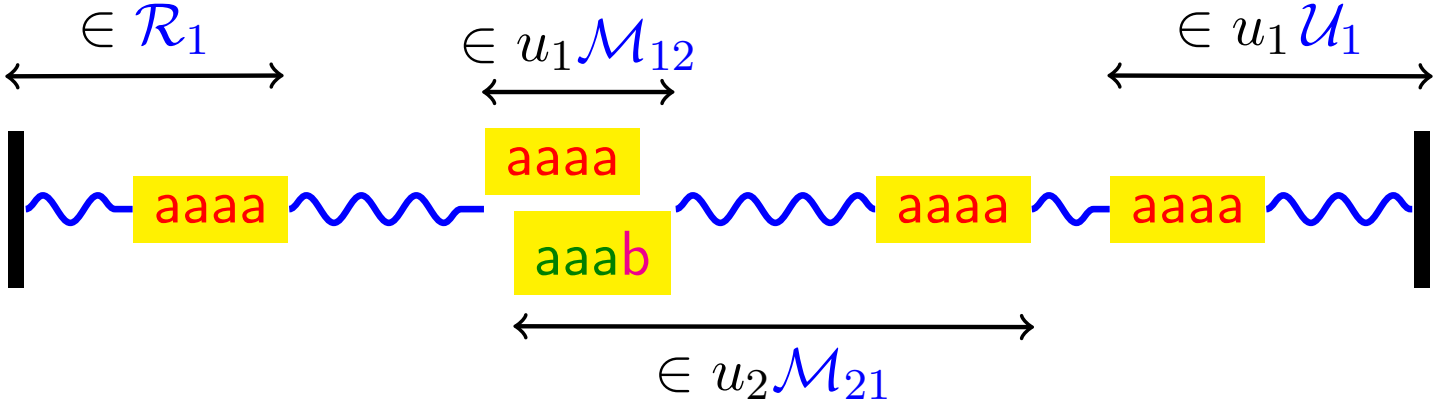
Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

$U = (\texttt{aaaa}, \texttt{aaab})$ $\quad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$



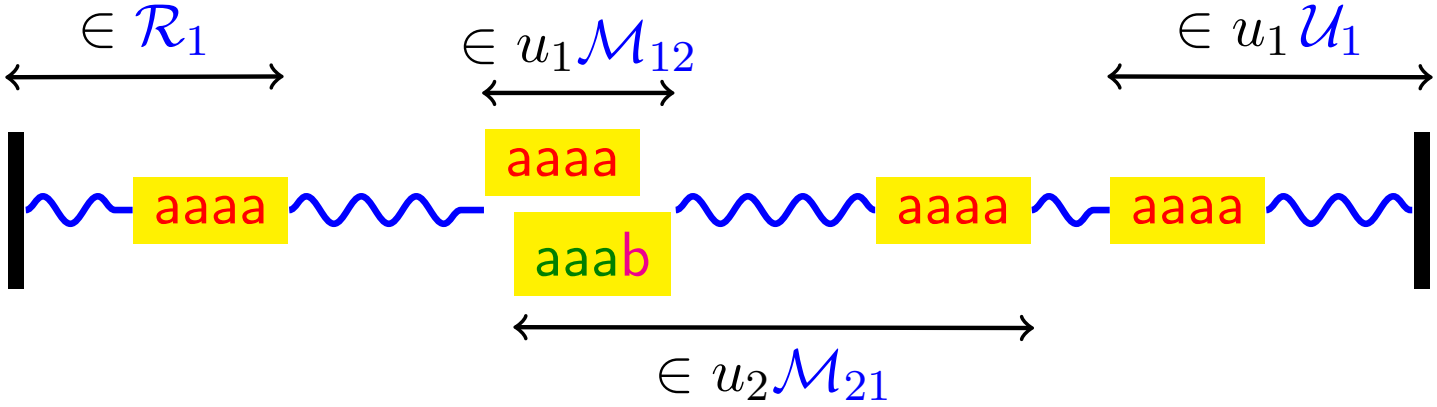
Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

$U = (\texttt{aaaa}, \texttt{aaab})$ $\quad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$



Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

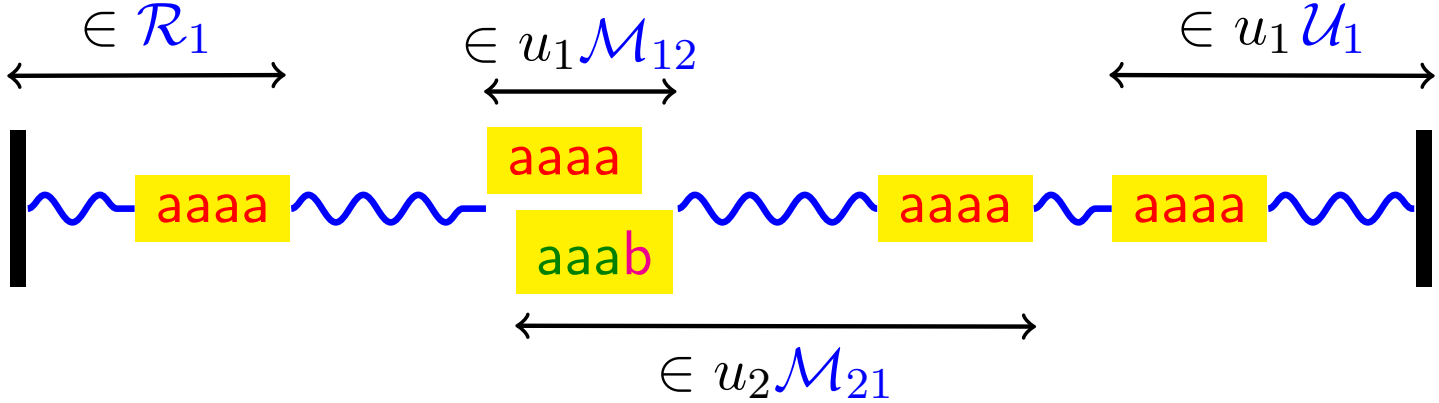
$U = (\texttt{aaaa}, \texttt{aaab})$ $\quad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$



- The "Right" language $\mathcal{R}_i$ associated to the word u_i is the set of words $\mathcal{R}_i = \{r \,|\, r = e \cdot u_i$ and there is no $v \in U$ such that $r = xvy$ with $|y| > 0\}$.
- The "Minimal" language $\mathcal{M}_{ij}$ leading from a word u_i to a word u_j is the set of words $\mathcal{M}_{ij} = \{m \,|\, u_i \cdot m = e \cdot u_j$ and there is no $v \in U$ such that $u_i \cdot m = xvy$ with $|x| > 0, |y| > 0\}$.
- The "Ultimate" language $\mathcal{U}_i$ of words following the last occurrence of the word u_i (such that this occurrence is the last occurrence of U in the text) is the set of words $\mathcal{U}_i = \{u \,|\,$ there is no $v \in U$ such that $u_i \cdot u = xvy$ with $|x| > 0\}$.
- The "Not" language $\mathcal{N}$ is the set of words with no occurrences of U, $\mathcal{N} = \{n \mid$ there is no $v \in U$ such that $n = xvy\}$.

Guibas-Odlyzko decomposition for a pattern (u_1, u_2)

$$U = (\texttt{aaaa}, \texttt{aaab}) \qquad \begin{cases} u_1 = \texttt{aaaa} \\ u_2 = \texttt{aaab} \end{cases}$$



$$F(z, x_1, x_2)$$
$$= \mathcal{N}(z) + (\mathcal{R}_1(z)x_1, \mathcal{R}_2(z)x_2) \begin{pmatrix} \mathcal{M}_{11}(z)x_1 & \mathcal{M}_{12}(z)x_2 \\ \mathcal{M}_{21}(z)x_1 & \mathcal{M}_{22}(z)x_2 \end{pmatrix} \begin{pmatrix} \mathcal{U}_1(z) \\ \mathcal{U}_2(z) \end{pmatrix}$$

Computing the languages

- **Régnier-Szpankowski Equations**

$$\bigcup_{k\geq 1}\left(\mathbb{M}^k\right)_{i,j} = \mathcal{A}^\star\cdot u_j + \mathcal{C}_{ij} - \delta_{ij}\epsilon, \quad \mathcal{U}_i\cdot\mathcal{A} = \bigcup_j \mathcal{M}_{ij} + \mathcal{U}_i - \epsilon,$$
$$\mathcal{A}\cdot\mathcal{R}_j - (\mathcal{R}_j - u_j) = \bigcup_i u_i\mathcal{M}_{ij}, \qquad \mathcal{N}\cdot u_j = \mathcal{R}_j + \bigcup_i \mathcal{R}_i\left(\mathcal{C}_{ij} - \delta_{ij}\epsilon\right),$$

$$L(z, x_1, x_2)$$
$$= N(z) + (R_1(z)x_1, R_2(z)x_2)\begin{pmatrix} M_{11}(z)x_1 & M_{12}(z)x_2 \\ M_{21}(z)x_1 & M_{22}(z)x_2 \end{pmatrix}\begin{pmatrix} U_1(z) \\ U_2(z)\end{pmatrix}$$

- **The generating functions are also computable by automata**

Markov case - One word w

- **Conditional probability matrix:**
  $\mathbb{P} = (p_{ij}), \quad i, j \in \{1, \ldots, |\mathcal{A}|\}$
- $w = aababa$
- $\mathcal{C} = \{baa, abaa\}$
- $C_w(z) = p_{ab}p_{ba}p_{aa}^2 z^3 + p_{aa}p_{ab}p_{aa}^2 z^4$
- $C_w(z)$ is the **conditionned** generating function of the autocorrelation set

Markov case - One word $w = w_1.w_2 \dots .w_m$

Conditional probability matrix: $\mathbb{P} = (p_{ij}), \quad i,j \in \{1,\dots,|\mathcal{A}|\}$

Stationnary vector: $\mu = (\mu_1,\dots,\mu_{|\mathcal{A}|})$

Stationnary matrix: $\Pi = \begin{pmatrix} \mu \\ \vdots \\ \mu \end{pmatrix} = \lim_{n\to\infty} \mathbb{P}^n$

$$R(z) = \frac{P(w)z^m}{D} \qquad M(z) = 1 + \frac{z-1}{D}$$

$$U(z) = \frac{1}{D} \qquad N(z) = \frac{D - P(w)z^m}{(1-z)D}$$

$$D = (1-z)C_w(z) + P(w)z^m\big(1-(1-z)F(z)\big)$$

$$F(z) = \frac{1}{\mu_{w_1}}\Big[(\mathbb{P}-\Pi)\Big(\mathbb{I}-(\mathbb{P}-\Pi)z\Big)^{-1}\Big]_{w_m,w_1}$$

Limit Laws

- **Poisson** when number of occurrences is $O(1)$
- **Gaussian** when number of occurrences is $\Theta(n)$

Also **large deviation result**

Bibliography

- **Lothaire** book (2005), Chapter 7
- **Régnier** (2000) "A unified approach to word occurrences probabilities"

Part II

Clump Analysis

Counting clumps

$w = aa \qquad T = bbbbaaaabbbbaaa$

- word occurrences counting with overlaps: 5 matches $(aa|a|a|, aa|a|)$
- clumps counting: 2 matches $(aaaa|, aaa|)$

Counting clumps

$w = aa \qquad T = bbbbaaaabbbbaaa$

- word occurrences counting with overlaps: 5 matches $(aa|a|a|, aa|a|)$
- clumps counting: 2 matches $(aaaa|, aaa|)$

A clump of occurrences of a word w is

- either composed of a **single occurrence** of w
- or of a **maximal set** of occurrences of w such that **each occurrence** of w **overlaps at least another occurrence** of w

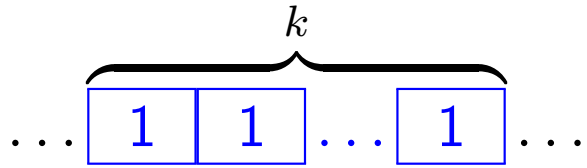
Generalization to clumps of occurrences of a set of words $\{u_1, \ldots, u_r\}$

Aim of our work

- a **combinatorial** analysis for counting clumps of reduced sets of words
- an **algorithmic** contruction by **automata** that solves the counting problem in the general case and implies a normal limit law

Probability of start at a position i

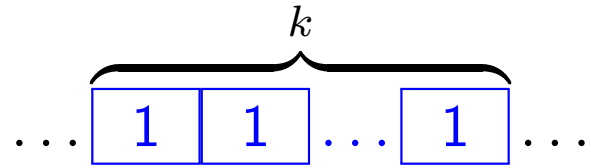
(A) word $w = 1^k$ $\qquad\qquad$ $p = \mathbf{P}(1) = 1 - \mathbf{P}(0) = 1 - q$



$\mathbf{P}(\text{start}) = p^k$

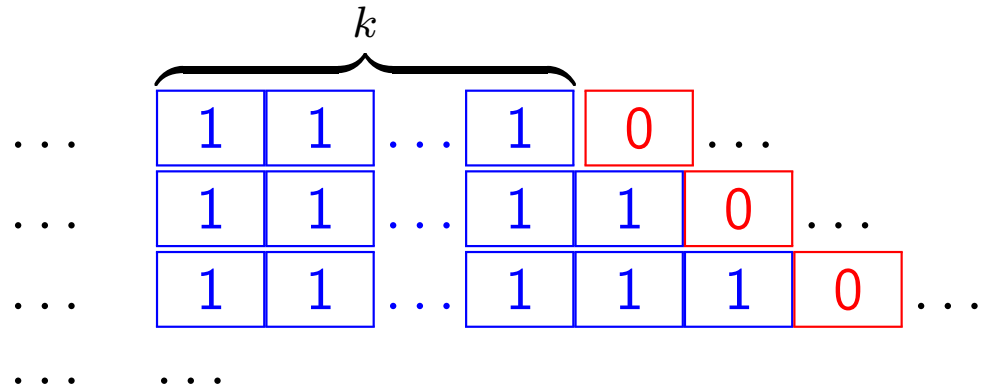
Probability of start at a position i

(A) word $w = 1^k$ $\qquad p = \mathbf{P}(1) = 1 - \mathbf{P}(0) = 1 - q$



$\mathbf{P}(\text{start}) = p^k$

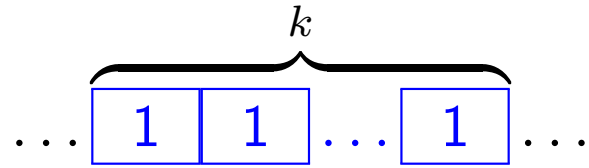
(B) clump $\Gamma = 1^k.1^\star$



$$\mathbf{P}(\text{start}) = p^k \times q \times (1 + p + p^2 + \dots) = p^k \times \frac{q}{1-p} = p^k$$

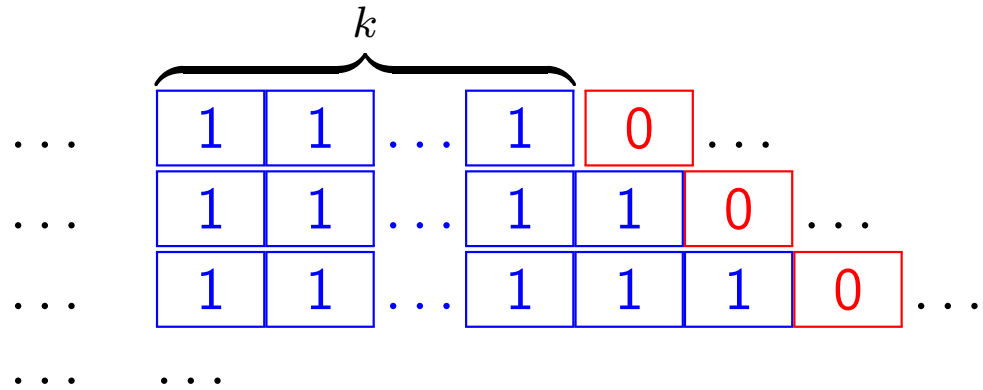
Probability of start at a position i

(A) word $w = 1^k$ $\qquad p = \mathbf{P}(1) = 1 - \mathbf{P}(0) = 1 - q$



$\mathbf{P}(\text{start}) = p^k$

(B) clump $\Gamma = 1^k.1^\star$

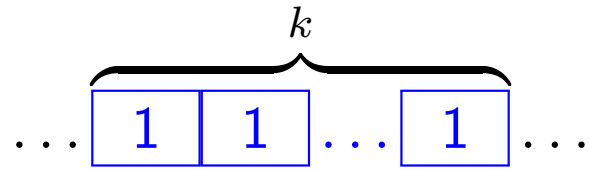


$$\mathbf{P}(\text{start}) = p^k \times q \times (1 + p + p^2 + \dots) = p^k \times \frac{q}{1-p} = p^k$$

FALSE

Probability of start at a position i

(A) word $w = 1^k$



$\mathbf{P}(\text{start}) = p^k$

(B) clump $\Gamma = 1^k.1^\star$

$\ldots 1|\overbrace{1 \ldots 1}^{k}\ldots$ no clump **beginning** at position i

$\ldots 0|\overbrace{1 \ldots 1}^{k}\ldots$ a clump **begins** at position i

$\mathbf{P}(\text{start}) = p^k \times q = p^k \times (1-p)$

Probabilistic approach

Prum, Reinert, Schbath, Pape, ...

$w = aaa$

$\mathbf{P}(a)$ small $\rightsquigarrow$ $\mathbf{P}$(start-of-a-clump-at-a-position) small

$\rightsquigarrow$

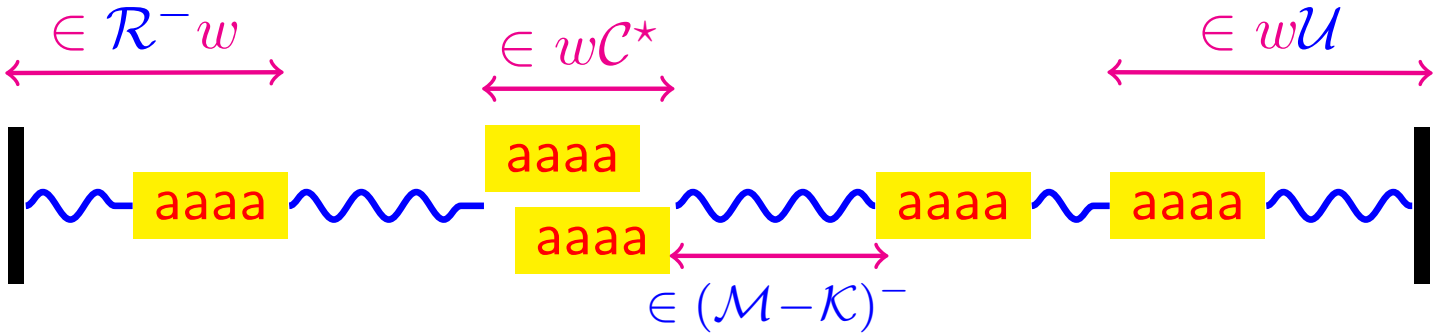
Poisson law for the number of clumps

Geometric law for the number of words in a clump

This extends to the general case.
Chen-Stein Poisson approximation provides bounds for the total variation distance to the composed law.

A - Combinatorial approach

Language equations for the clumps



Notation: if $\mathcal{L} = \mathcal{W} \cdot w$ we write $\mathcal{L}^- = \mathcal{W}$

Combinatorial decomposition

$$\begin{aligned}\mathcal{A}^\star &= \mathcal{N} + \mathcal{R}^- \mathbf{\Gamma} \Big((\mathcal{M} - \mathcal{K})^- \mathbf{\Gamma} \Big)^\star \mathcal{U} \\ &= \mathcal{N} + \mathcal{R}^- w\, \mathcal{C}^\star \Big((\mathcal{M} - \mathcal{K})^- w\, \mathcal{C}^\star \Big)^\star \mathcal{U} \\ &= \mathcal{N} + \mathcal{R}^- w\, \mathcal{K}^\star \Big((\mathcal{M} - \mathcal{K}) - w\, \mathcal{K}^\star \Big)^\star \mathcal{U}\end{aligned}$$

Clump: $\mathbf{\Gamma} = w\, \mathcal{C}^\star = w\, \mathcal{K}^\star$

Some combinatorial properties

$w = aaaaa$
$\mathcal{C} - \{\epsilon\} = \{a, aa, aaa, aaaa\}$
$\mathcal{K} = \{a\}$
$\mathcal{M} = \{a, b(b + ab + aab + aaab + aaaab)^\star aaaaa\}$

Properties

- $\mathcal{K} \subset \mathcal{M}$
- $\mathcal{M} - \mathcal{K} = \mathcal{L}w$
- $\mathcal{K}^\star = \mathcal{C}^\star$ and $\mathcal{K}^\star$ is **unambiguous**

A Prefix Code generating the clumps

Lemma.
Let $\mathcal{C}_\circ = \mathcal{C} - \{\epsilon\}$ be the strict autocorrelation set of a word w

- the Prefix code $\mathcal{K} = \mathcal{C}_\circ - \mathcal{C}_\circ\mathcal{A}^+$ generates **unambiguously** $\mathcal{C}^+ - \{\epsilon\}$, which implies that $\mathcal{K}^\star = {\mathcal{C}_\circ}^\star$
- $\mathcal{K}^\star$ is **unambiguous**

Generating functions

$$F(z,\bullet,\bullet,\bullet) = \mathcal{N}(z) + \frac{\mathcal{R}(z)}{\omega_w z^{|w|}} \mathbf{\Gamma}(\bullet z,\bullet,\bullet) \frac{1}{1 - \frac{\mathcal{M}(z) - \mathcal{K}(z)}{\omega_w z^{|w|}} \times \mathbf{\Gamma}(\bullet z,\bullet,\bullet)} \mathcal{U}(z)$$

- number of w and number of clumps:

  $$\mathbf{\Gamma}(z,u,x) = ux\omega_w z^{|w|} \frac{1}{1 - x\mathcal{K}(z)}$$

- number of clumps and total number of positions inside clumps:

  $$\mathbf{\Gamma}(tz,u) = u\omega_w (tz)^{|w|} \frac{1}{1 - \mathcal{K}(tz)}$$

- number of w and total number of positions inside clumps:

  $$\mathbf{\Gamma}(tz,x) = x\omega_w (tz)^{|w|} \frac{1}{1 - x\mathcal{K}(tz)}$$

- number of "stuttering" w and total number of positions inside clumps:

  $$\mathbf{\Gamma}(tz,x) = \frac{x}{1-x} \omega_w (tz)^{|w|} \frac{1}{1 - \frac{x}{1-x}\mathcal{K}(tz)}$$

One word - Expectation - Variance

clumps

$$\mathbf{E}(O_n^{\mathfrak{K}}) = (\mathbf{n} - |w| + 1)\omega_w(1 - \mathcal{K}(1)) - \omega_w\mathcal{K}'(1)$$

$$\mathbf{Var}(O_n^{\mathfrak{K}}) = \mathbf{n} \times (1 - \mathcal{K}(1))^2\mathbf{V}_w - \mathbf{n} \times \omega_w(1 - \mathcal{K}(1))(\mathcal{K}(1) - 2\omega_w\mathcal{K}'(1))$$

one word

$$\mathbf{E}(O_n^w) = (\mathbf{n} - |w| + 1)\omega_w, \qquad \mathbf{Var}(O_n^w) = \mathbf{n} \times \mathbf{V}_w + O(1).$$

$$\mathbf{V}_w = \omega_w\big(2\mathcal{C}(1) - 1 - (2|w| - 1)\omega_w\big)$$

Putting up equations for clumps of two words

Minimal Correlation Language: $\mathcal{K}_{ij} = \mathcal{C}_{ij} - \mathcal{C}_{ij}\mathcal{A}^{+}$
Lemma: $\mathcal{M}_{ij} - \mathcal{K}_{ij} = \mathcal{L}w_j$

$$\mathbb{K} = \begin{pmatrix} \mathcal{K}_{11} & \mathcal{K}_{12} \\ \mathcal{K}_{21} & \mathcal{K}_{22} \end{pmatrix}$$

$$\boldsymbol{\Xi} = \mathbb{K}^{\star} \qquad \mathbb{G} = \begin{pmatrix} w_1\boldsymbol{\Xi}_{11} & w_1\boldsymbol{\Xi}_{12} \\ w_2\boldsymbol{\Xi}_{21} & w_2\boldsymbol{\Xi}_{22} \end{pmatrix}$$

$$\mathcal{A}^{\star} = \mathcal{N} + (\mathcal{R}_1^{-}, \mathcal{R}_2^{-})\,\mathbb{G}\left((\mathbb{M} - \mathbb{K})^{-}\mathbb{G}\right)^{\star}\begin{pmatrix} \mathcal{U}_1 \\ \mathcal{U}_2 \end{pmatrix}$$

B - Automaton approach

Clumps of the set of words $\mathcal{U} = \{aabaa, baab\}$

$\mathcal{E}_{w_1,w_2}$ extension set from w_1 to w_2

$$\mathcal{E}_{aabaa,aabaa} = \{baa, abaa\} \qquad \mathcal{E}_{baab,baab} = \{aab\}$$
$$\mathcal{E}_{aabaa,baab} = \{b\} \qquad \mathcal{E}_{baab,aabaa} = \{aa\}$$

Clumps of the set of words $\mathcal{U} = \{aabaa, baab\}$

$\mathcal{E}_{w_1,w_2}$ extension set from w_1 to w_2

$$\begin{array}{ll} \mathcal{E}_{aabaa,aabaa} = \{baa, abaa\} & \mathcal{E}_{baab,baab} = \{aab\} \\ \mathcal{E}_{aabaa,baab} = \{b\} & \mathcal{E}_{baab,aabaa} = \{aa\} \end{array}$$

Algorithm

1. Build the set of strings

Clumps of the set of words $\mathcal{U} = \{aabaa, baab\}$

$\mathcal{E}_{w_1,w_2}$ extension set from w_1 to w_2

$$\begin{array}{llll} \mathcal{E}_{aabaa,aabaa} & = \{baa, abaa\} & \mathcal{E}_{baab,baab} & = \{aab\} \\ \mathcal{E}_{aabaa,baab} & = \{b\} & \mathcal{E}_{baab,aabaa} & = \{aa\} \end{array}$$

Algorithm

1. Build the set of strings

$$\begin{array}{rlll} X = & \{aabaa.(\epsilon + \mathcal{E}_{aabaa,aabaa})\} & \cup & \{aabaa.\mathcal{E}_{aabaa,baab}\} \\ \cup & \{baab.(\epsilon + \mathcal{E}_{baab,baab})\} & \cup & \{baab.\mathcal{E}_{baab,aabaa}\} \end{array}$$

Clumps of the set of words $\mathcal{U} = \{aabaa, baab\}$

$\mathcal{E}_{w_1,w_2}$ extension set from w_1 to w_2

$$\begin{array}{llll} \mathcal{E}_{aabaa,aabaa} & = \{baa, abaa\} & \mathcal{E}_{baab,baab} & = \{aab\} \\ \mathcal{E}_{aabaa,baab} & = \{b\} & \mathcal{E}_{baab,aabaa} & = \{aa\} \end{array}$$

Algorithm

1. Build the set of strings

$$\begin{array}{rlll} X = & \{aabaa.(\epsilon + \mathcal{E}_{aabaa,aabaa})\} & \cup & \{aabaa.\mathcal{E}_{aabaa,baab}\} \\ \cup & \{baab.(\epsilon + \mathcal{E}_{baab,baab})\} & \cup & \{baab.\mathcal{E}_{baab,aabaa}\} \end{array}$$

2. Build a trie $\mathcal{T}$ on X

Clumps of the set of words $\mathcal{U} = \{aabaa, baab\}$

$\mathcal{E}_{w_1,w_2}$ extension set from w_1 to w_2

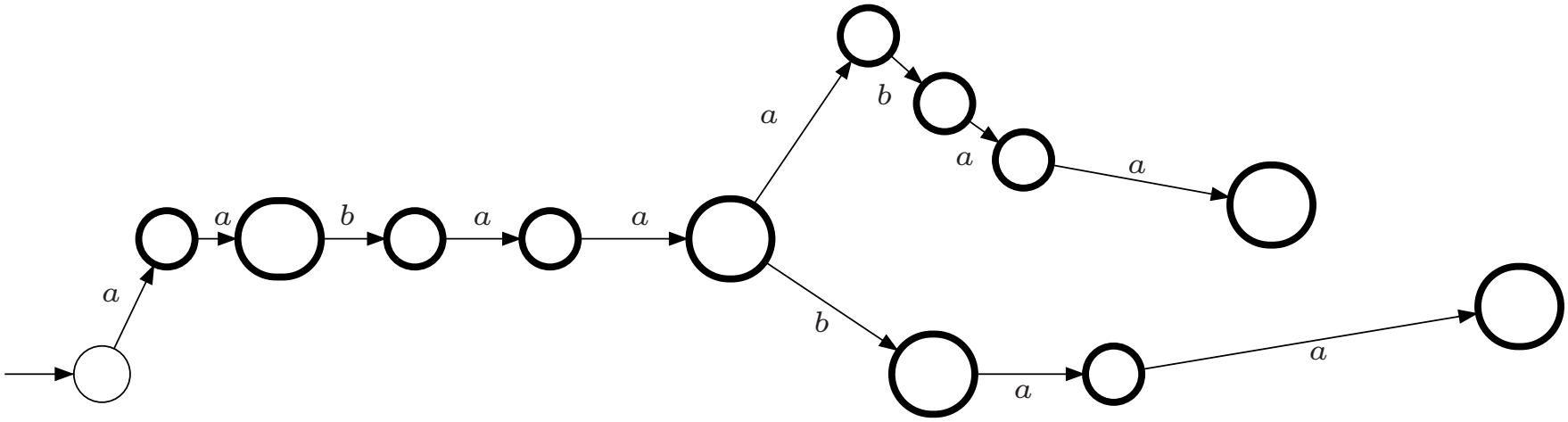
$$\begin{array}{llll} \mathcal{E}_{aabaa,aabaa} & = \{baa, abaa\} & \mathcal{E}_{baab,baab} & = \{aab\} \\ \mathcal{E}_{aabaa,baab} & = \{b\} & \mathcal{E}_{baab,aabaa} & = \{aa\} \end{array}$$

Algorithm

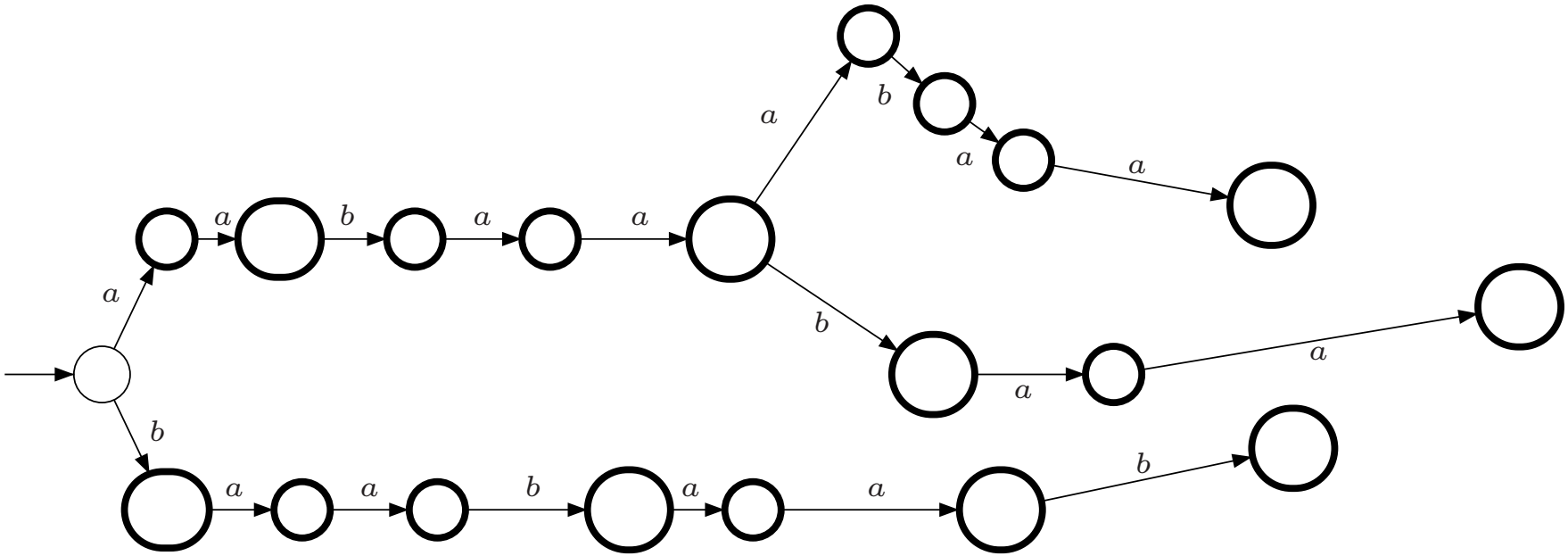
1. Build the set of strings
$$\begin{array}{rlll} X = & \{aabaa.(\epsilon + \mathcal{E}_{aabaa,aabaa})\} & \cup & \{aabaa.\mathcal{E}_{aabaa,baab}\} \\ \cup & \{baab.(\epsilon + \mathcal{E}_{baab,baab})\} & \cup & \{baab.\mathcal{E}_{baab,aabaa}\} \end{array}$$
2. Build a trie $\mathcal{T}$ on X
3. Build a Aho-Corasick like automaton upon $\mathcal{T}$. For each node ν of $\mathcal{T}$ with "access word" v, use the transition function δ

   $\delta(\nu, \ell) =$ node accessed by the **longuest prefix** in X that is **suffix** of $v.\ell$

$X = \{aabaa, aabaabaa, aabaaabaa, aabaab \quad \}$

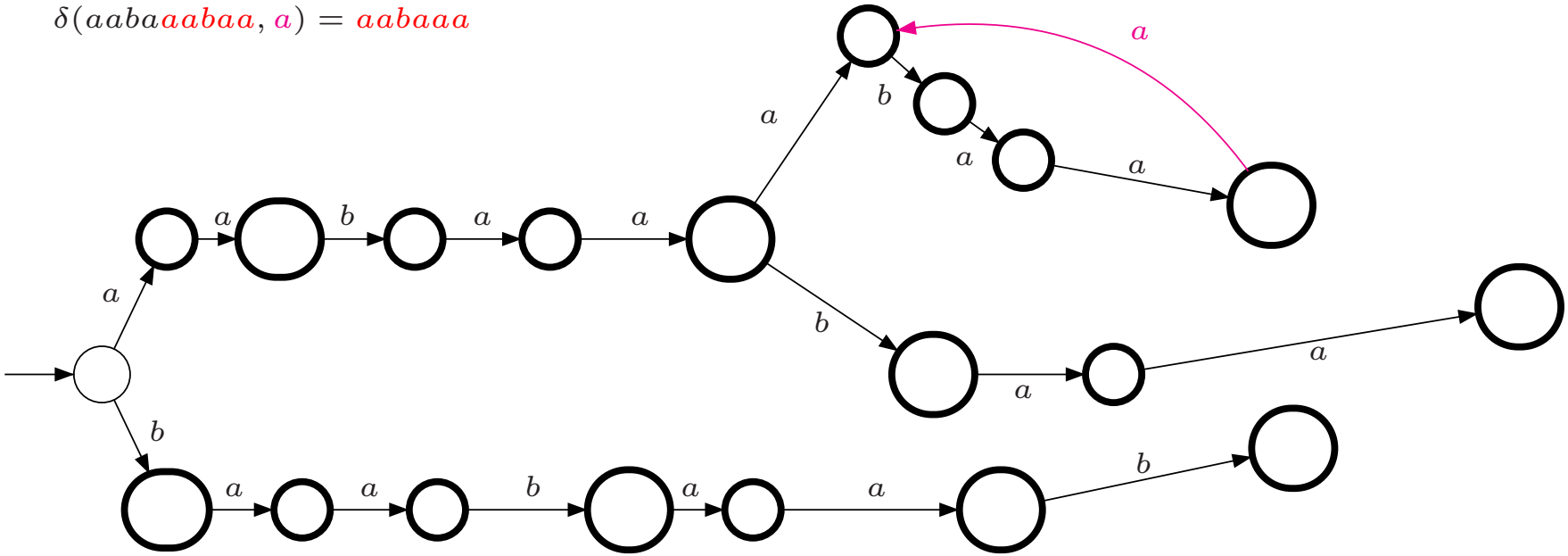


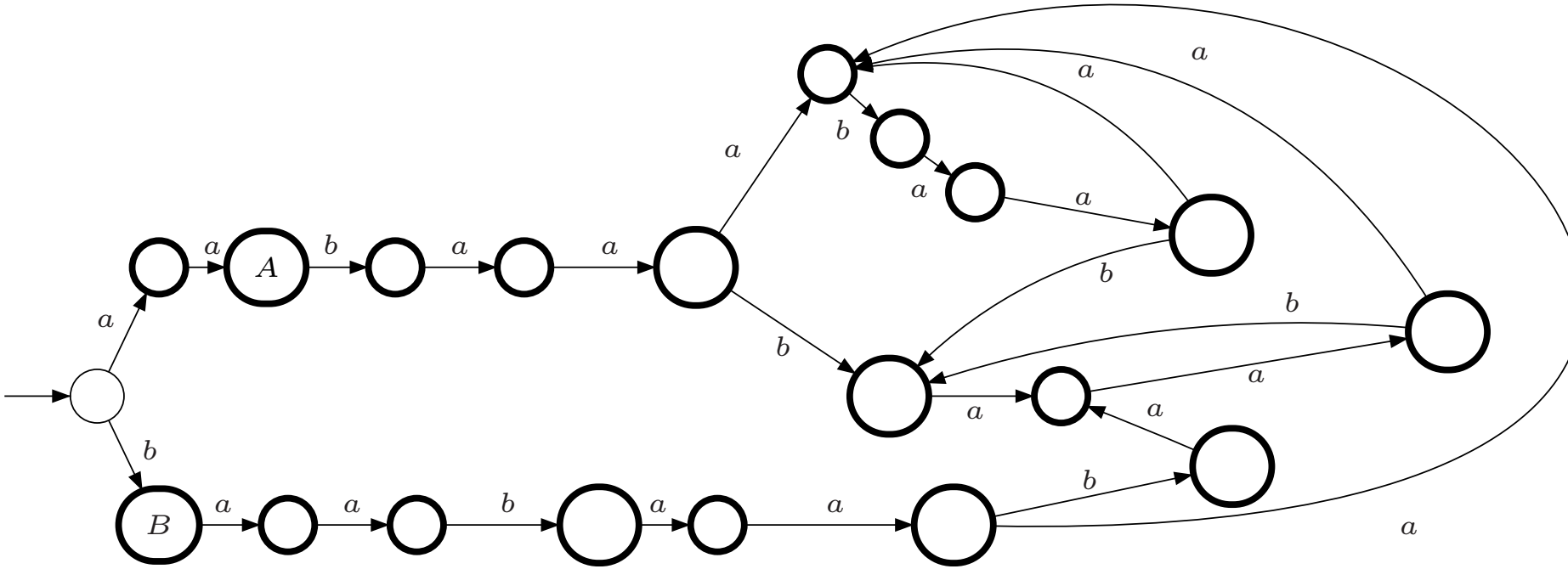
$X = \{aabaa, aabaabaa, aabaaabaa, aabaab, baab, baabaab\}$



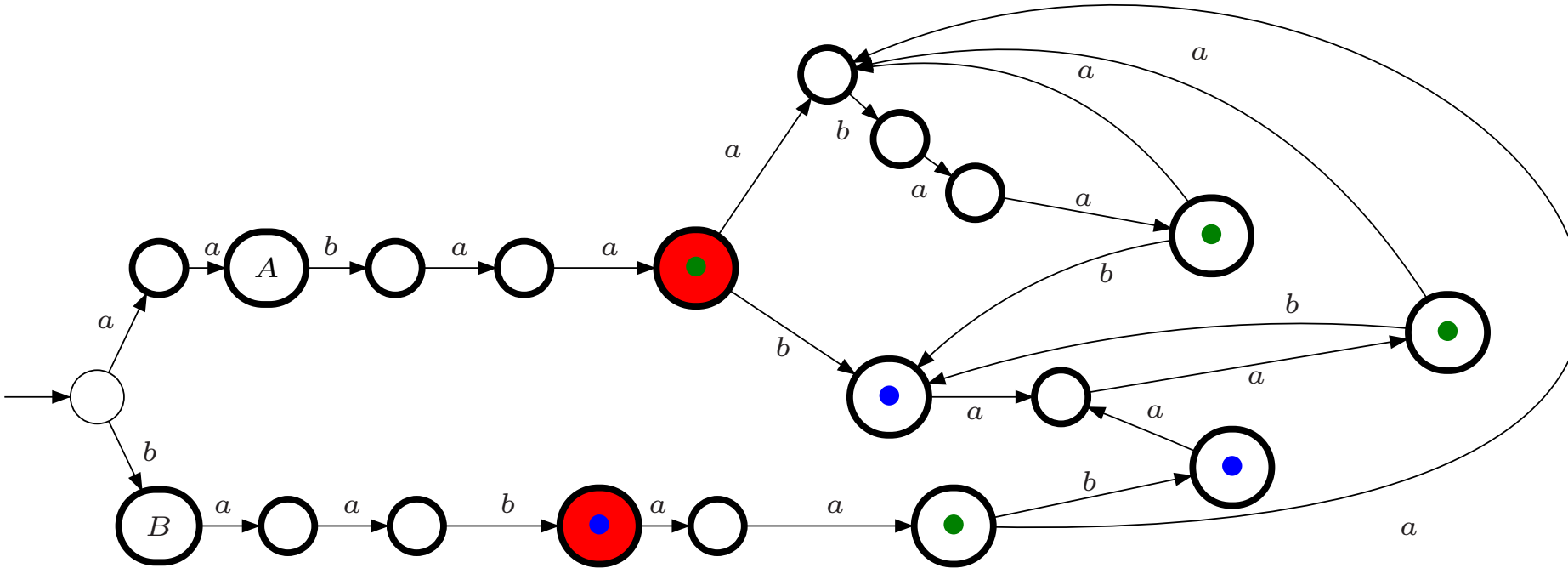
$X = \{aabaa, aabaabaa, aabaaabaa, aabaab, baab, baabaab\}$

$\delta(aabaaabaa, a) = aabaaa$



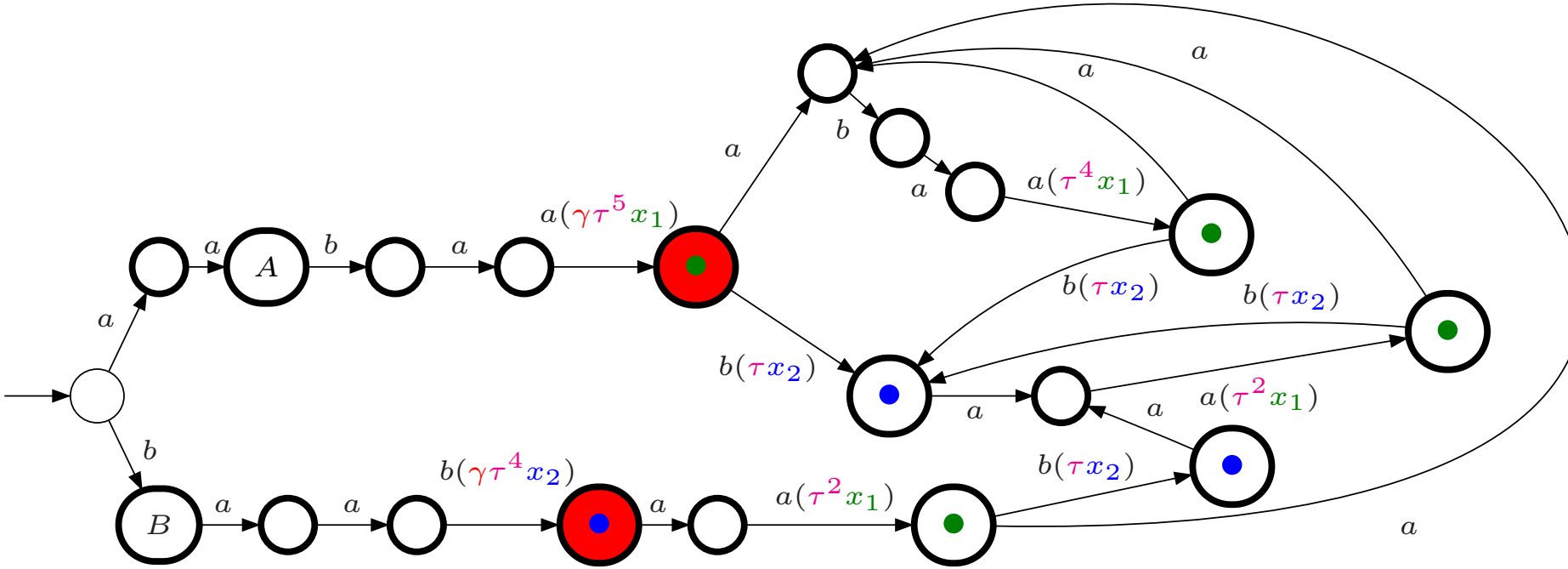


An automaton for $\mathcal{V} = \{v_1 = aabaa, v_2 = baab\}$. All transitions labeled by a and b ending respectively on state A and B are omitted.



An automaton for $\mathcal{V} = \{v_1 = aabaa, v_2 = baab\}$. All transitions labeled by a and b ending respectively on state A and B are omitted.

- ●, ● → the corresponding prefix (or state) ends with some occurrence of $aabaa, baab$.
- **red states** → states where we have entered a **new clump**



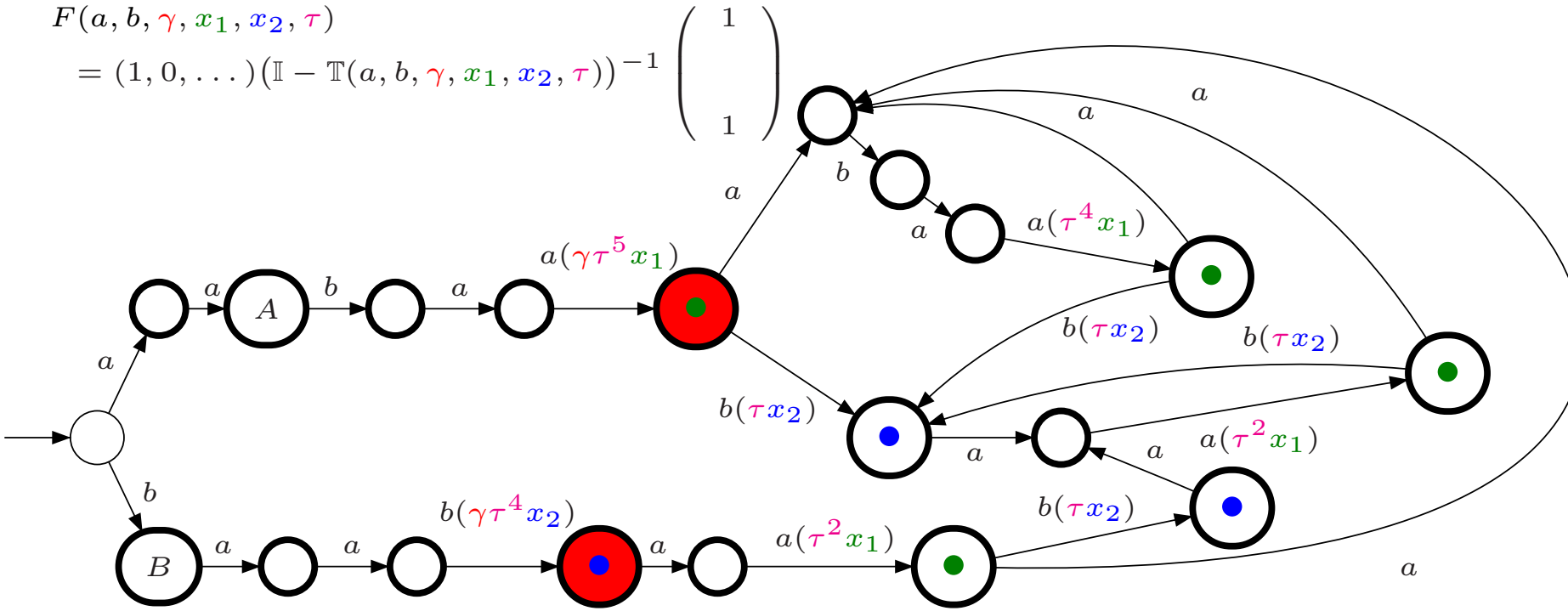
An automaton for $\mathcal{V} = \{v_1 = aabaa, v_2 = baab\}$. All transitions labeled by a and b ending respectively on state A and B are omitted.

- $\bullet$, $\bullet$ $\rightarrow$ the corresponding prefix (or state) ends with some occurrence of $aabaa, baab$.
- **red states** $\rightarrow$ states where we have entered a **new clump**

Formal weights on transitions

- $\gamma \rightarrow$ the number of clumps
- $\tau \rightarrow$ total length of clumps
- $x_1, x_2 \rightarrow$ occurrences of $aabaa, baab$

$$F(a,b,\gamma,x_1,x_2,\tau) = (1,0,\dots)\big(\mathbb{I}-\mathbb{T}(a,b,\gamma,x_1,x_2,\tau)\big)^{-1}\begin{pmatrix}1\\ \\ 1\end{pmatrix}$$

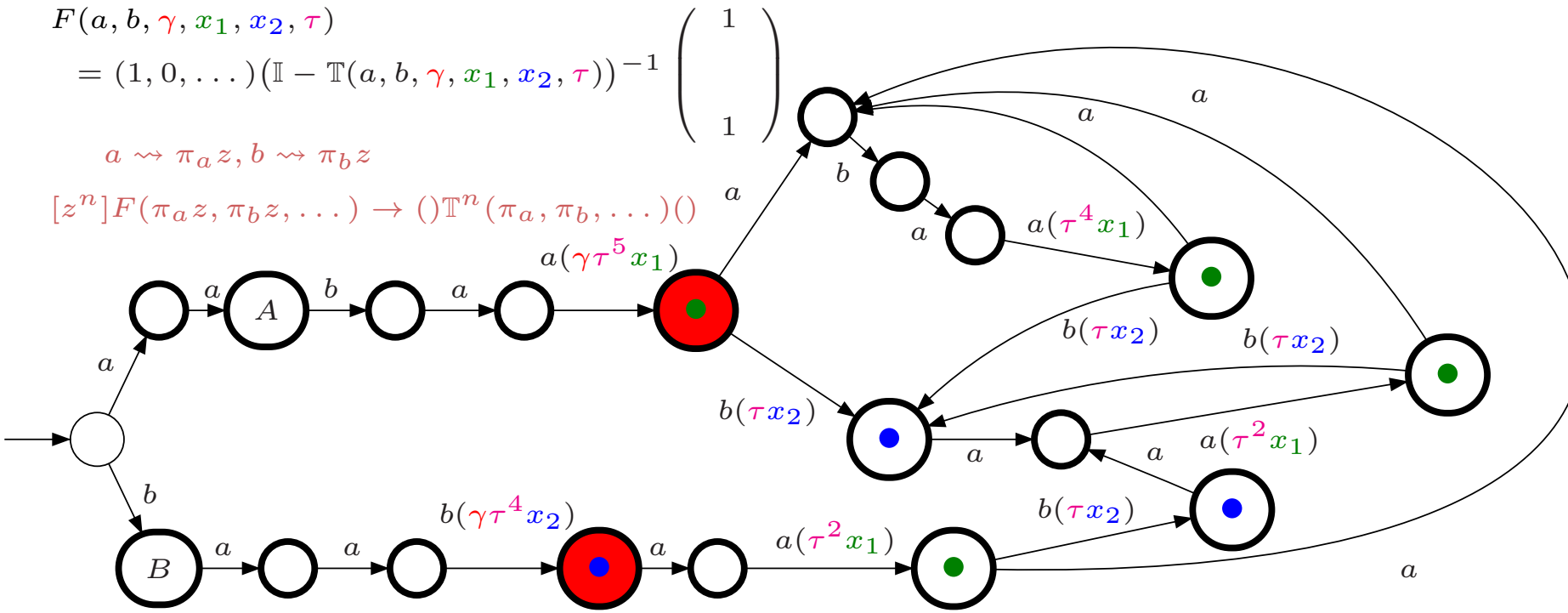


An automaton for $\mathcal{V} = \{v_1 = aabaa, v_2 = baab\}$. All transitions labeled by a and b ending respectively on state A and B are omitted.

- ●, ● → the corresponding prefix (or state) ends with some occurrence of $aabaa, baab$.
- **red states** → states where we have entered a **new clump**

Formal weights on transitions

- γ → the number of clumps
- τ → total length of clumps
- x_1, x_2 → occurrences of $aabaa, baab$



An automaton for $\mathcal{V} = \{v_1 = aabaa, v_2 = baab\}$. All transitions labeled by a and b ending respectively on state A and B are omitted.

- ●, ● → the corresponding prefix (or state) ends with some occurrence of $aabaa, baab$.
- **red states** → states where we have entered a **new clump**

Formal weights on transitions

- γ → the number of clumps
- τ → total length of clumps
- x_1, x_2 → occurrences of $aabaa, baab$

Asymptotic Limit Laws

- **One word**
  - $O_n^{\mathfrak{K}} = O(1)$ Poisson law for the number of clumps
- **General non-reduced sets**
  - $O_n^{\mathfrak{K}} = \Theta(n)$ Normal limit law (number of clumps, size covered)

Proofs

- Poisson law: Rouché theorem, singularity analysis
- Normal law: automaton, Perron-Frobenius for $\mathbb{T}(\dots)$, singularity analysis, large powers theorem

Complexity of computing the prefix code(s)

- **one word**

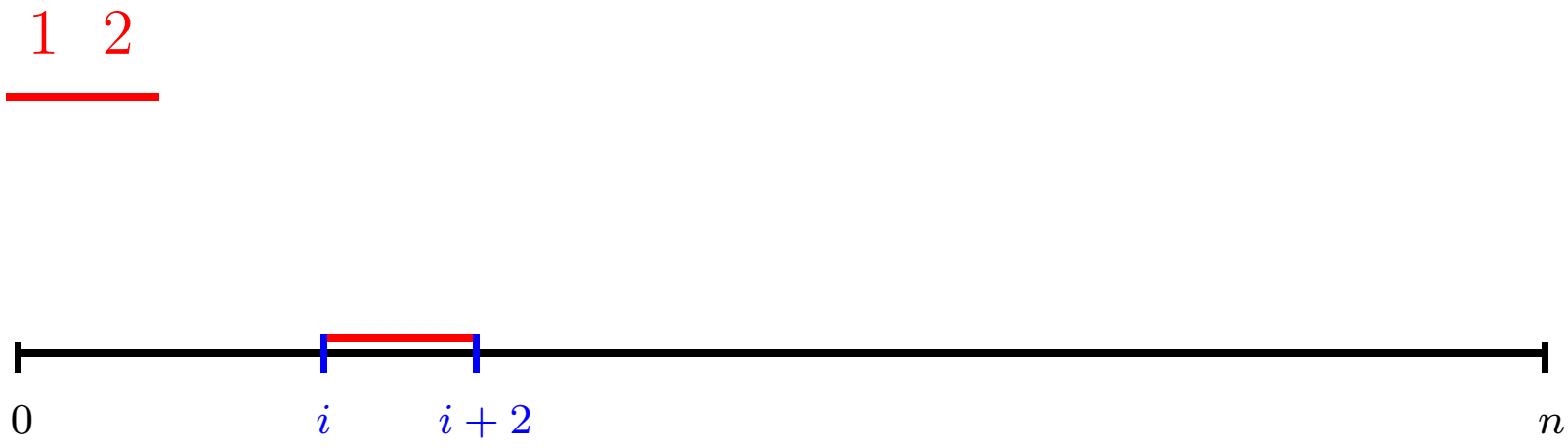
$$|\mathcal{K}| \log(|\mathcal{K}|)$$

- **several words** (reduced case)

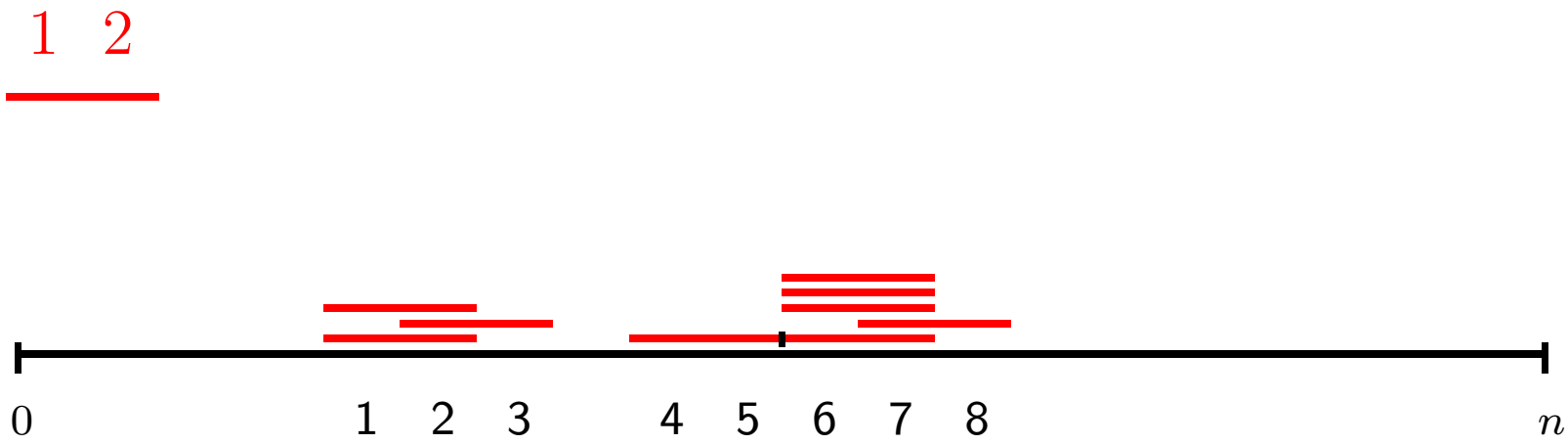
$$\left(\sum_{i,j} |\mathcal{K}_{i,j}| \right) \log \left(\sum_{i,j} |\mathcal{K}_{i,j}| \right)$$

Complexity of **insertion** of random keys in **tries**

Throwing flat dimers on an "integral" segment



Throwing flat dimers on an "integral" segment

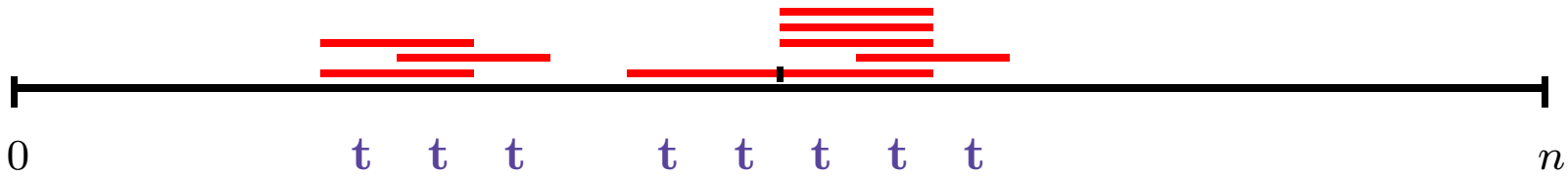


Size covered by the dimers?

Throwing flat dimers on an "integral" segment

$a \quad a$

stuttering occurrences of **aa** on $(\mathbf{a}+\mathbf{b})^{\mathbf{n}}$

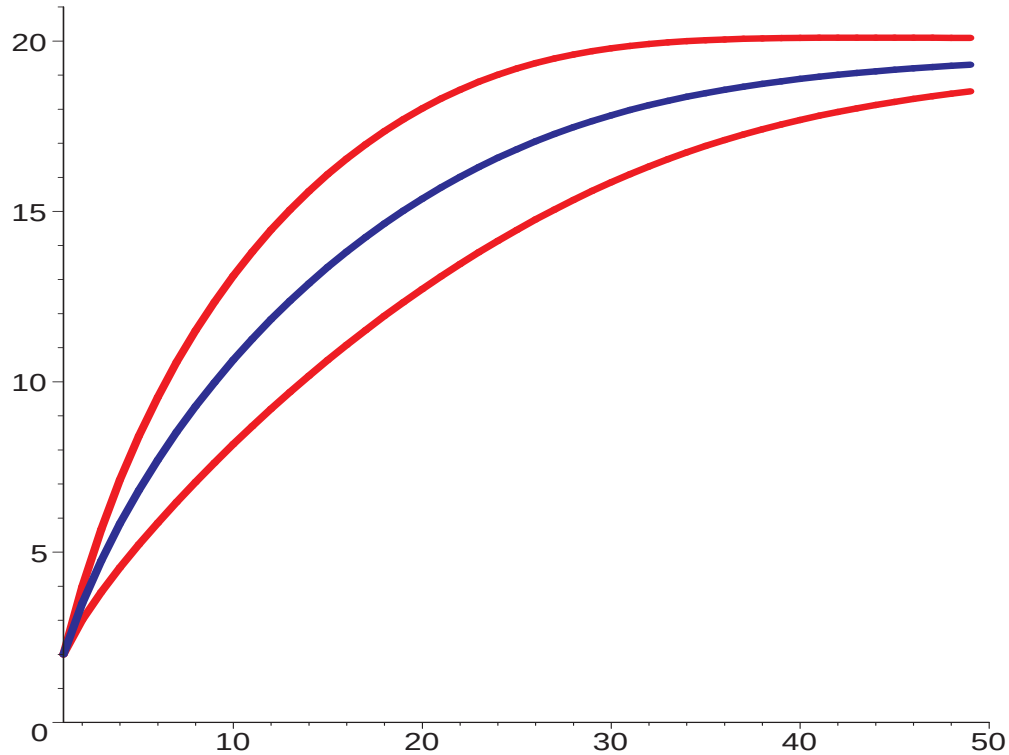


$$\mathbf{P}(a) = \mathbf{P}(b) = \frac{1}{2} \qquad \mathbf{P}(aa) = \frac{\mathbf{1}}{\mathbf{4}}$$

stuttering by **substituting** $\mathbf{x} \rightsquigarrow \dfrac{\mathbf{x}}{1 - \dfrac{\mathbf{1}}{\mathbf{4}}\mathbf{x}}$ **in** $\mathbf{\Gamma}(\mathbf{z}\mathbf{t}, \mathbf{x})$

Throwing flat dimers on an "integral" segment

Size covered



Number of dimers

$$\mathbf{n} = |\mathbf{segment}| = \mathbf{20}$$

Part III

Application of Clump Analysis to Genomics

Revisiting waiting times in DNA evolution

Biological Motivation

- **Promoters** are DNA sequences located upstream of the gene they regulate; regulation can be positive for enhancers or negative for repressors.
- The promoters contain binding sites for regulatory proteins such as **Transcription Factors** (**TFs**) that are **short stretches of DNA**.

Biological Motivation

- **Promoters** are DNA sequences located upstream of the gene they regulate; regulation can be positive for enhancers or negative for repressors.
- The promoters contain binding sites for regulatory proteins such as **Transcription Factors** (**TFs**) that are **short stretches of DNA**.

- **Waiting time:** **how long** it takes for a **Transcription Factor** to **appear** in a **promoter** under a **probabilistic model of evolution** helps understanding the **overall evolution of promoters** within species and between species?

From infinitesimal to discrete evolution model

- $\mathbb{Q}(t)dt$ evolution matrix for **infinitesimal time**
- $\mathbb{P}(t)$ evolution matrix **from time** x and **time** $x+t$

$$\mathbb{P}(t) = e^{\mathbb{Q}(t)} \qquad \text{(Karlin-Taylor 1975)}$$

- $\mathbb{P}(1) = (\pi_{\alpha\to\beta})$ evolution matrix for **one generation** (**20 years**), $\quad \alpha, \beta \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$

Probability of occurrence of a k-mer at time 1

- $S_n(0)$ **random DNA sequence** of **length** n at **time 0**
- $S_n(1)$ sequence obtained **from** $S_n(0)$ **by evolution** at **time 1**
- b a k-**mer** (word of **length** k over $\mathcal{A} = \{\mathsf{A},\mathsf{C},\mathsf{G},\mathsf{T}\}$)

- $\mathfrak{P}_n(b)$ probability that b
  - **occurs at time 1**
  - **while not occurring at time 0**

$$\mathfrak{P}_n(b) = \mathbb{P}(b \in S_n(1) \mid b \notin S_n(0))$$

Probability of occurrence of a k-mer at time 1

- $S_n(0)$ **random DNA sequence** of **length** n at **time 0**
- $S_n(1)$ sequence obtained **from** $S_n(0)$ **by evolution** at **time 1**
- b a k-**mer** (word of **length** k over $\mathcal{A} = \{\mathsf{A},\mathsf{C},\mathsf{G},\mathsf{T}\}$)

- $\mathfrak{P}_n(b)$ probability that b
  - **occurs at time 1**
  - **while not occurring at time 0**

$$\mathfrak{P}_n(b) = \mathbb{P}(b \in S_n(1) \mid b \notin S_n(0))$$

Expectation of the Waiting time $\mathfrak{E}_n(b)$

-

$$\mathfrak{E}_n(b) \approx \frac{1}{\mathfrak{P}_n(b)} \quad (\textbf{geometric distribution} - \text{BehVin2010})$$

Different computations of $\mathfrak{P}_n$

1. Behrens-Vingron (2010)
   - Approach **neglecting words correlation**.
   - **Efficient computation** of $\mathfrak{P}_n$ with respect to this assumption.
2. Behrens-Nicaud-P.N. (2012)
   - **Rigorous and efficient approach by automata**.
   - Approach **hiding the quasi-linear behaviour** of $\mathfrak{P}_n$
3. P.N. (NCMA2012)
   - **Non-efficient** approach by **clump analysis**, either by **combinatorics of words** or by **automata**.
   - **Proof by singularity analysis** of the **quasi-linear behaviour** of $\mathfrak{P}_n$

Initial $\nu(\alpha)$ and Substitution Probabilities $\pi_{\alpha\to\beta}$

α	$\nu(\alpha)$
A	0.23889
C	0.26242
G	0.25865
T	0.24004

$\rightsquigarrow$

substitution probability $\pi_{\alpha\to\beta}$ for one generation (20 years)

A	$\rightsquigarrow$	A	0.9999999763
A	$\rightsquigarrow$	C	$4.54999994943 \times 10^{-9}$
A	$\rightsquigarrow$	G	$1.57499995613 \times 10^{-8}$
A	$\rightsquigarrow$	T	$3.40000001733 \times 10^{-9}$
C	$\rightsquigarrow$	A	$6.14999993408 \times 10^{-9}$
C	$\rightsquigarrow$	C	0.99999996495
C	$\rightsquigarrow$	G	$7.14999984731 \times 10^{-9}$
C	$\rightsquigarrow$	T	$2.17499993935 \times 10^{-8}$
G	$\rightsquigarrow$	A	$2.17499993935 \times 10^{-8}$
G	$\rightsquigarrow$	C	$7.14999984731 \times 10^{-9}$
G	$\rightsquigarrow$	G	0.99999996495
G	$\rightsquigarrow$	T	$6.14999993408 \times 10^{-9}$
T	$\rightsquigarrow$	A	$3.40000001733 \times 10^{-9}$
T	$\rightsquigarrow$	C	$1.57499995613 \times 10^{-8}$
T	$\rightsquigarrow$	G	$4.54999994943 \times 10^{-9}$
T	$\rightsquigarrow$	T	0.9999999763

Numerical remarks

- **length** of promoters $n \in [500 - 2000]$
- $k \in \{5, 6, 7, 8, 9, 10\}$ for the k-**mers**
- **Mutation probability** $\pi_{\alpha \to \beta} \approx 10^{-9}$

Numerical remarks

- **length** of promoters $n \in [500 - 2000]$
- $k \in \{5, 6, 7, 8, 9, 10\}$ for the k-**mers**
- **Mutation probability** $\pi_{\alpha \to \beta} \approx 10^{-9}$

We have

- p_r: **probability of Mutation** to b from a r-neighbour of b with $r \geq 2$
  $p_r \leq n \times \pi^r_{\alpha \to \beta} \leq 2000 \times 10^{-18} < 2.10^{-6} \times \pi_{\alpha \to \beta}$
- q_s: **probability** that s 1-**neighbours simultaneously** mutate to b with $s \geq 2$
  $q_s \leq n \times \pi^s_{\alpha \to \beta} \leq 2000 \times 10^{-18} < 2.10^{-6} \times \pi_{\alpha \to \beta}$

Numerical remarks

- **length** of promoters $n \in [500 - 2000]$
- $k \in \{5, 6, 7, 8, 9, 10\}$ for the k-**mers**
- **Mutation probability** $\pi_{\alpha \to \beta} \approx 10^{-9}$

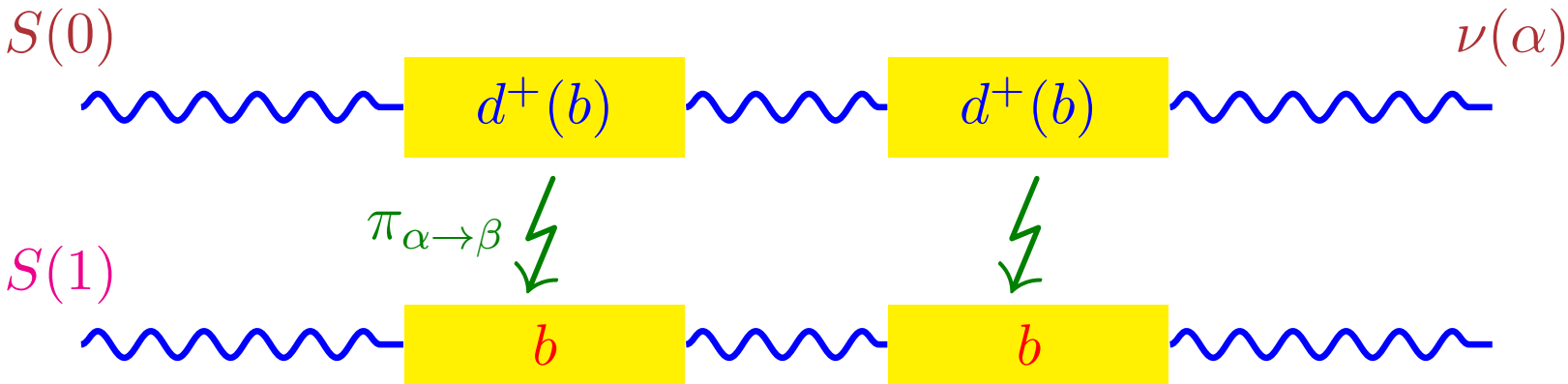
We have

- p_r: **probability of Mutation** to b from a r-neighbour of b with $r \geq 2$
  $p_r \leq n \times \pi^r_{\alpha \to \beta} \leq 2000 \times 10^{-18} < 2.10^{-6} \times \pi_{\alpha \to \beta}$
- q_s: **probability** that s 1-**neighbours simultaneously** mutate to b with $s \geq 2$
  $q_s \leq n \times \pi^s_{\alpha \to \beta} \leq 2000 \times 10^{-18} < 2.10^{-6} \times \pi_{\alpha \to \beta}$

Therefore assuming a single mutation in the promoter is numerically sound

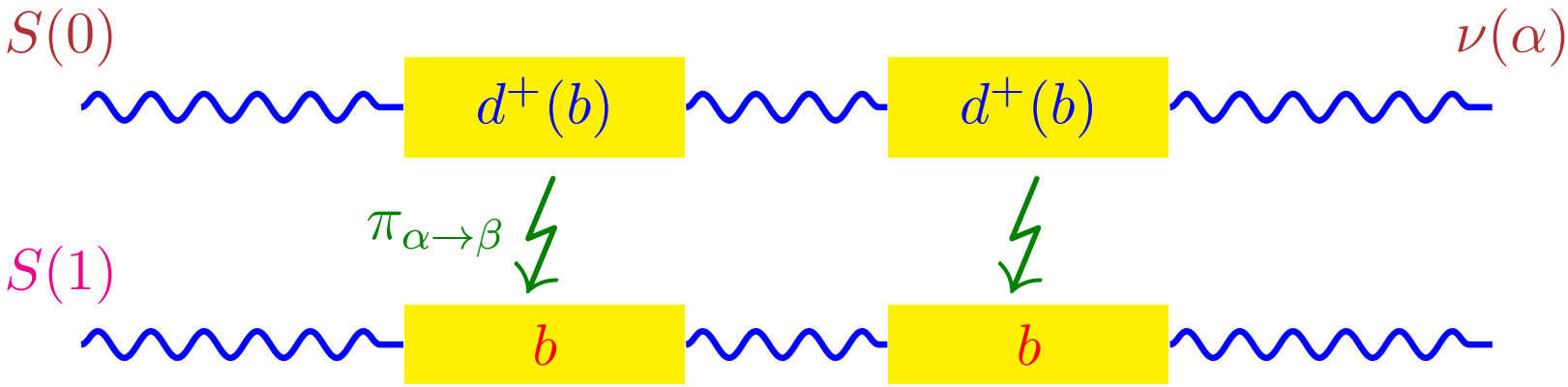
Behrens-Vingron 2010

- $d^+(b)$ neighbors of b by substitution



Behrens-Vingron 2010

- $d^+(b)$ neighbors of b by substitution



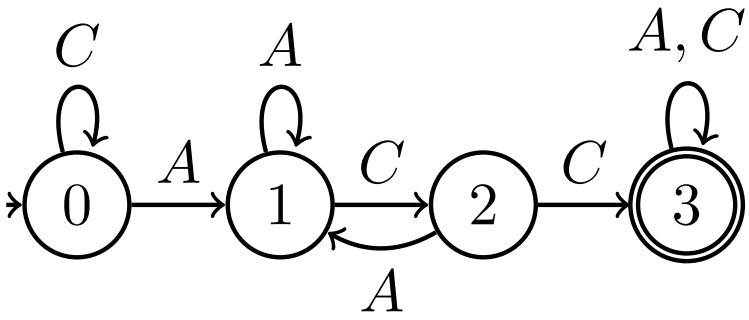
$$\begin{cases} \mathfrak{P}_n \approx \displaystyle\sum_{i=1}^{\lfloor n/k \rfloor} (-1)^{i+1} \binom{n-i(k-1)}{i} \Phi^i \\ \Phi = \displaystyle\sum_{(a_1,\ldots,a_k)\in\mathcal{A}^k\setminus\{b_1,\ldots,b_k\}} \nu(a_1)\times\cdots\times\nu(a_k)\cdot\prod_{j=1}^{k}\pi_{a_i\to b_i}(1) \end{cases}$$

Behrens-Nicaud-P.N. 2012

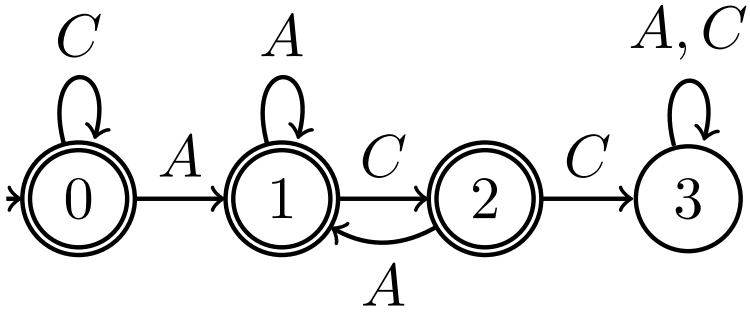
Construct an **automaton**

- on the **alphabet** $\Sigma = \mathcal{A} \times \mathcal{A}$ with $\mathcal{A} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$
- **recognizing sequences** $\mathsf{S}(b) = S(0) \otimes S(1)$
- **such that**
  1. $b \notin S(0)$
  2. $b \in S(1)$

Using the Knuth-Morris-Pratt automaton



$$\mathcal{M}_{\mathsf{ACC}} = \{Q, \delta, s = 0, F\}$$

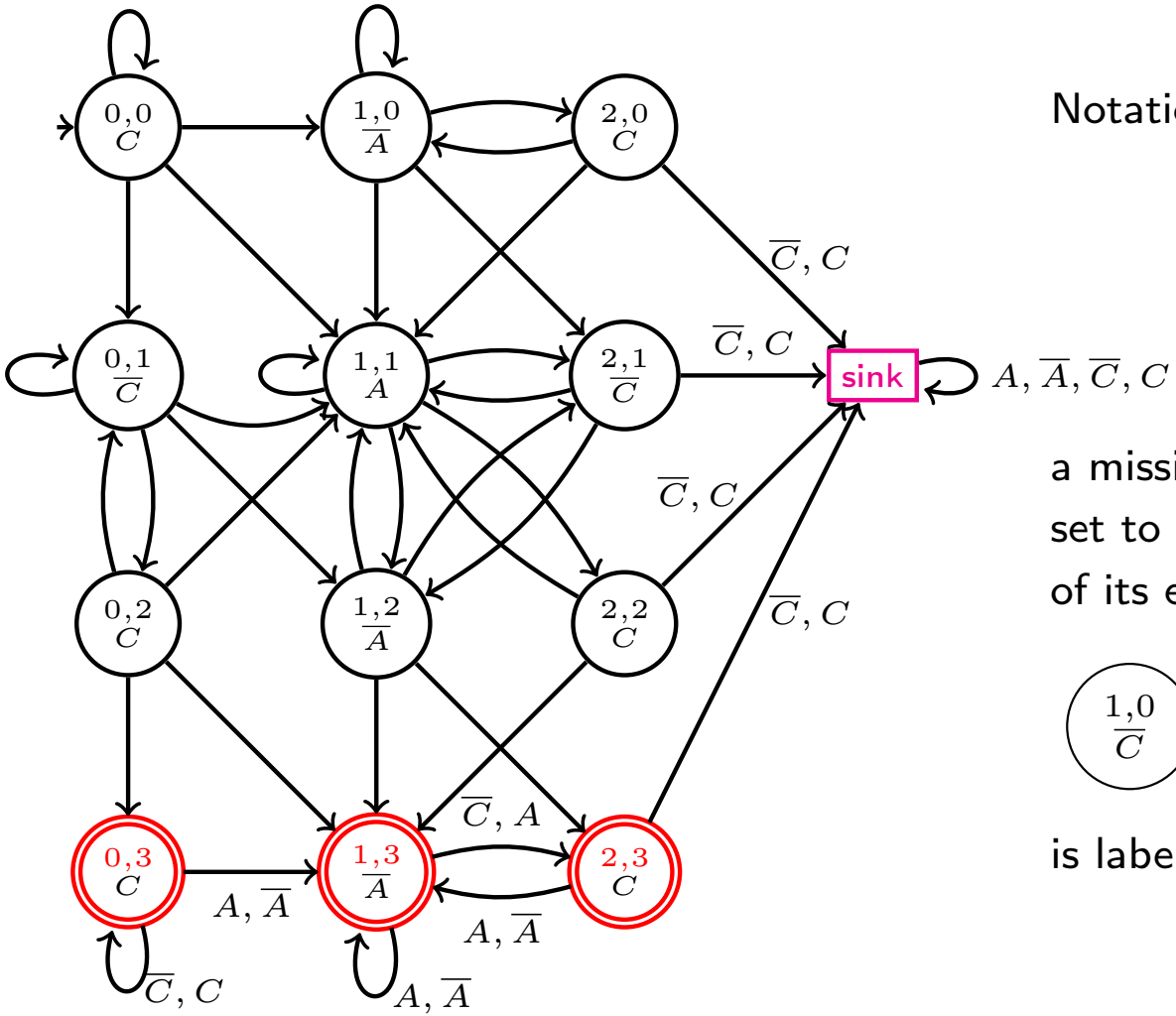


$$\overline{\mathcal{M}}_{\mathsf{ACC}} = \{Q, \delta, s = 0, Q \setminus F\}$$

$$\begin{cases} \mathcal{M}_b = (Q = \{0, \ldots, k\}, \delta_b, 0, \{k\}) \\ \overline{\mathcal{M}}_b = (Q = \{0, \ldots, k\}, \delta_b, 0, \{0, \ldots, k-1\}) \\ \mathcal{N}_b = \overline{\mathcal{M}}_b \otimes \mathcal{M}_b = (Q \times Q, \Delta, q'_0 = (0,0), F' = \{0,\ldots,k-1\} \times \{k\}) \end{cases}$$

$$\Delta((r,s),(\alpha,\beta)) = (\delta_b(r,\alpha), \delta_b(s,\beta))$$

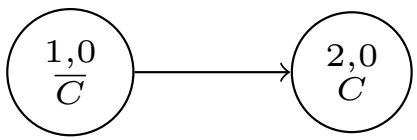
The automaton $\mathcal{N}_{\mathsf{ACC}} = \overline{\mathcal{M}}_{\mathsf{ACC}} \otimes \mathcal{M}_{\mathsf{ACC}}$ with matrix $\mathbb{P}$



Notations for the transitions:

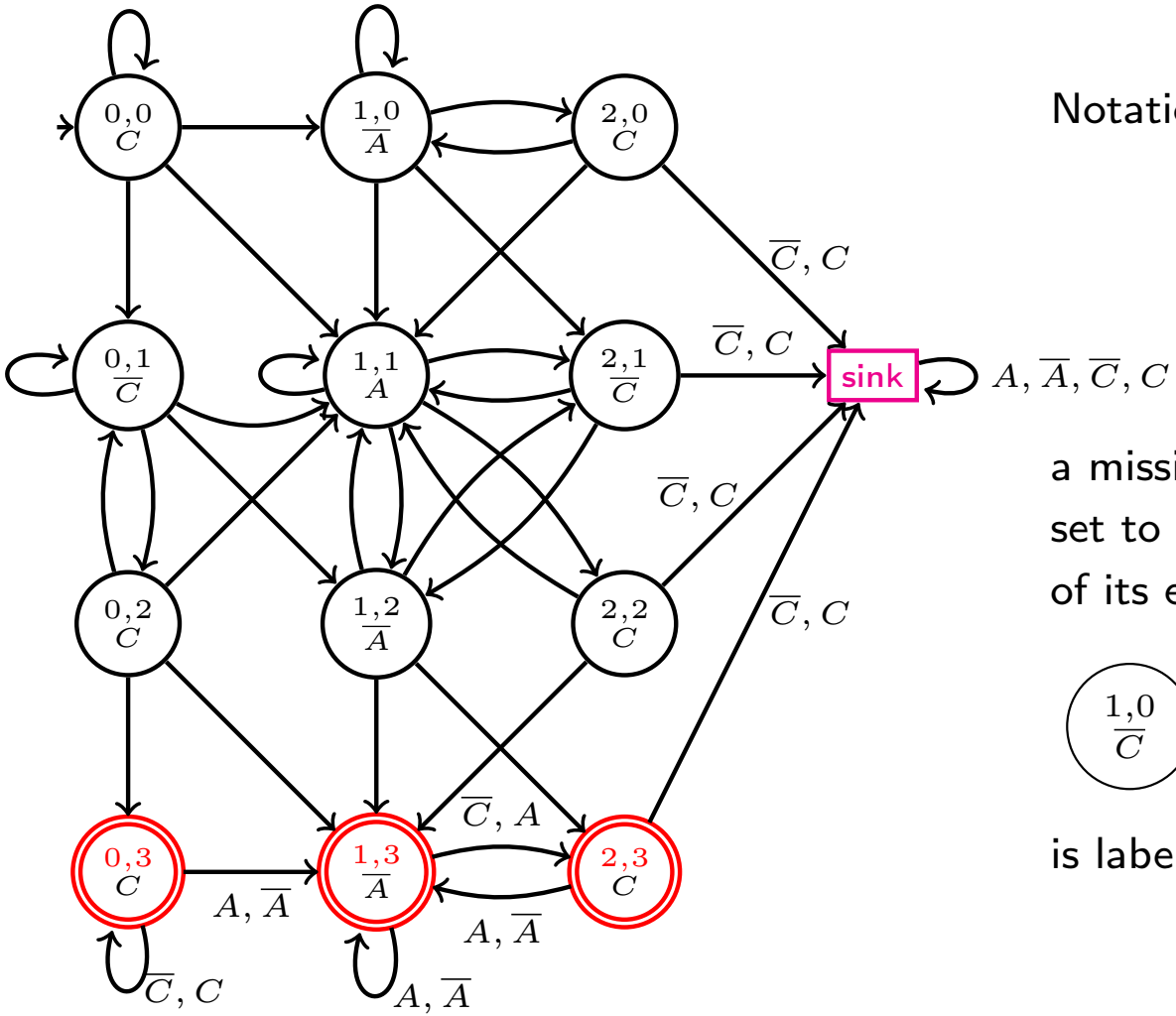
$$\begin{cases} A = \binom{A}{A}, & C = \binom{C}{C} \\ \overline{A} = \binom{A}{C}, & \overline{C} = \binom{C}{A} \end{cases}$$

a missing label of a transition is set to the letter at the bottom of its ending state



is labelled by C

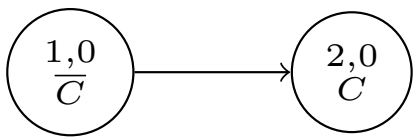
The automaton $\mathcal{N}_{\mathsf{ACC}} = \overline{\mathcal{M}}_{\mathsf{ACC}} \otimes \mathcal{M}_{\mathsf{ACC}}$ with matrix $\mathbb{P}$



Notations for the transitions:

$$\begin{cases} A = \binom{A}{A}, & C = \binom{C}{C} \\ \overline{A} = \binom{A}{C}, & \overline{C} = \binom{C}{A} \end{cases}$$

a missing label of a transition is set to the letter at the bottom of its ending state



is labelled by C

$$\mathfrak{P}_n = \mathbf{P}(S_n(1) \in \mathcal{A}^\star b \mathcal{A}^\star | S_n(0) \notin \mathcal{A}^\star b \mathcal{A}^\star) = \frac{V_{q'_0} \mathbb{P}^n {V_{F'}}^t}{1 - V_{q'_0} \mathbb{P}^n V_{\mathsf{sink}}^t}$$

Results for 5-mers of DNA

	BNN		BV		
	$\mathbf{E}_{\mathrm{BNN}}(T_{1000})/10^6$	Rank	$\mathbf{E}_{\mathrm{BV}}(T_{1000})/10^6$	Rank	$\frac{\mathbf{E}_{\mathrm{BNN}}(T_{1000})}{\mathbf{E}_{\mathrm{BV}}(T_{1000})}$
CCCCC	9,105	1021	6,304	1	1.44
GGGGG	9,570	1022	6,666	142	1.44
TTTTT	10,401	1023	7,457	993	1.39
AAAAA	10,656	1024	7,654	1024	1.39
CGCGC	7,047	699	6,446	11	1.09
TCCCC	7,076	737	6,477	17	1.09
CCCCT	7,076	738	6,477	21	1.09
GCGCG	7,127	787	6,518	31	1.09
CTCTC	7,263	883	6,679	148	1.09
...	...	...	...	...	...

$$\left\{\begin{array}{l} 4\% \text{ of the } 5\text{-mers} \\ 0.2\% \text{ of the } 7\text{-mers} \\ 0.002\% \text{ of the } 10\text{-mers} \end{array}\right. \Bigg| \text{ verify } \frac{\mathbf{E}_{\mathrm{BNN}}(T_{1000})}{\mathbf{E}_{\mathrm{BV}}(T_{1000})} > 1.05\%$$

Putative-hit positions.

- Given a **sequence $S(0)$ not containing a k-mer b**,
- a **putative-hit position** is any position of $S(0)$ that can **lead by a mutation to an occurrence of b in $S(1)$**,
- where we assume that a **single** mutation has occurred.

$$S(0) = \texttt{CCCAACAC}, \quad b = \texttt{ACC} \qquad \rightsquigarrow \qquad \underline{S}(0) = \underline{\texttt{C}}\texttt{CCA}\underline{\texttt{A}}\texttt{CAC},$$

putative-hit positions **underlined** in $\underline{S}(0)$.

Putative-hit positions.

- Given a **sequence $S(0)$ not containing a k-mer b**,
- a **putative-hit position** is any position of $S(0)$ that can **lead by a mutation to an occurrence of b in $S(1)$**,
- where we assume that a **single** mutation has occurred.

$$S(0) = \texttt{CCCAACAC}, \quad b = \texttt{ACC} \qquad \rightsquigarrow \qquad \underline{S}(0) = \underline{\texttt{C}}\texttt{CCA}\underline{\texttt{A}}\texttt{CAC},$$

putative-hit positions **underlined** in $\underline{S}(0)$.

In a random sequence of length n, let

- $H^{(n)}_{\texttt{A}\to\texttt{C}}$ number of putative-hit-positions $\texttt{A} \to \texttt{C}$,
- $H^{(n)}_{\texttt{C}\to\texttt{A}}$ number of putative-hit-positions $\texttt{C} \to \texttt{A}$,

Then

$$\mathfrak{P}_n \approx \mathbf{E}\big(H^{(n)}_{\texttt{A}\to\texttt{C}}\big) \times \pi_{\texttt{A}\to\texttt{C}} + \mathbf{E}\big(H^{(n)}_{\texttt{C}\to\texttt{A}}\big) \times \pi_{\texttt{C}\to\texttt{A}}$$

Computing via generating functions

Aim:

Compute

$$F_b(z, t_{\mathtt{A}\to\mathtt{C}}, t_{\mathtt{C}\to\mathtt{A}}) = \sum_{n\geq 0} \sum_{0\leq i\leq n-|b|} \sum_{0\leq j\leq n-|b|} f_{n,i,j} t^i_{\mathtt{A}\to\mathtt{C}}, t^j_{\mathtt{C}\to\mathtt{A}} z^n$$

where $f_{n,i,j}$ is the probability that a sequence $S_n(0)$ **with no** b, of length n, contains

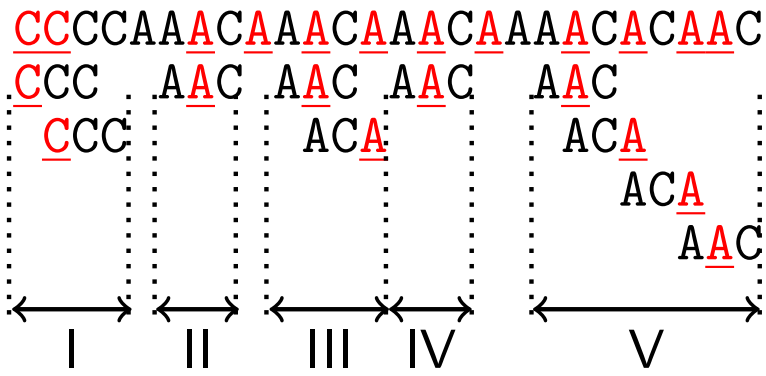
- i putative-hit positions $\mathtt{A}\to\mathtt{C}$
- and j putative-hit positions $\mathtt{C}\to\mathtt{A}$

We have

$$\mathfrak{P}_n = [z^n] \left(\pi_{\mathtt{A}\to\mathtt{C}} \left.\frac{\partial F(z, t_{\mathtt{A}\to\mathtt{C}}, 1)}{\partial t_{\mathtt{A}\to\mathtt{C}}}\right|_{t_{\mathtt{A}\to\mathtt{C}}=1} + \pi_{\mathtt{C}\to\mathtt{A}} \left.\frac{\partial F(z, 1, t_{\mathtt{C}\to\mathtt{A}})}{\partial t_{\mathtt{C}\to\mathtt{A}}}\right|_{t_{\mathtt{C}\to\mathtt{A}}=1} \right)$$

Putative-Hit-Positions and clump analysis

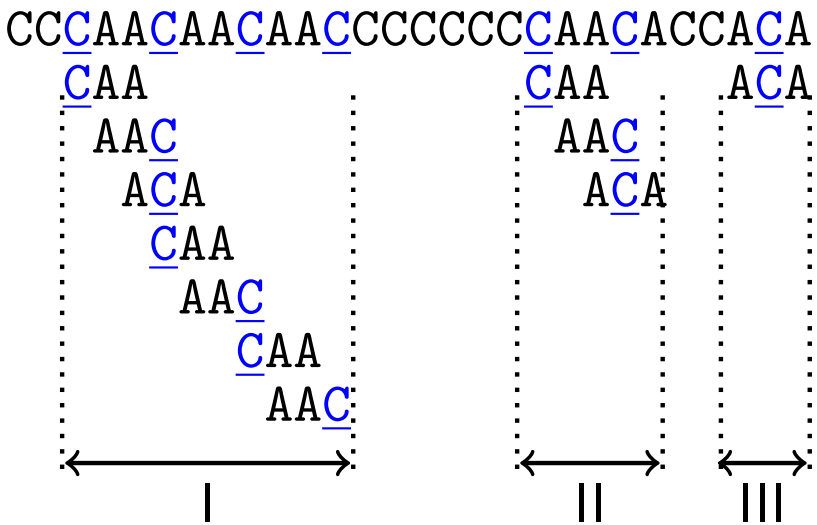
$\mathcal{A} = \{\mathtt{A}, \mathtt{C}\}$ $b = \mathtt{ACC} \longrightarrow d(\mathtt{ACC}, 1) = \{\underline{\mathtt{C}}\mathtt{CC}, \mathtt{A}\underline{\mathtt{A}}\mathtt{C}, \mathtt{AC}\underline{\mathtt{A}}\}$



- (left) $b = \mathtt{ACC}$ - in clump I, when the right extension of a clump adds a new putative-hit position, this position is not necessarily in the extension, but possibly backwards left

Putative-Hit-Positions and clump analysis

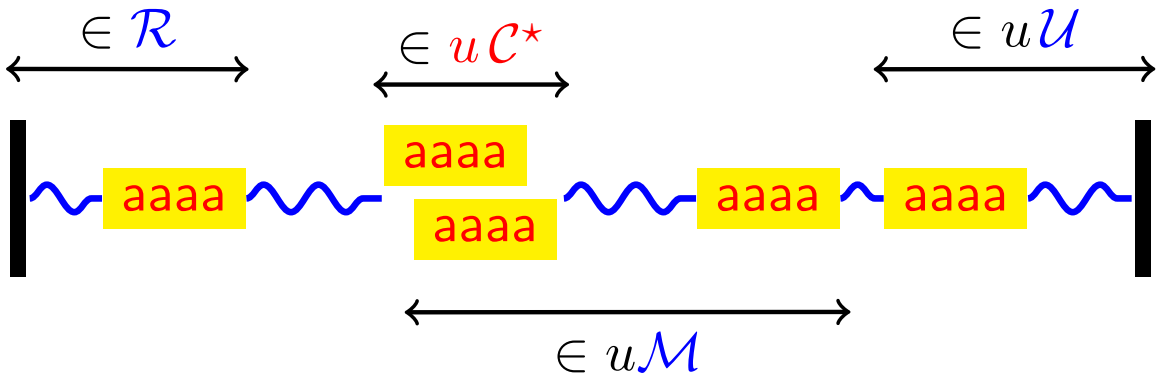
$\mathcal{A} = \{\mathtt{A}, \mathtt{C}\} \quad b' = \mathtt{AAA} \longrightarrow d(\mathtt{AAA}, 1) = \{\underline{\mathtt{C}}\mathtt{AA}, \mathtt{A}\underline{\mathtt{C}}\mathtt{A}, \mathtt{AA}\underline{\mathtt{C}}\}$



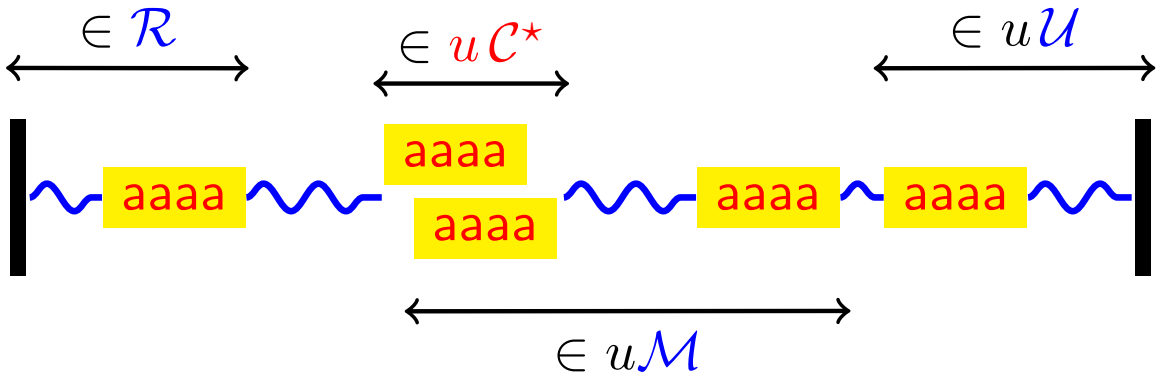
- (right)$b' = \mathtt{AAA}$ - clump I contains 7 occurrences of $d(\mathtt{AAA})$, but only 4 putative-hit positions for $b' = \mathtt{AAA}$. The number of word occurrences is not the relevant statistics for counting putative-hit positions

A - Language approach

Clump Analysis (Bassino-Clément-Fayolle-P.N. 2008)



Clump Analysis (Bassino-Clément-Fayolle-P.N. 2008)

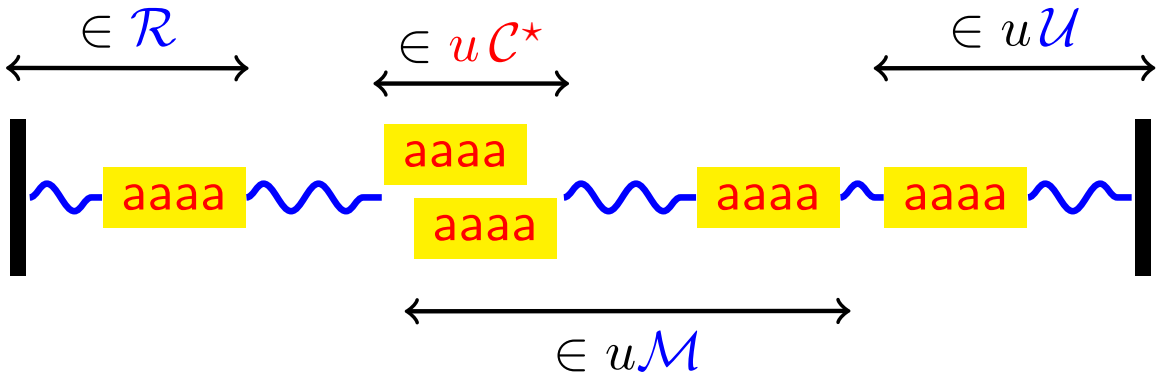


- *residual* language $\mathcal{D} = \mathcal{L}.u^-$: $\mathcal{D} = \{h,\ h \cdot u \in \mathcal{L}\}$
- $\mathcal{L}_2 - \mathcal{L}_1 = \mathcal{L}_2 \setminus \mathcal{L}_1 = \{h;\ h \in \mathcal{L}_2, h \notin \mathcal{L}_1\}$

Combinatorial decomposition (one word)

$$\mathcal{A}^\star = \mathcal{N} + \mathcal{R}u^- u\,\mathcal{C}^\star \Big((\mathcal{M} - \mathcal{K}) u^- u\,\mathcal{C}^\star \Big)^\star \mathcal{U}$$

Clump Analysis (Bassino-Clément-Fayolle-P.N. 2008)



- *residual* language $\mathcal{D} = \mathcal{L}.u^-$: $\mathcal{D} = \{h,\ h \cdot u \in \mathcal{L}\}$
- $\mathcal{L}_2 - \mathcal{L}_1 = \mathcal{L}_2 \setminus \mathcal{L}_1 = \{h;\ h \in \mathcal{L}_2, h \notin \mathcal{L}_1\}$

Combinatorial decomposition (one word)

$$\mathcal{A}^\star = \mathcal{N} + \mathcal{R}u^- u\,\mathcal{C}^\star \Big((\mathcal{M} - \mathcal{K}) u^- u\,\mathcal{C}^\star \Big)^\star \mathcal{U}$$

$$= \mathcal{N} + \mathcal{R}u^- u\,\mathcal{K}^\star \Big((\mathcal{M} - \mathcal{K}) u^- u\,\mathcal{K}^\star \Big)^\star \mathcal{U}$$

$$= \mathcal{N} + \mathcal{R}u^- \mathbf{S} \Big((\mathcal{M} - \mathcal{K}) u^- \mathbf{S} \Big)^\star \mathcal{U}$$

Clumps: $\mathbf{S} = u\,\mathcal{C}^\star = u\,\mathcal{K}^\star$

Constrained Guibas-Odlyzko languages

Example: $b = \mathtt{AA}, \qquad d_\ell(b) = (\mathtt{AC}, \mathtt{CA})$

- We need **avoiding** $\mathtt{AA}$ in $S(0)$ and therefore **inthe Right**, **Minimal** and **Ultimate** languages
- **Build** the **Régnier-Szpankowski languages** **for the pattern** $(\mathtt{AC}, \mathtt{CA}, \mathtt{AA})$

$$\mathcal{L} = \mathcal{N} + (\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3) \begin{pmatrix} \mathcal{M}_{11} & \mathcal{M}_{12} & \mathcal{M}_{13} \\ \mathcal{M}_{21} & \mathcal{M}_{22} & \mathcal{M}_{23} \\ \mathcal{M}_{31} & \mathcal{M}_{32} & \mathcal{M}_{33} \end{pmatrix} \begin{pmatrix} \mathcal{U}_1 \\ \mathcal{U}_2 \\ \mathcal{U}_3 \end{pmatrix}$$

$$\widehat{\mathcal{L}} = \mathcal{N} + (\mathcal{R}_1, \mathcal{R}_2) \begin{pmatrix} \mathcal{M}_{11} & \mathcal{M}_{12} \\ \mathcal{M}_{21} & \mathcal{M}_{22} \end{pmatrix} \begin{pmatrix} \mathcal{U}_1 \\ \mathcal{U}_2 \end{pmatrix}$$

Notations: write $\widehat{\mathcal{N}}, \widehat{\mathcal{R}_i}, \widehat{\mathcal{M}_{ij}}, \widehat{\mathcal{U}_j}$ for **constrained languages**

Constrained clumps

- **finite** code languages $\mathcal{K}_{ij}$ **easy to compute**
- we must however **avoid** the **forbidden word** b while **extending clumps**
- $v_i,\ v_j \in d_\ell(b) \quad \rightsquigarrow \quad \widehat{\mathcal{K}}_{ij} = \{h \in \mathcal{K}_{ij}; \qquad |v_i.h|_b = 0\}$

  sets $\mathcal{K}_{ij}$ finite $\quad \Longrightarrow \quad$ computation of $\widehat{\mathcal{K}}_{ij}$ by string-matching

Constrained clumps

- **finite** code languages $\mathcal{K}_{ij}$ **easy to compute**
- we must however **avoid** the **forbidden word** b while **extending clumps**
- $v_i,\ v_j \in d_\ell(b) \quad \rightsquigarrow \quad \widehat{\mathcal{K}}_{ij} = \{h \in \mathcal{K}_{ij}; \qquad |v_i.h|_b = 0\}$

  sets $\mathcal{K}_{ij}$ finite $\quad \Longrightarrow \quad$ computation of $\widehat{\mathcal{K}}_{ij}$ by string-matching

- **Decomposition by constrained clumps**

$$\widehat{\mathcal{A}_b^\star} = \widehat{\mathcal{N}} + (\widehat{\mathcal{R}}_1 v_1^-, \ldots, \widehat{\mathcal{R}}_r v_r^-)\ \widehat{\mathbb{G}}\Big((\widehat{\mathbb{M}} - \widehat{\mathbb{K}})^- \widehat{\mathbb{G}}\Big)^\star \begin{pmatrix} \widehat{\mathcal{U}}_1 \\ \vdots \\ \widehat{\mathcal{U}}_r \end{pmatrix}$$

$$\text{with} \begin{cases} \widehat{\mathbb{K}} = (\widehat{\mathcal{K}}_{ij}), \\ \widehat{\mathbb{S}} = \widehat{\mathbb{K}}^\star, \\ \widehat{\mathbb{G}} = \left(v_i \widehat{\mathbb{S}}_{ij}\right) \end{cases}$$

Counting putative hit-positions

$b = \mathtt{ACAC} \qquad \mathbf{P}(\mathtt{A}) = \mathbf{P}(\mathtt{C}) = 1/2$

$$d_\ell(b) = (\mathtt{AAAC}, \mathtt{ACAA}, \mathtt{ACCC}, \mathtt{CCAC}) \quad d_\ell(b)(z,t) = \left(\frac{z^4 t}{16}, \frac{z^4 t}{16}, \frac{z^4 t}{16}, \frac{z^4 t}{16}\right)$$

$$\widehat{\mathbb{K}_b}(z,t) = \begin{pmatrix} 0 & \frac{z^2 t}{4} & \frac{z^2 t}{4} & \frac{z^3 t}{8} \\ \frac{z^3 t}{8} + \frac{z^2}{4} & \frac{z^3 t}{8} & \frac{z^3 t}{8} & 0 \\ 0 & 0 & 0 & \frac{z^3 t}{8} + \frac{z^2}{4} \\ 0 & \frac{z^2 t}{4} & \frac{z^2 t}{4} & \frac{z^3 t}{8} \end{pmatrix}$$

The extension $\mathtt{AC} \in \widehat{\mathcal{K}}_{\mathtt{ACAA},\mathtt{AAAC}}$ as in $\mathtt{ACA}\underline{\mathtt{A}}|\mathtt{AC}$ leads to *no new* putative-hit position, in contrast to $\mathtt{ACA}\underline{\mathtt{A}}|\underline{\mathtt{A}}\mathtt{AC}$

$$\widehat{\mathcal{K}}_{\mathtt{ACAA},\mathtt{AAAC}}(z,t) = \frac{z^3 t}{8} + \frac{z^2}{4}$$

Generating function of the number of putative-hit positions

- $v_i(z,t) = \nu(v_i) t z^{|v_i|}$ for each $v_i \in d(b)$.
- for each $\widehat{\mathcal{K}}_{ij}$, we can compute by string matching the number of putative-hit positions in each word of $v_i.\widehat{\mathcal{K}}_{ij}$.

$$\widehat{\mathcal{K}}_{ij}(z,t) = \sum_{w \in \widehat{\mathcal{K}}_{ij}} \nu(w) t^{\text{put-hit-pos}(v_i.w)-1} z^{|w|},$$

Generating function of the number of putative-hit positions

- $v_i(z,t) = \nu(v_i) t z^{|v_i|}$ for each $v_i \in d(b)$.
- for each $\widehat{\mathcal{K}}_{ij}$, we can compute by string matching the number of putative-hit positions in each word of $v_i.\widehat{\mathcal{K}}_{ij}$.

$$\widehat{\mathcal{K}}_{ij}(z,t) = \sum_{w \in \widehat{\mathcal{K}}_{ij}} \nu(w) t^{\text{put-hit-pos}(v_i.w)-1} z^{|w|},$$

$$\widehat{\mathbb{K}}(z,t) = \left(\widehat{\mathcal{K}}_{ij}(z,t)\right), \quad \widehat{\mathbb{S}}(z,t) = \left(\mathbb{I} - \widehat{\mathbb{K}}(z,t)\right)^{-1},$$
$$\widehat{\mathbb{G}}(z,t) = \left(v_i(z,t)\widehat{\mathbb{S}}_{ij}(z,t)\right).$$

Generating function of the number of putative-hit positions

- $v_i(z,t) = \nu(v_i) t z^{|v_i|}$ for each $v_i \in d(b)$.
- for each $\widehat{\mathcal{K}}_{ij}$, we can compute by string matching the number of putative-hit positions in each word of $v_i.\widehat{\mathcal{K}}_{ij}$.

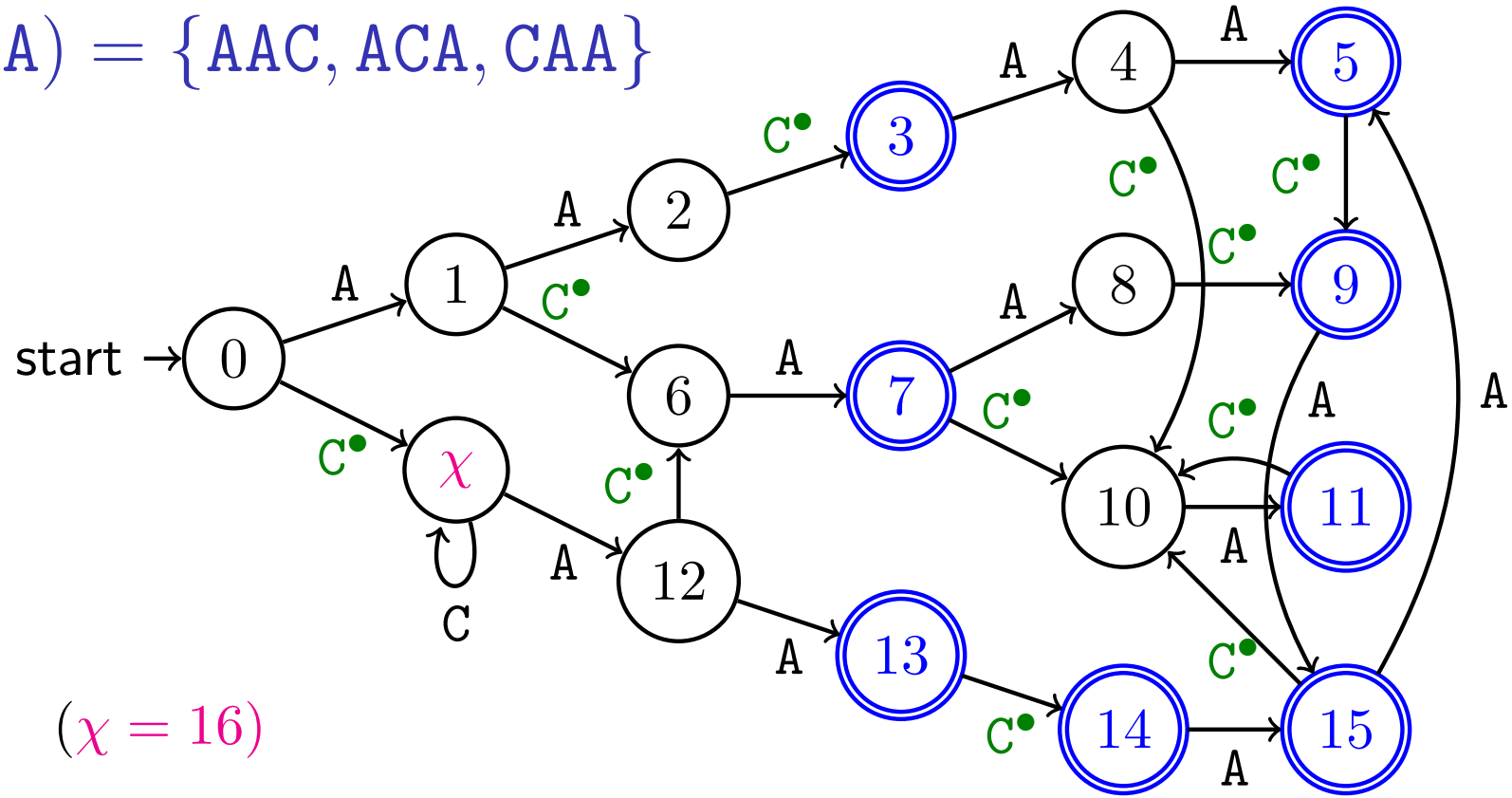
$$\widehat{\mathcal{K}}_{ij}(z,t) = \sum_{w \in \widehat{\mathcal{K}}_{ij}} \nu(w) t^{\text{put-hit-pos}(v_i.w)-1} z^{|w|},$$

$$\widehat{\mathbb{K}}(z,t) = \Big(\widehat{\mathcal{K}}_{ij}(z,t)\Big), \quad \widehat{\mathbb{S}}(z,t) = \Big(\mathbb{I} - \widehat{\mathbb{K}}(z,t)\Big)^{-1},$$
$$\widehat{\mathbb{G}}(z,t) = \Big(v_i(z,t)\widehat{\mathbb{S}}_{ij}(z,t)\Big).$$

$$F_b(z,t) = \widehat{\mathcal{A}_b^\star}(z,t)$$
$$= \widehat{\mathcal{N}}(z) + (\widehat{\mathcal{R}}_1 v_1^-(z), \ldots, \widehat{\mathcal{R}}_r v_r^-(z))\ \widehat{\mathbb{G}}(z,t)\Big((\widehat{\mathbb{M}} - \widehat{\mathbb{K}})^-(z)\widehat{\mathbb{G}}(z,t)\Big)^\star \begin{pmatrix} \widehat{\mathcal{U}}_1(z) \\ \vdots \\ \widehat{\mathcal{U}}_r(z) \end{pmatrix}$$

B - Automaton approach

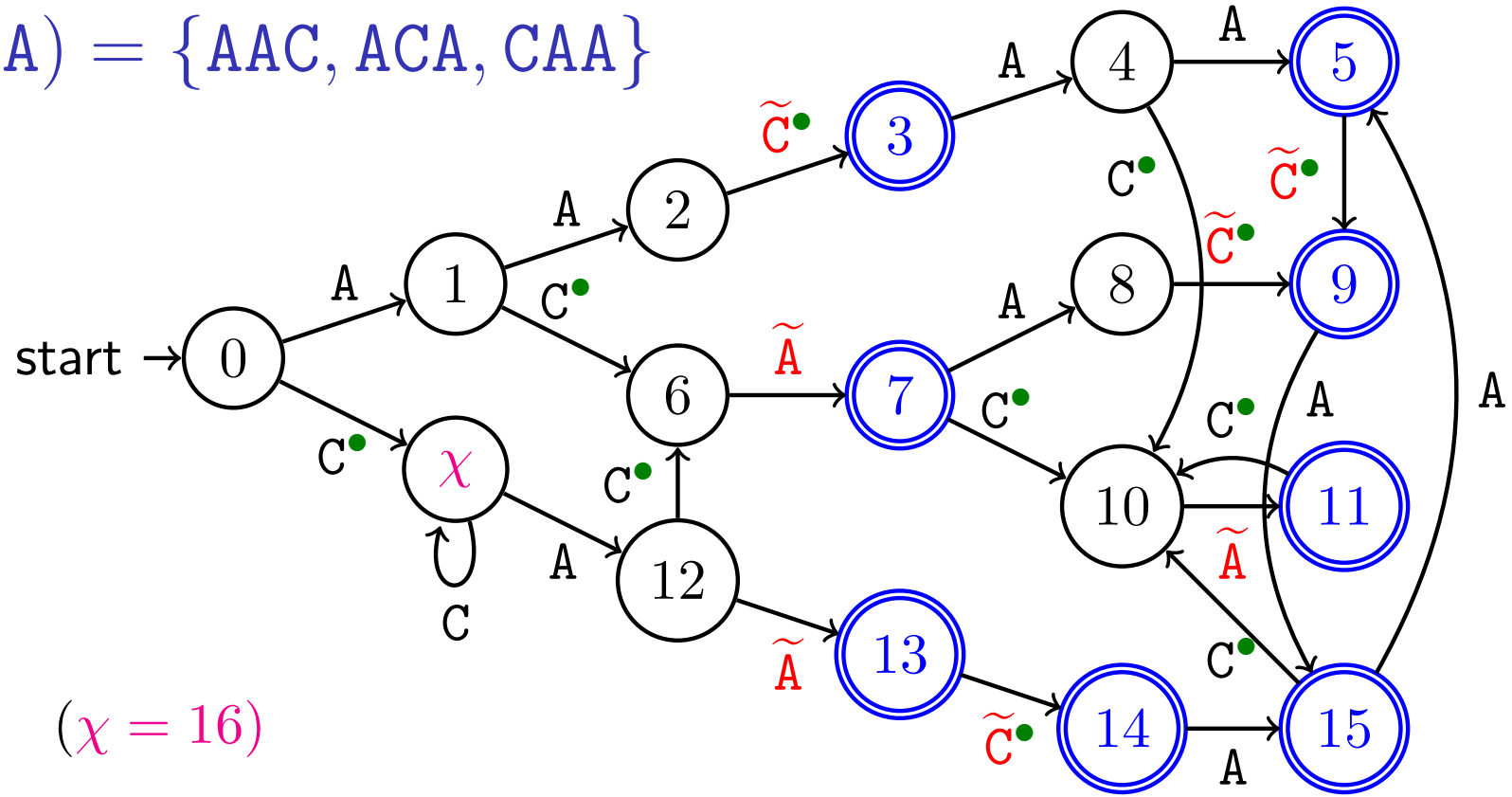
Automaton for constrained clumps of $d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



$(\chi = 16)$

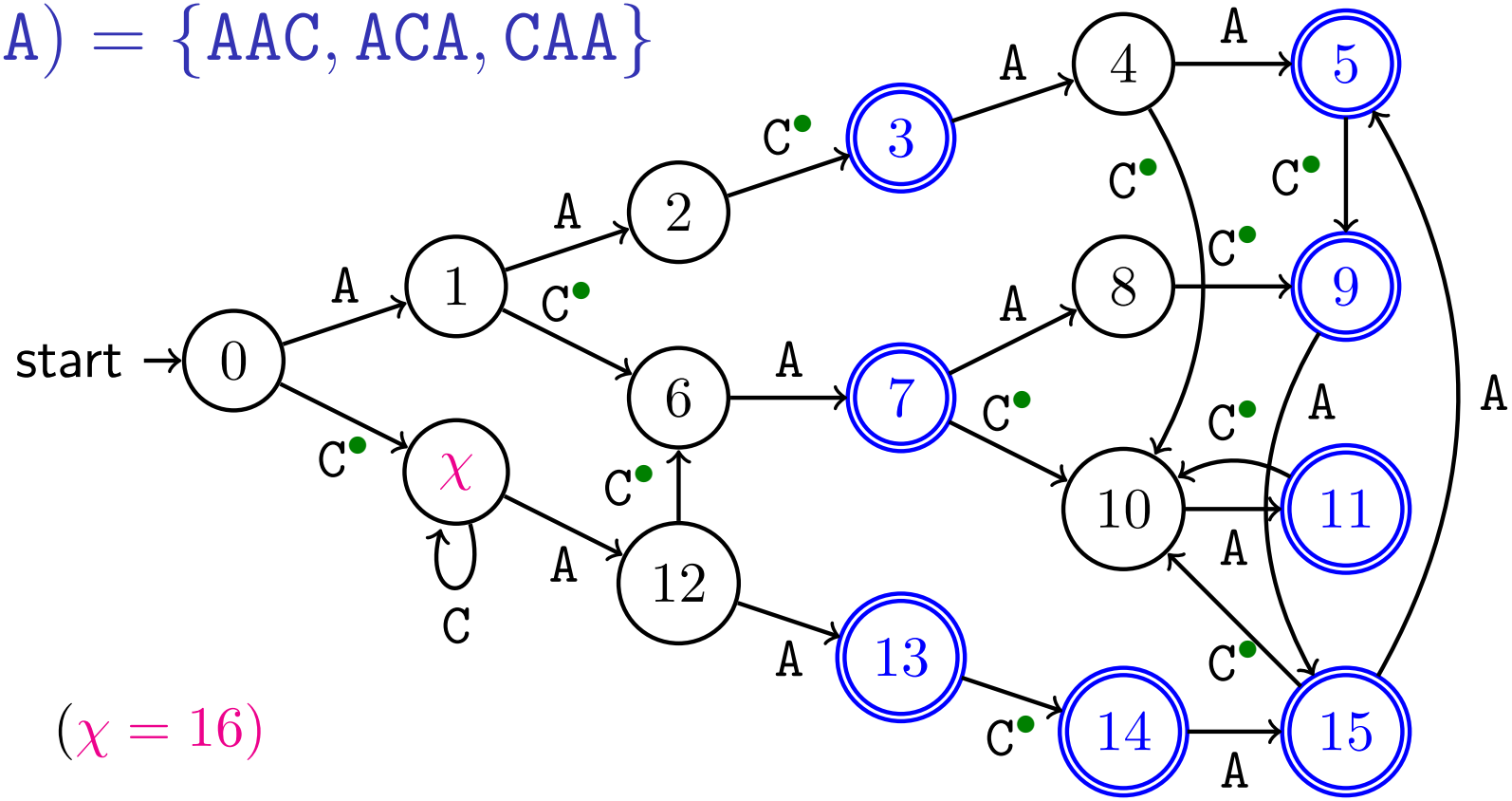
- **Double circles** signal an occurrence of a word of $d(aaa)$.
- **Avoiding** AAA leads to **missing transitions** A
- The **missing transitions** C **point to** the **state** χ.
- • **characters** mark **putative-hit-positions**

Automaton for constrained clumps of $d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



- **Double circles** signal an occurrence of a word of $d(aaa)$.
- **Avoiding** AAA leads to **missing transitions** A
- The **missing transitions** C **point to** the **state** χ.
- • **characters** mark **putative-hit-positions**
- Transitions covered by tildes ($\widetilde{\mathtt{A}}, \widetilde{\mathtt{C}}$) emit a signal counting a putative-hit position.

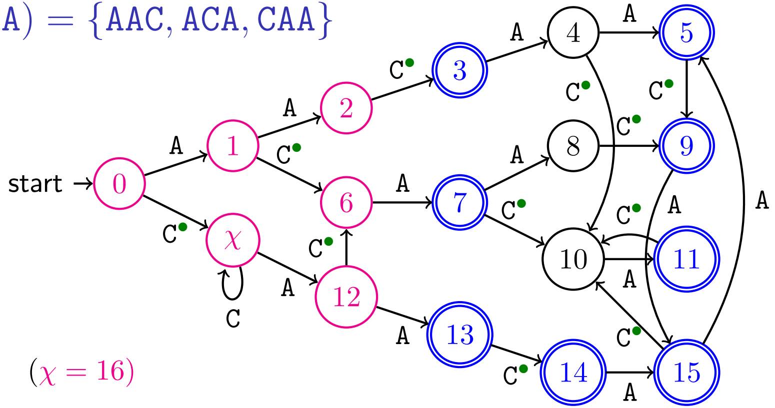
Automaton for constrained clumps of $d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



$(\chi = 16)$

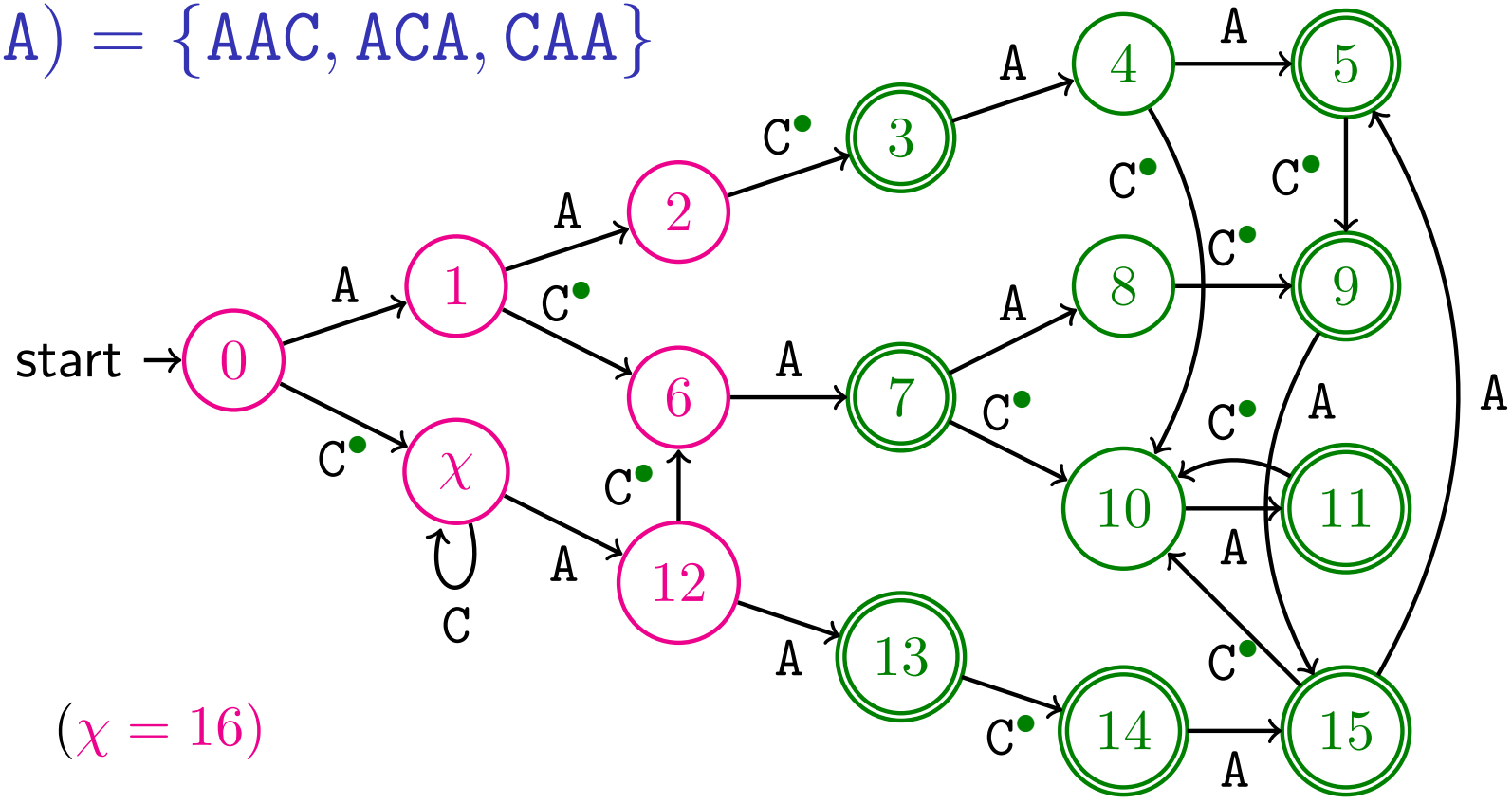
- $O = \{q, \quad \delta(0, w) = q, \ w \in X\}$, (**occurrence** of a word of $d(aaa)$).

Automaton for constrained clumps of $d(\texttt{AAA}) = \{\texttt{AAC}, \texttt{ACA}, \texttt{CAA}\}$



- $\overline{E} = \{q, \quad \delta(0, w) = q,\ w \in \widehat{\mathrm{Pref}}(d(b))\}$, with $\widehat{\mathrm{Pref}}(d(b))$ set of **strict prefixes** of words of $d(b)$.

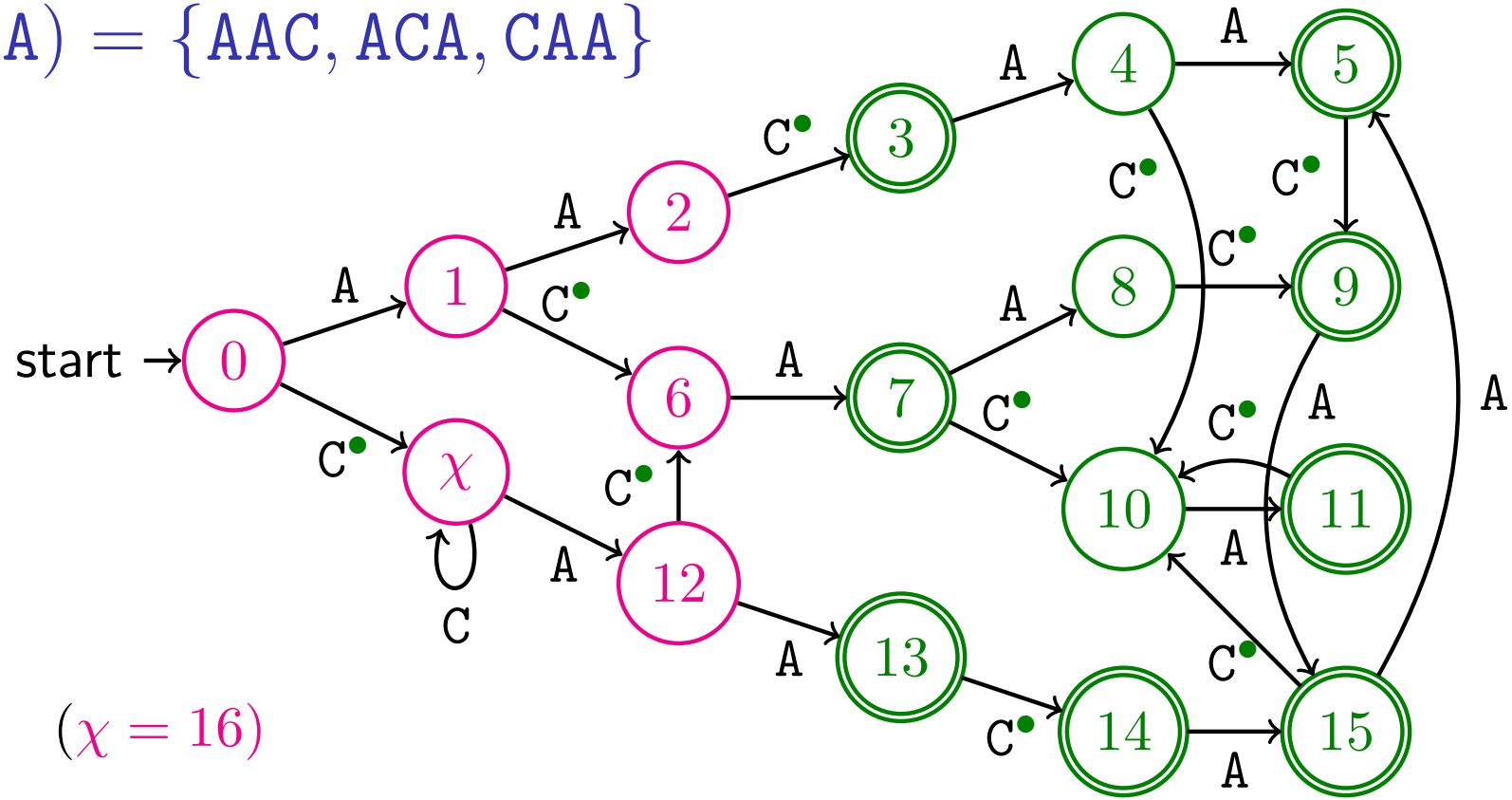
Automaton for constrained clumps of $d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



$(\chi = 16)$

- $\overline{E} = \{q, \quad \delta(0, w) = q, \ w \in \widehat{\mathrm{Pref}}(d(b))\}$, with $\widehat{\mathrm{Pref}}(d(b))$ set of **strict prefixes** of words of $d(b)$.
- **Clump-Core** of the automaton $E = Q \setminus \overline{E}$

Automaton for constrained clumps of $d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$

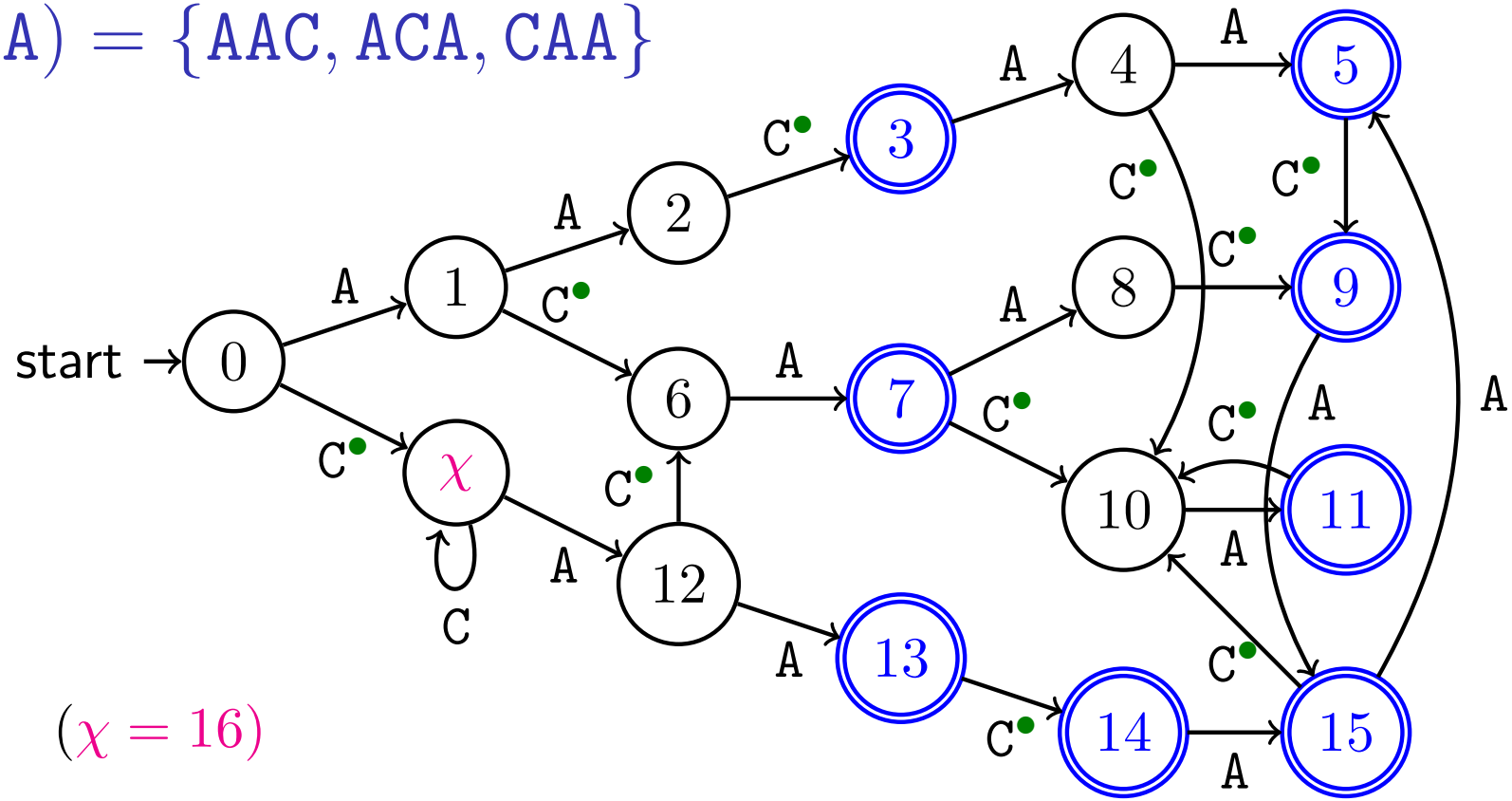


$(\chi = 16)$

- $\overline{E} = \{q, \quad \delta(0, w) = q,\ w \in \widehat{\mathrm{Pref}}(d(b))\}$, with $\widehat{\mathrm{Pref}}(d(b))$ set of **strict prefixes** of words of $d(b)$.
- **Clump-Core** of the automaton $E = Q \setminus \overline{E}$
- **Markov property:** $\forall q \in E, \qquad |\{w \in \mathcal{A}^{|b|}; \delta(x, w) = q\}| = 1$

Definition of an auxiliary function θ

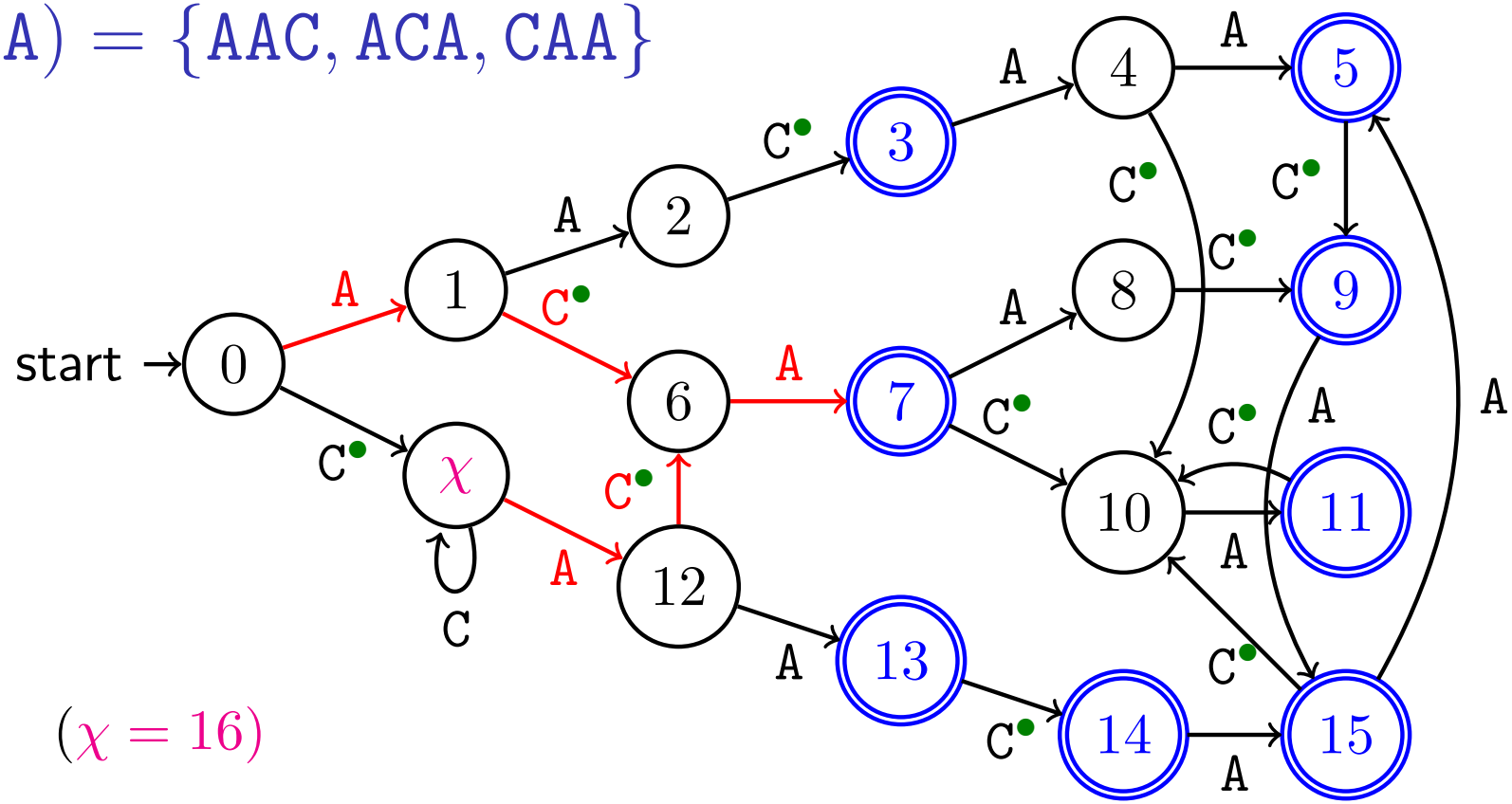
$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



$(\chi = 16)$

Definition of an auxiliary function θ

$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$

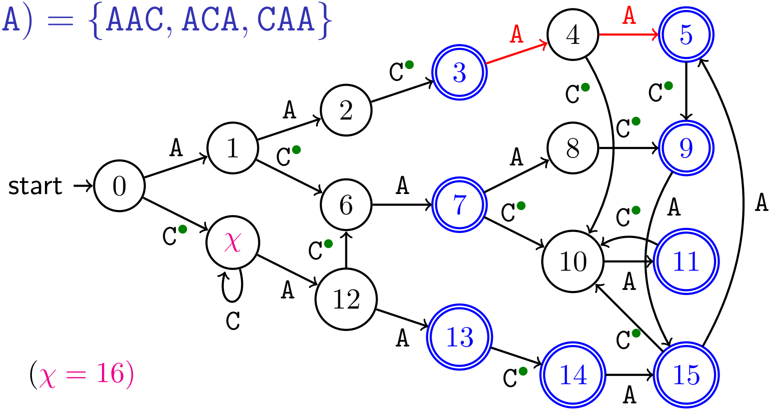


$(\chi = 16)$

$\theta(7) = \mathtt{ACA}$

Definition of an auxiliary function θ

$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$

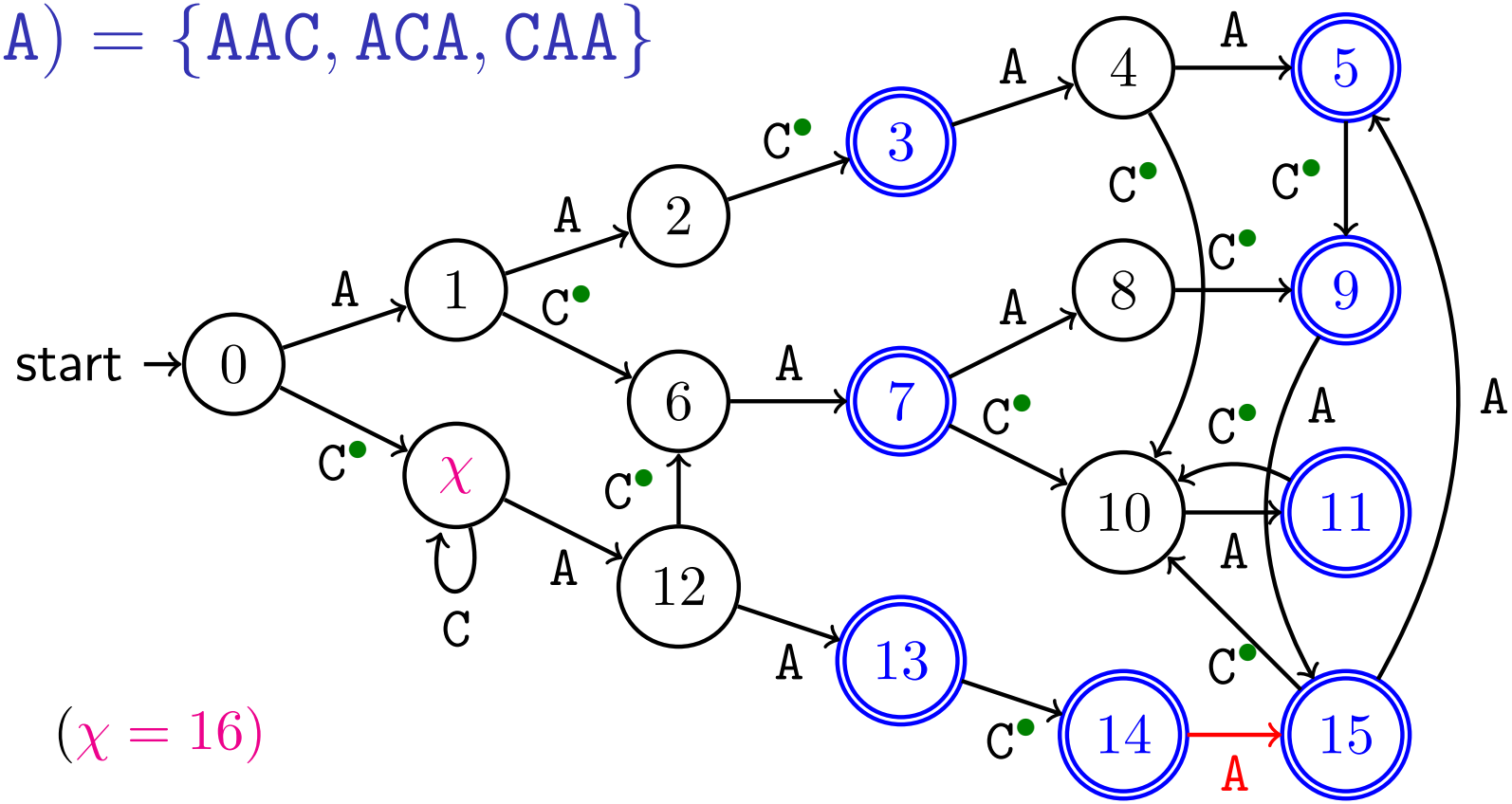


$(\chi = 16)$

$\theta(7) = \mathtt{ACA}$, $\theta(5) = \mathtt{AA}$

Definition of an auxiliary function θ

$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



$(\chi = 16)$

$$\theta(7) = \mathtt{ACA}\ , \quad \theta(5) = \mathtt{AA}\ , \quad \theta(15) = \mathtt{A}$$

Formal definition of θ

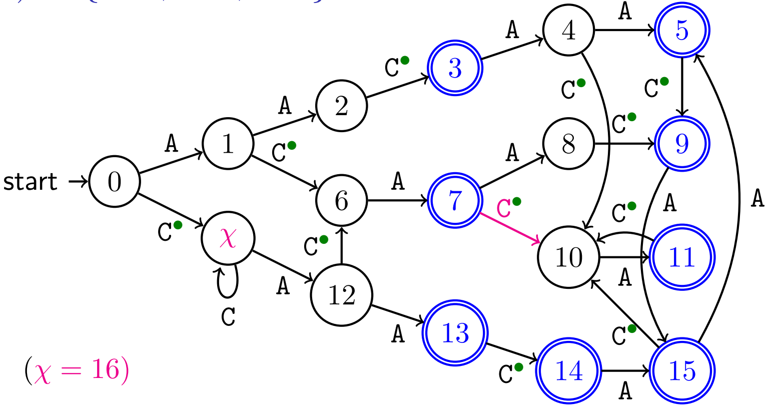
For each state $o \in O$ (recognizing an occurrence of $d(b)$),

$$\theta(o) = \left\{ \begin{array}{l} w \text{ with } |w| \leq |b|, \text{ of } \textbf{maximal length}, \\ \text{verifying} \left| \begin{array}{l} (a) \text{ there exists } q \text{ such that } \delta(q, w) = o, \\ (b) \textbf{ there is no } u \in \widehat{\mathrm{Pref}}(w) \\ \qquad \text{such that } \delta(q, u) \in O \end{array} \right. \end{array} \right.$$

By the **Markov** property, $\theta(o)$ defines a **unique word**

Adjacency matrix $\mathbb{H}(t) = (h_{ij}(t))$

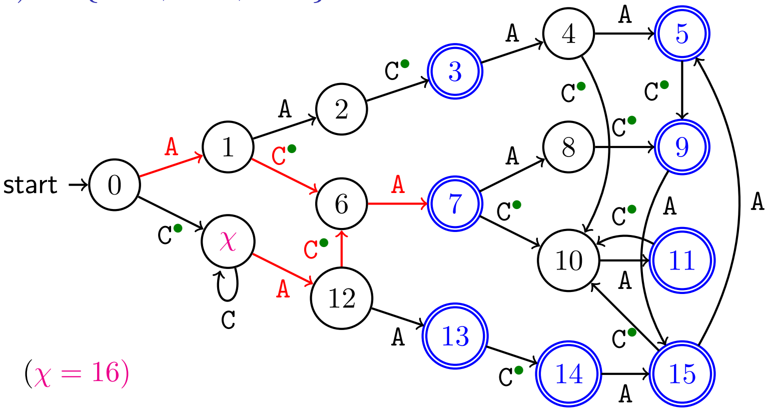
$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



- $h_{7,10} = \nu_{\mathtt{C}} z$

Adjacency matrix $\mathbb{H}(t) = (h_{ij}(t))$

$d(\texttt{AAA}) = \{\texttt{AAC}, \texttt{ACA}, \texttt{CAA}\}$

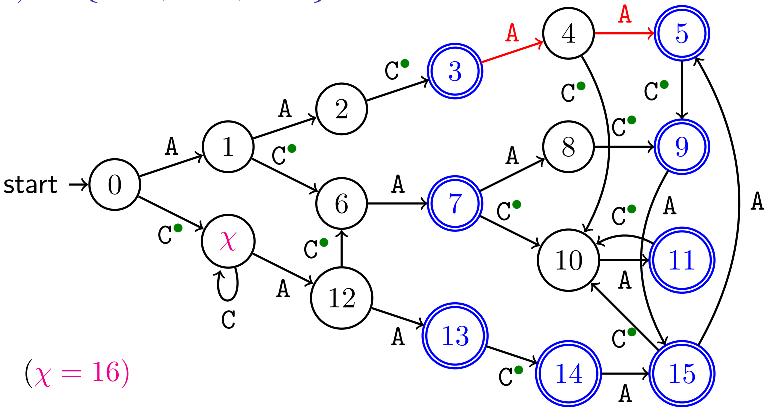


$(\chi = 16)$

- $h_{7,10} = \nu_{\texttt{C}} z$
- $h_{6,7}(t) = \nu_{\texttt{A}} t_{\texttt{C} \to \texttt{A}} z$

Adjacency matrix $\mathbb{H}(t) = (h_{ij}(t))$

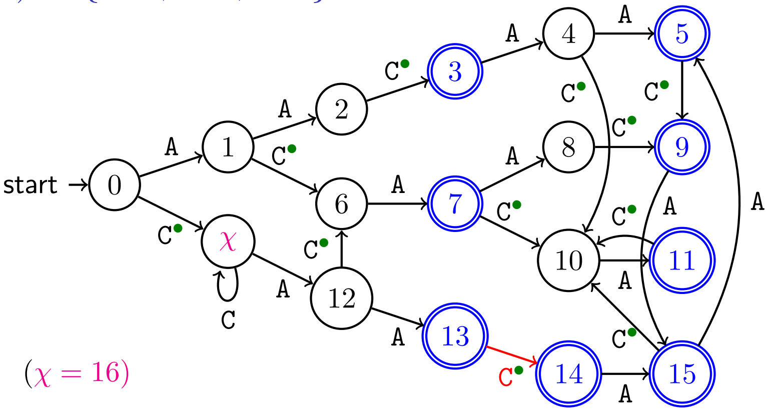
$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



- $h_{7,10} = \nu_{\mathtt{C}} z$
- $h_{6,7}(t) = \nu_{\mathtt{A}} t_{\mathtt{C}\to\mathtt{A}} z$
- $h_{3,4}(t) = h_{45}(t) = \nu_{\mathtt{A}} z$

Adjacency matrix $\mathbb{H}(t) = (h_{ij}(t))$

$d(\mathtt{AAA}) = \{\mathtt{AAC}, \mathtt{ACA}, \mathtt{CAA}\}$



- $h_{7,10} = \nu_{\mathtt{C}} z$
- $h_{6,7}(t) = \nu_{\mathtt{A}} t_{\mathtt{C} \to \mathtt{A}} z$
- $h_{3,4}(t) = h_{45}(t) = \nu_{\mathtt{A}} z$
- $h_{13,14}(t) = \nu_{\mathtt{C}} t_{\mathtt{C} \to \mathtt{A}} z$

Formal definition of the adjacency matrix $\mathbb{H}(t)$

(a) $h_{ij}(t) = 0$ if there is no transition from i to j

(b) With $\delta(i, \alpha) = j$,

$$h_{i,j}(t) = \begin{cases} \nu(\alpha) \text{ if } \begin{vmatrix} j \notin O, \\ j \in O \text{ and } \theta(j) \text{ contains } \textbf{no putative-hit position} \end{vmatrix} \\ \nu(\alpha) \times t \text{ elsewhere} \end{cases}$$

Formal definition of the adjacency matrix $\mathbb{H}(t)$

(a) $h_{ij}(t) = 0$ if there is no transition from i to j

(b) With $\delta(i, \alpha) = j$,

$$h_{i,j}(t) = \begin{cases} \nu(\alpha) \text{ if } \begin{vmatrix} j \notin O, \\ j \in O \text{ and } \theta(j) \text{ contains } \textbf{no putative-hit position} \end{vmatrix} \\ \nu(\alpha) \times t \text{ elsewhere} \end{cases}$$

From matrix to generating function

$$\begin{aligned} F_b(z, t) &= (1, 0, \ldots, 0) \times \left(\mathbb{I} + z\mathbb{H}(t) + \cdots + z^n \mathbb{H}^n(t) + \ldots \right) \times \mathbf{1}^t \\ &= (1, 0, \ldots, 0) \times \left(\mathbb{I} - z\mathbb{H}(t) \right)^{-1} \times \mathbf{1}^t. \end{aligned}$$

Rational functions and gfun

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

Rational functions, Taylor coefficient of order n and `gfun`

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \quad \text{and} \quad F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)},$$

$$E(z) = \sum_n \eta_n z^n = \frac{\partial}{\partial t} F_b(z,t)\Big|_{t=1} = \frac{P_t'(z,1)}{Q(z,1)} - \frac{P(z,1)Q_t'(z,1)}{Q^2(z,1)}$$

Rational functions, Taylor coefficient of order n and `gfun`

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \quad \text{and} \quad F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)},$$

$$E(z) = \sum_n \eta_n z^n = \left.\frac{\partial}{\partial t} F_b(z,t)\right|_{t=1} = \frac{P'_t(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_t(z,1)}{Q^2(z,1)}$$

where, $P(z,t)$ and $Q(z,t)$ are **polynoms**, and, in a **random** sequence $S_n(0)$ of length n with **no occurrence** of b,

Rational functions, Taylor coefficient of order n and `gfun`

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \quad \text{and} \quad F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)},$$

$$E(z) = \sum_n \eta_n z^n = \left.\frac{\partial}{\partial t} F_b(z,t)\right|_{t=1} = \frac{P'_t(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_t(z,1)}{Q^2(z,1)}$$

where, $P(z,t)$ and $Q(z,t)$ are **polynoms**, and, in a **random** sequence $S_n(0)$ of length n with **no occurrence** of b,

- $\widehat{f}_n^{(b)} = \mathbf{P}(S_n(0))$

Rational functions, Taylor coefficient of order n and `gfun`

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \quad \text{and} \quad F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)},$$

$$E(z) = \sum_n \eta_n z^n = \left.\frac{\partial}{\partial t} F_b(z,t)\right|_{t=1} = \frac{P'_t(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_t(z,1)}{Q^2(z,1)}$$

where, $P(z,t)$ and $Q(z,t)$ are **polynoms**, and, in a **random** sequence $S_n(0)$ of length n with **no occurrence** of b,

- $\widehat{f}_n^{(b)} = \mathbf{P}(S_n(0))$
- η_n is the **unconditionned probability** of the expectation of the count of putative-hit positions

Rational functions, Taylor coefficient of order n and `gfun`

rational function $\frac{f(z)}{g(z)}$	$\rightarrow$	`gfun[diffeqtorec]`	$\rightarrow$	**recurrence** equations
recurrence equations	$\rightarrow$	`gfun[rectoproc]`	$\rightarrow$	**procedure** Proc(n)

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \quad \text{and} \quad F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)},$$

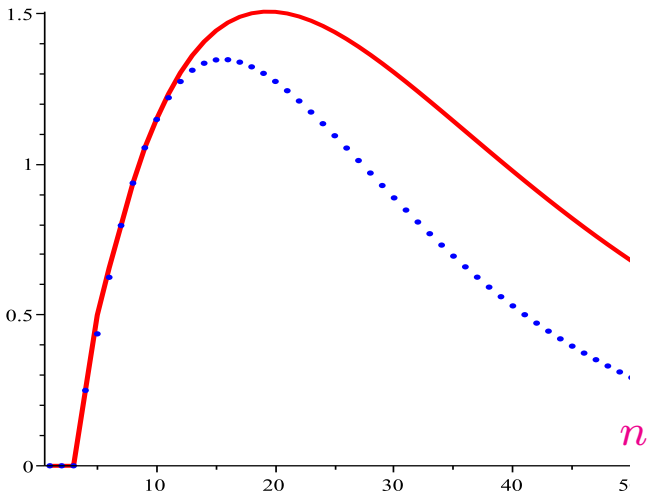
$$E(z) = \sum_n \eta_n z^n = \left.\frac{\partial}{\partial t} F_b(z,t)\right|_{t=1} = \frac{P'_t(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_t(z,1)}{Q^2(z,1)}$$

where, $P(z,t)$ and $Q(z,t)$ are **polynoms**, and, in a **random** sequence $S_n(0)$ of length n with **no occurrence** of b,

- $\widehat{f}_n^{(b)} = \mathbf{P}(S_n(0))$
- η_n is the **unconditionned probability** of the expectation of the count of putative-hit positions
- **Conditionned expectation:** $\widetilde{\eta}_n = \eta_n / \widehat{f}_n^{(b)}$

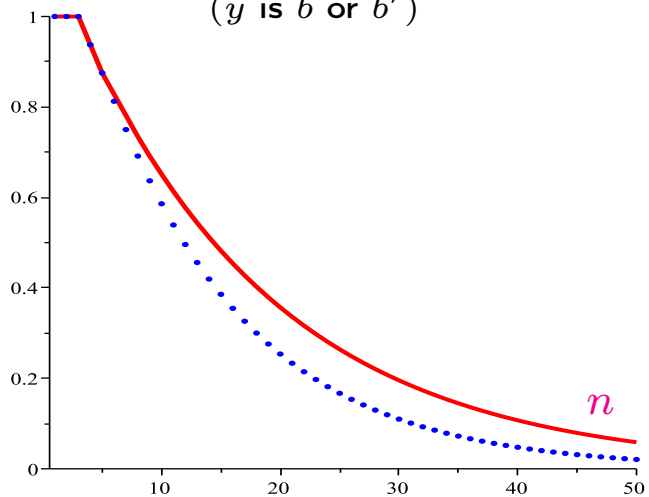
An unexpected behaviour

$\eta_n = \mathbf{E}(H_n^{(\mathtt{A}\to\mathtt{C})}) + \mathbf{E}(H_n^{(\mathtt{C}\to\mathtt{A})})$



$\widehat{f}_n^{(y)} = \mathbf{P}(|S_n(0)|_y = 0)$

(y is b or b')



$b = \mathtt{ACAC} \qquad b' = \mathtt{AACC}$

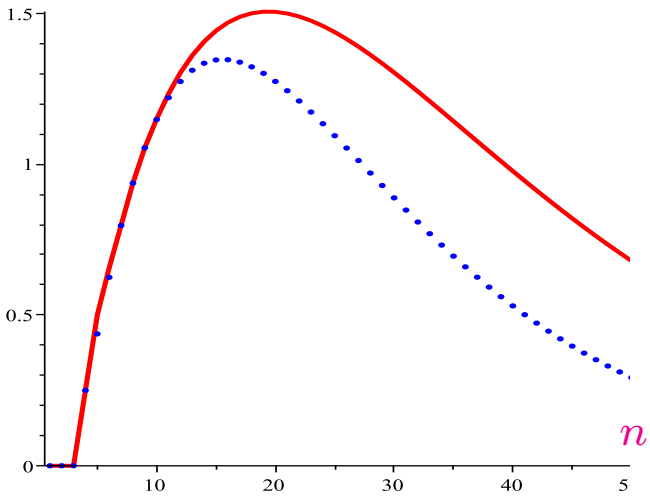
$\nu(\mathtt{A}) = \nu(\mathtt{C}) = \dfrac{1}{2}$

$\pi_{\mathtt{A}\to\mathtt{C}} = \pi_{\mathtt{C}\to\mathtt{A}}$

$$\mathbf{E}(H) = \left.\frac{\partial F_b(z,t)}{\partial t}\right|_{t=1}$$

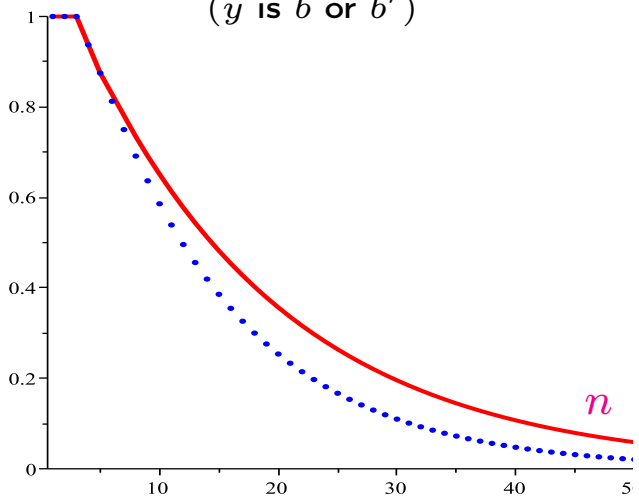
An unexpected behaviour

$\eta_n = \mathbf{E}(H_n^{(\mathtt{A}\to\mathtt{C})}) + \mathbf{E}(H_n^{(\mathtt{C}\to\mathtt{A})})$

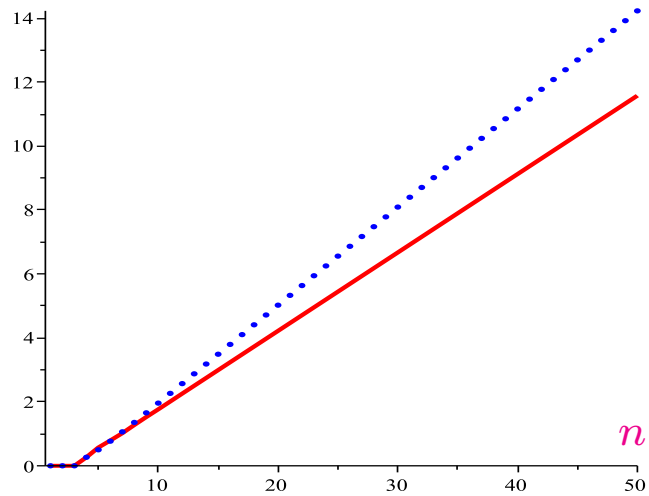


$\widehat{f}_n^{(y)} = \mathbf{P}(|S_n(0)|_y = 0)$

(y is b or b')



$\widetilde{\eta}_n = \eta_n \Big/ \widehat{f}_n^{(y)}$



$b = \mathtt{ACAC} \qquad b' = \mathtt{AACC}$

$\nu(\mathtt{A}) = \nu(\mathtt{C}) = \dfrac{1}{2}$

$\pi_{\mathtt{A}\to\mathtt{C}} = \pi_{\mathtt{C}\to\mathtt{A}}$

$$\mathbf{E}(H) = \left.\frac{\partial F_b(z,t)}{\partial t}\right|_{t=1}$$

A proof by singularity analysis

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \qquad P(z,t) \text{ and } Q(z,t) \text{ polynomials}$$

$$F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)}$$

$\widehat{f}_n^{(b)}$ probability that $S_n(0)$ has **no occurrence** of b.

$$E(z) = \sum_{n\geq 0} \mathbf{E}(H_n) z^n = \frac{P'_x(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_x(z,1)}{Q^2(z,1)}$$

A proof by singularity analysis

$$F_b(z,t) = \frac{P(z,t)}{Q(z,t)} \qquad P(z,t) \text{ and } Q(z,t) \text{ polynomials}$$

$$F_b(z,1) = \sum_{n\geq 0} \widehat{f}_n^{(b)} z^n = \frac{P(z,1)}{Q(z,1)}$$

$\widehat{f}_n^{(b)}$ probability that $S_n(0)$ has **no occurrence** of b.

$$E(z) = \sum_{n\geq 0} \mathbf{E}(H_n) z^n = \frac{P'_x(z,1)}{Q(z,1)} - \frac{P(z,1)Q'_x(z,1)}{Q^2(z,1)}$$

The **dominant singularity** τ is the smallest positive solution of $Q(z,1) = 0$. Use suitable Cauchy integrals

$$\begin{cases} \widehat{f}_n^{(b)} = \psi \times \tau^{-(n-1)} \left(1 + \mathcal{O}\left(B^n\right)\right), \qquad (B<1) \\ \mathbf{E}(H_n) = [z^n]E(z) = \tau^{-n}(\phi_1 \times n + \phi_2) \times \left(1 + \mathcal{O}\left(B^n\right)\right) \end{cases}$$

$$\Longrightarrow \mathbf{E}(\widetilde{H}_n) = \frac{\mathbf{E}(H_n)}{\widehat{f}_n^{(b)}} = (c_1 \times n + c_2) \times \left(1 + \mathcal{O}\left(B^n\right)\right), \ (B<1).$$

General case

Compute $F_b(z, t_{\mathtt{A}\to\mathtt{C}}, t_{\mathtt{A}\to\mathtt{G}}, t_{\mathtt{A}\to\mathtt{T}}, t_{\mathtt{C}\to\mathtt{A}}, \dots, t_{\mathtt{T}\to\mathtt{C}}, t_{\mathtt{T}\to\mathtt{G}})$

$$\widehat{f}_n^{(b)} = [z^n] F_b(z, 1, 1, \dots, 1, 1)$$

$$\mathfrak{P}_n \approx [z^n] \sum_{\alpha\neq\beta\in\{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} \left.\frac{\partial F_b(z, 1, \dots, 1, \pi_{\alpha\to\beta} t_{\alpha\to\beta}, 1, \dots)}{\partial t_{\alpha\to\beta}}\right|_{t_{\alpha\to\beta}=1} \Big/ \widehat{f}_n^{(b)}$$

- The **dominant singularities** of **all the terms of the sum** are **equal** to the **dominant singularity** of $F_b(z, 1, 1, \dots, 1, 1)$
- $\mathfrak{P}_n$ behaves **quasi-linearly**

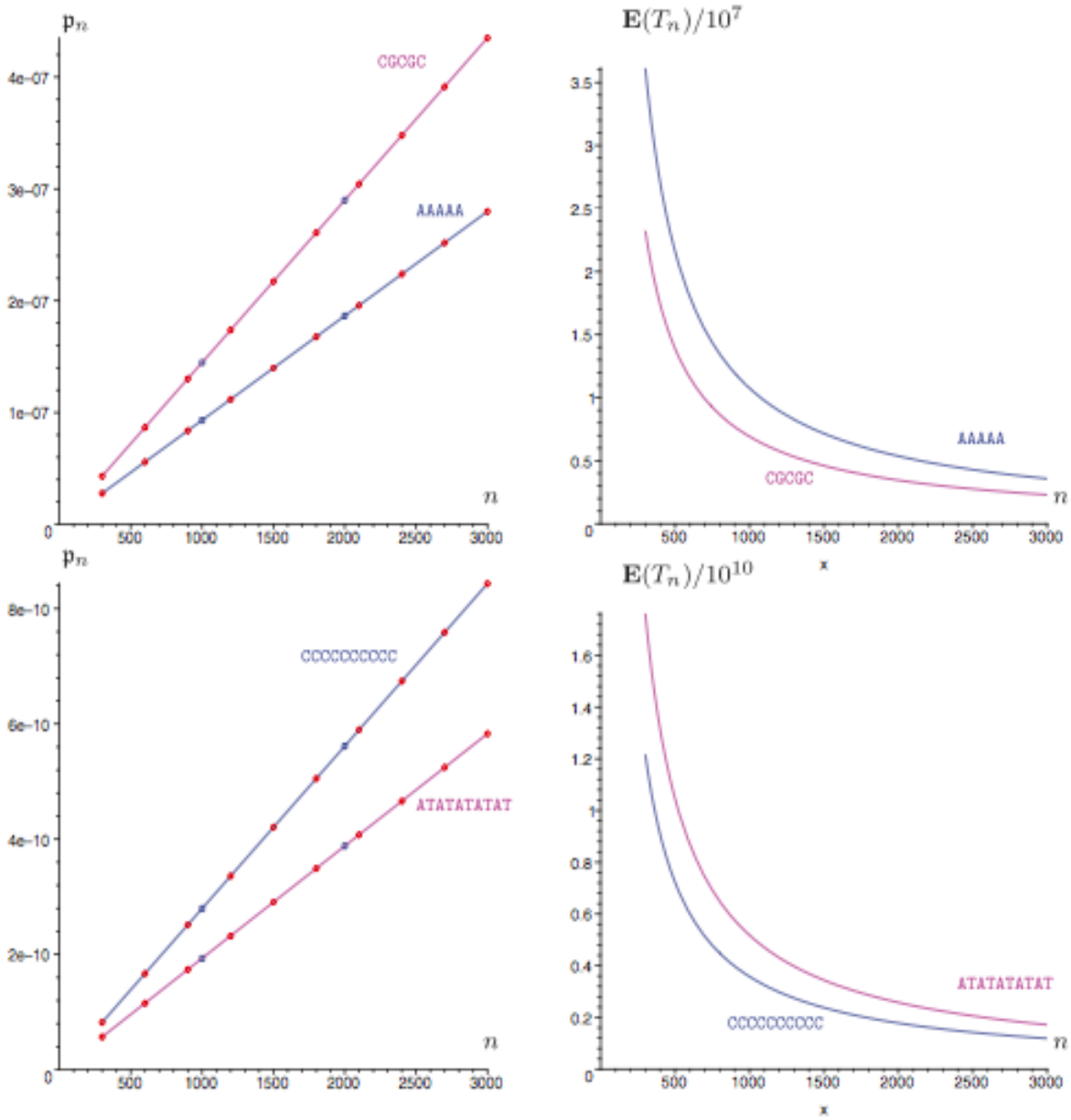


FIG. 5. Plots of the probability p_n **(left)** and of the expected waiting time $E(T_n)$ **(right)**. **(Top)** b = AAAAA (blue) and b' = CGCGC (magenta). **(Botton)** b = CCCCCC CCCC (blue) and b' = ATATATA TAT (magenta). In the linear plots of the probability, the anchors values for $n = 1000$ and $n = 2000$ (computed by automata) are represented by boxes; the straight lines are the straight lines going through the corresponding points and the circles are test values also computed by automata. The fit is perfect as expected from singularity analysis.

(from Behrens-Nicaud-P.N., JCB 19,5, 2012)