On the compressiveness of the Burrows-Wheeler transform

Hideo Bannai[1], Tomohiro I[2], Yuto Nakashima[3]

[1] Institute of Science Tokyo
[2] Kyushu Institute of Technology
[3] Kyushu University

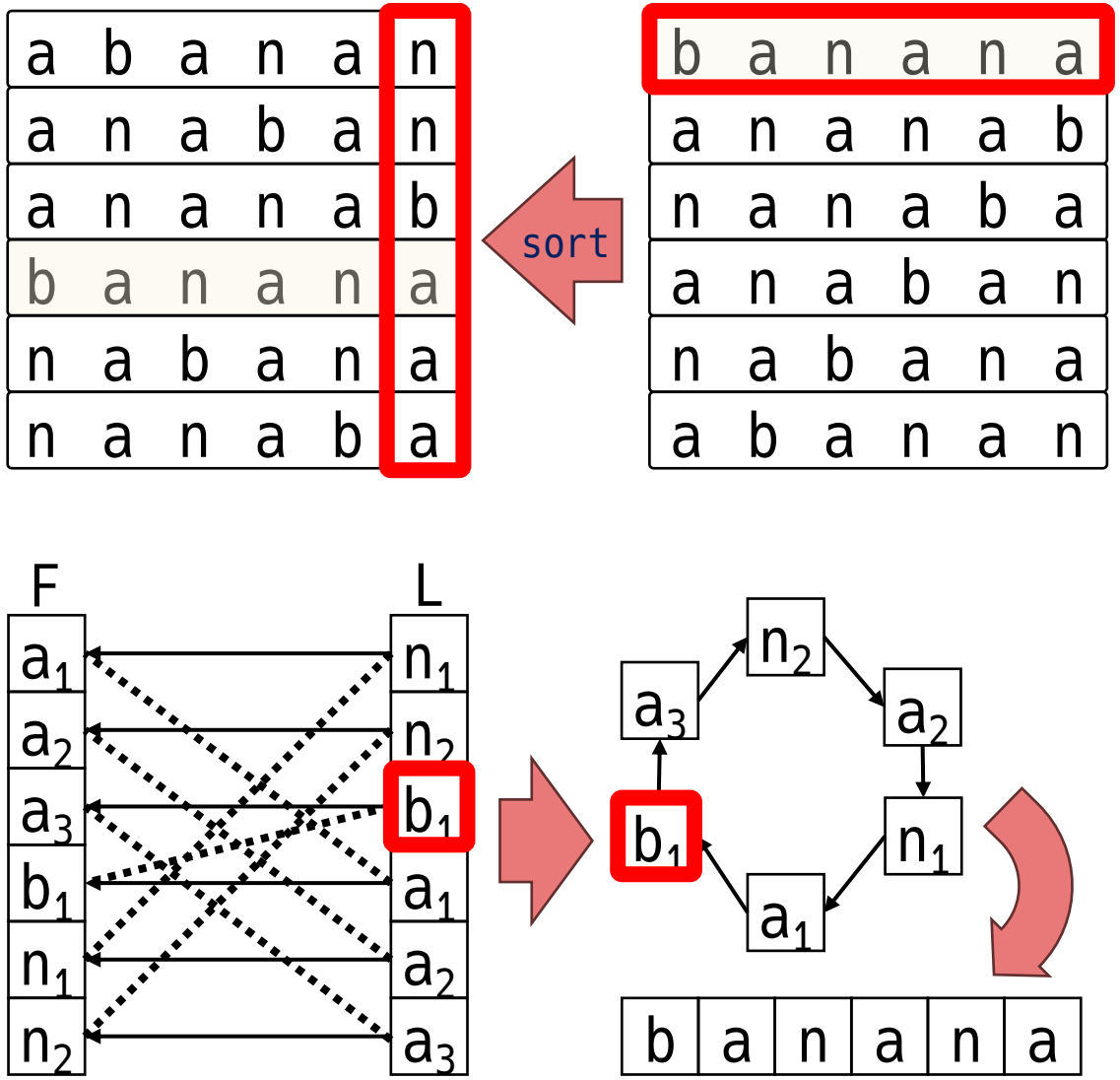


Institute of SCIENCE TOKYO

June 17-19, 2025 CPM 2025 @ Milano

Burrows-Wheeler Transform (BWT) [Burrows&Wheeler 1994]

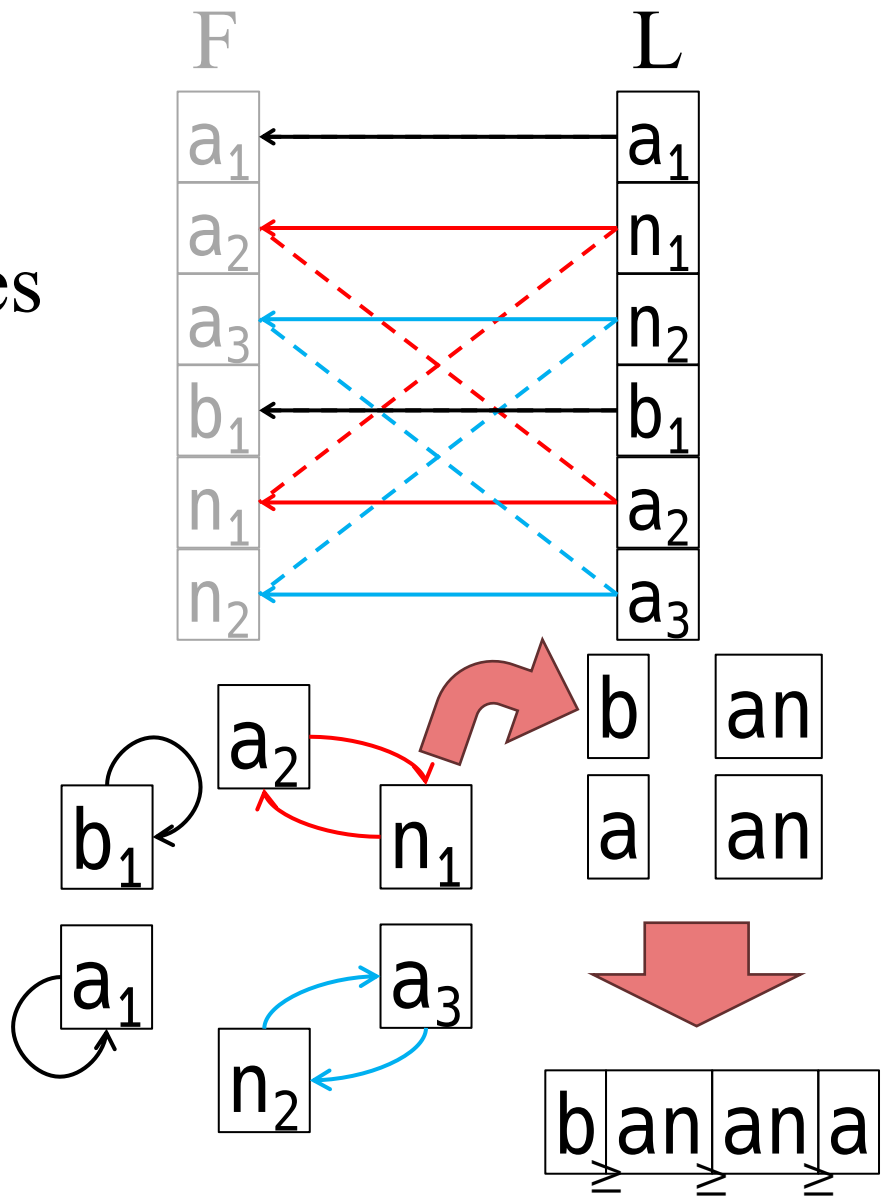
- A "reversible" transform
  - The last symbol of all rotations in lex. order of the rotations
    - BWT(banana) = nnbaaa
  - The inverse transform: via LF-mapping induced by a stable sorting of the symbols
  - BWT is actually a mapping from a *cyclic string* to a string, so we need more information (e.g. the starting position or use end-of-string marker $) to get the original string.
    - $BWT^{-1}(nnbaaa, 3) = banana$



Bijective BWT (BBWT) [Scott 2007]

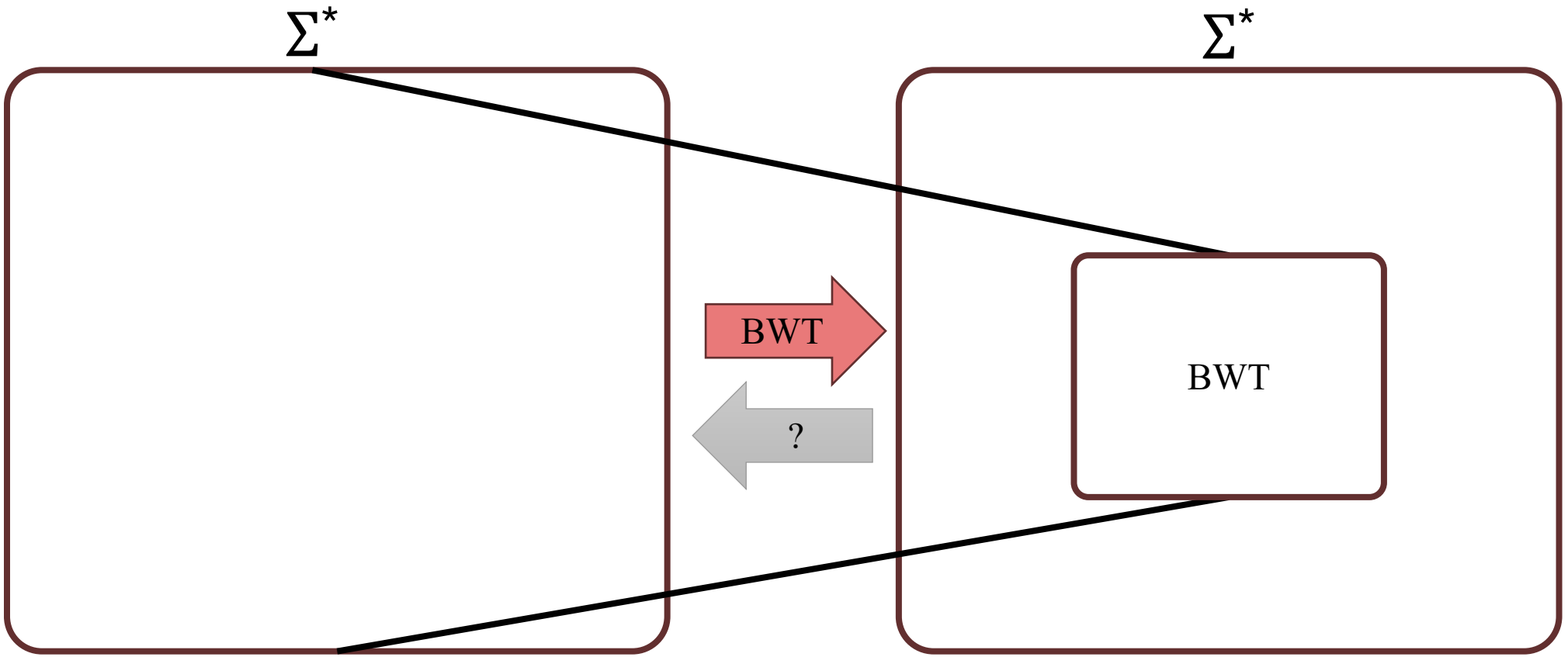
- BWT is **not** bijective on set of all strings
  - even with starting pos or $
- Inverse transform can lead to multiple cycles
  - actually, a bijection between a **string** and a **multiset of cyclic primitive strings**
    [Gessel&Reutenauer 1993][Mantaci et al. 2007]
- We can make it a bijection on strings by
  - Taking lex. smallest rotation of each cycle (⇔ Lyndon words [Lyndon 1954])
  - Concatenating in non-increasing lex order (⇔ Lyndon factorization [Chen-Fox-Lyndon 1958])

BBWT = eBWT of Lyndon factorization



Bijective BWT is a generalization of BWT

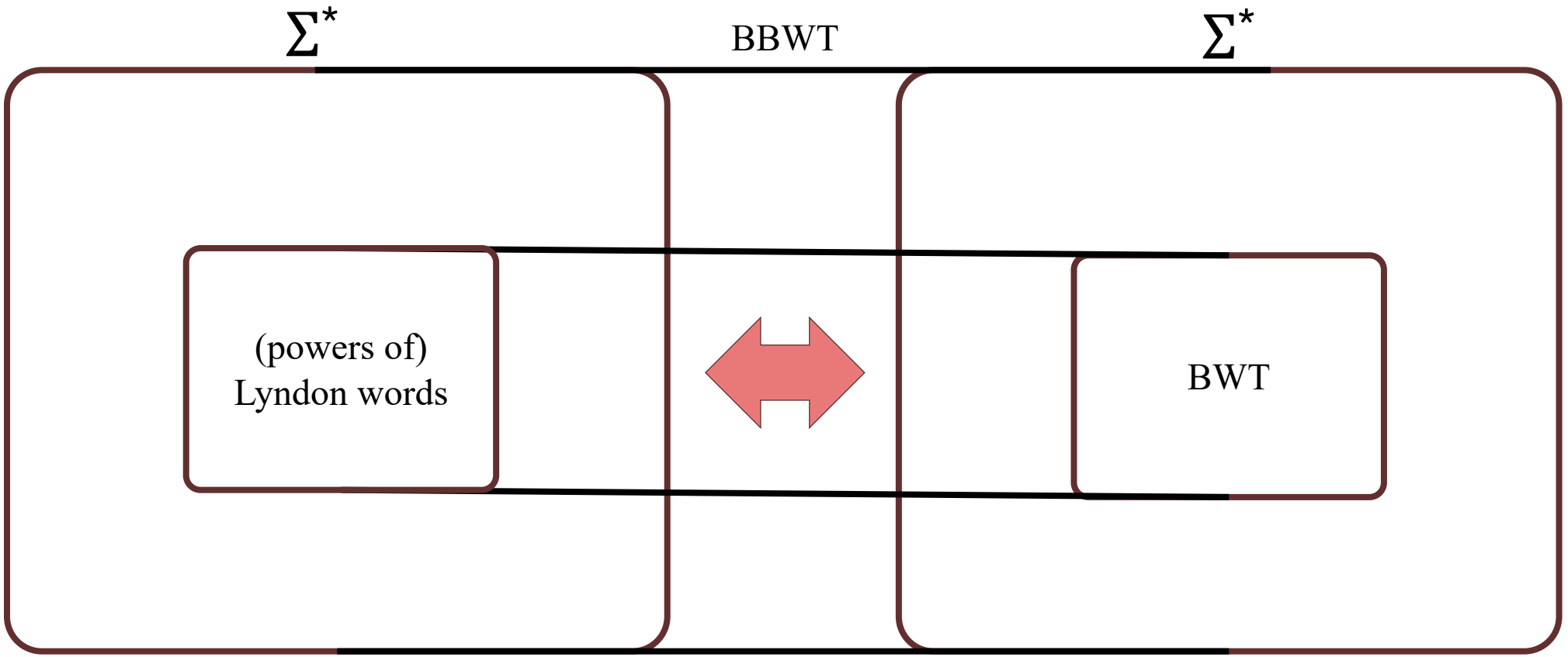
Although BWT is not a bijection when the domain is Σ^*…



Bijective BWT is a generalization of BWT

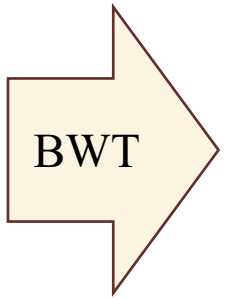
It is a bijection when the domain = (powers of) Lyndon words.

➔ BBWT(T) = BWT(T) for any T = w^k for w: Lyndon, integer $k \geq 1$.



The “clustering effect” of BWT

```
On_31_October_2023,_the_Japanese_cabinet_decid
ed_to_amend_the_National_University_Corporatio
n_Act._The_former_National_University_Corporat
ion_Tokyo_Institute_of_Technology_was_renamed_
the_National_University_Corporation_Institute_
of_Science_Tokyo,_and_the_National_University_
Corporation_Tokyo_Medical_and_Dental_Universit
y_was_dissolved_at_the_same_time._After_being_
passed_by_the_National_Diet_of_Japan,_the_amen
dment_took_effect_on_1_October_2024._The_new_u
niversity's_headquarters_is_located_in_Meguro_
ku,_Tokyo,_where_the_former_Tokyo_Institute_of
_Technology_headquarters_was_located.
```




```
yuo3nodet4223___002_2_____________________.__
_____________nrrnn.yyyydlnoofeoneeere11fff..e,
rnnllllloel,drdetskeesydsosseteeetgse,yddd,ete
dtwysy,ceennnnntcsn__p__pJJuuwwwp_ccrrrrNNNNN_
ooa__iooneeeSAeOOeneeeenene_ie_naamhhhhhchshhht
tthrmhhTTdftvstmdM_MbrimmmDbbmmthttvvvvvvnnino
ooofe__AneoottttttTtTt__wccdccDt_betttttttttt_d
ttttsssssnnnnnno_ooooaaaaaaa__oooiaaaadrraoOo
oioooooooeeaeaeai_iuUUUUUhhIIIeeyyyyytrttll___
_lloTTTTTnns_iiiiiiiiitppppffCCCCaa_rrrrddeeee
eooooe_oouoooooeeeeeeeeeaaa'raiar_esrrrrrrsainnn
ceecannuuuaafrr_________aaaaaaaaasss_cc_iiiiii
iiikqq_gtttliiiiie____tttttgbtgkkkkk
```

Text from: https://en.wikipedia.org/wiki/Institute_of_Science_Tokyo

BWT makes the string “easier to compress”

- Intuitive reason: rotations with similar context are brought closer together.
- There is an “asymmetry” in the mapping with respect to RLE:
  - $^{\forall}$T: $r(T) = |RLE(BWT(T))| \leq 2 \cdot |RLE(T)|$ [Mantaci et al. 2017]
  - There exist infinite string families (= Fibonacci words F_k) s.t. $|RLE(F_k)| = \Theta(|F_k|)$ and $r(F_k) = |RLE(BWT(F_k))| = O(1)$ [Mantaci et al. 2003]

Can the clustering effect be quantified with other repetitiveness measures?

Main Results

=|RLE(BBWT(T))|

New results on the compressibility of BWT, BBWT

1. Existence of infinite string families s.t. $r_B(T) = o(r(T))$.
2. For any T, $r(T)/r_B(T)$, $r_B(T)/r(T) = O(\text{polylog } n)$
3. On clustering effects of BWT and BBWT

A) $r_B(T) = \text{RLE}(\text{BBWT}(T)) \leq 2 \cdot \text{RLE}(T)$ (same as BWT)

$= \max_k(\#\{\text{length-}k \text{ substrings of T}\}/k)$
(lower bound for dictionary compression)

B) For any μ s.t. $\delta(T) = O(\mu(T))$ and $\mu(T)=O(\text{RLE}(T))$:
$\mu(\text{BWT}(T)), \mu(\text{BBWT}(T)) = O(\mu(T)\text{polylog}(n))$

C) There exists infinite family of strings s.t.
$\delta(T) = \Omega(n/\log n), \delta(\text{BBWT}(T)) \leq r(\text{BBWT}(T)) = O(1)$

➔BBWT+dictionary compression is **strictly stronger** than dictionary compression

Asymmetry in clustering effect (1)

Theorem

$$\mu(\mathrm{BWT}(\mathrm{T})), \mu(\mathrm{BBWT}(\mathrm{T})) = \mathrm{O}(\mu(\mathrm{T})\mathrm{polylog}(n))$$

for any μ s.t. $\delta(\mathrm{T}) = \mathrm{O}(\mu(\mathrm{T}))$ and $\mu(\mathrm{T}) = \mathrm{O}(\mathrm{RLE}(\mathrm{T}))$.

Proof

[Kempa&Kociumaka 2020, 2022]

$$\mu(\mathrm{BWT}(\mathrm{T})) = \mathrm{O}(\mathrm{RLE}(\mathrm{BWT}(\mathrm{T}))) = \mathrm{O}(\delta(\mathrm{T})\log^2|\mathrm{T}|) = \mathrm{O}(\mu(\mathrm{T})\log^2|\mathrm{T}|).$$

[Badkobeh et al. 2024]

$$\mu(\mathrm{BBWT}(\mathrm{T})) = \mathrm{O}(\mathrm{RLE}(\mathrm{BBWT}(\mathrm{T}))) = \mathrm{O}(z(\mathrm{T})\log^2|\mathrm{T}|)$$

$$= \mathrm{O}(\delta(\mathrm{T})\log^3|\mathrm{T}|) = \mathrm{O}(\mu(\mathrm{T})\log^3|\mathrm{T}|).$$

[Raskhodnikova et al. 2013]

Asymmetry in clustering effect (2)

Theorem

There exists infinite family of strings s.t.

$$\delta(T) = \Omega(n/\log n),\ r(\mathrm{BBWT}(T)) = O(1).$$

Asymmetry in clustering effect (2)

Theorem

There exists infinite family of strings s.t.

$\delta(T) = \Omega(n/\log n)$, $r(\mathrm{BBWT}(T)) = O(1)$.

$\boldsymbol{\delta(\mathrm{BBWT}^{-1}(F_k)) = \Omega(n/\log n)}$, $r(F_k) = O(1)$

Fibonacci numbers

- $f_0 = 1$, $f_1 = 1$, $f_i = f_{i-1} + f_{i-2}$
- $f_2 = 2$
- $f_3 = 3$
- $f_4 = 5$
- $f_5 = 8$

Fibonacci words

- $F_0 = \mathtt{b}$, $F_1 = \mathtt{a}$, $F_i = F_{i-1}F_{i-2}$
- $F_2 = \mathtt{ab}$
- $F_3 = \mathtt{aba}$
- $F_4 = \mathtt{abaab}$
- $F_5 = \mathtt{abaababa}$

$|F_k| = f_k$ length of F_k

$|F_k|_{\mathtt{a}} = f_{k-1}$ # of a's in F_k

$|F_k|_{\mathtt{b}} = f_{k-2}$ # of b's in F_k

Fibonacci numbers/words and Zeckendorf representation

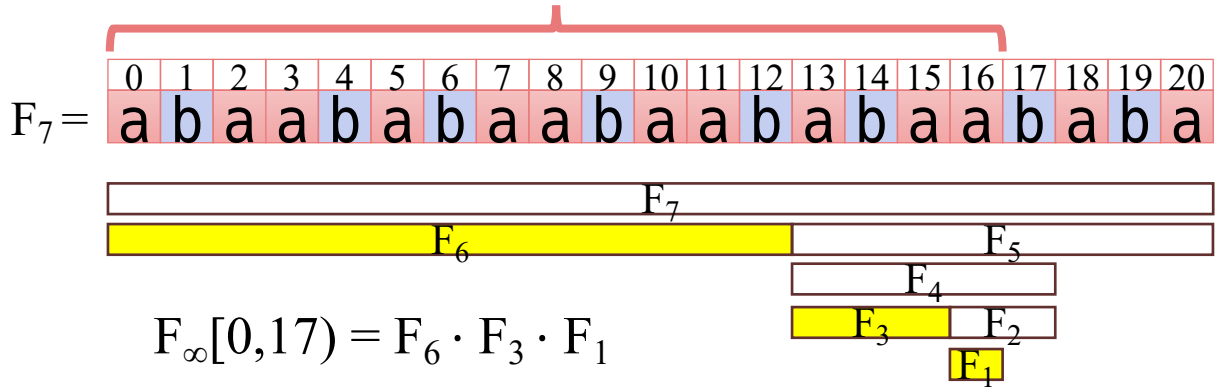
- Zeckendorf's Theorem [Zeckendorf 1972]
  - Any integer can be uniquely represented as the sum of non-consecutive Fibonacci numbers.

$17 = 1+3+13$

1	2	3	5	8	13	21
f_1	f_2	f_3	f_4	f_5	f_6	f_7
1	0	1	0	0	1	0

The bit representation does not contain "11"

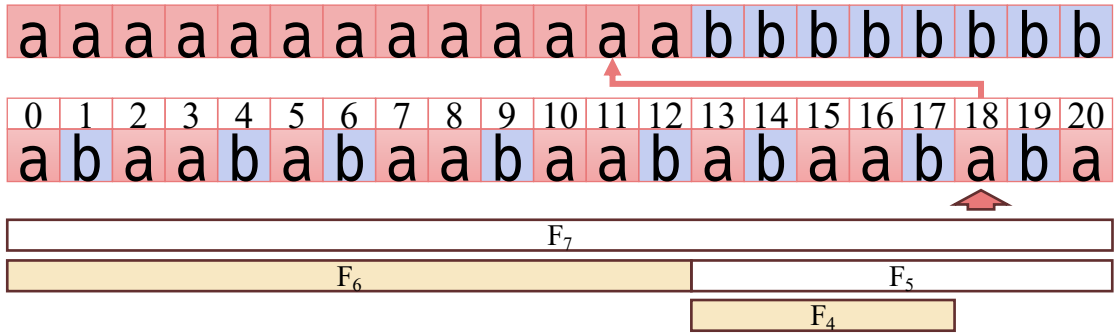
- Observation: The length i prefix of (infinite) Fibonacci word is concatenation of Fibonacci words whose lengths are used in Zeckendorf representation, in order of decreasing length.



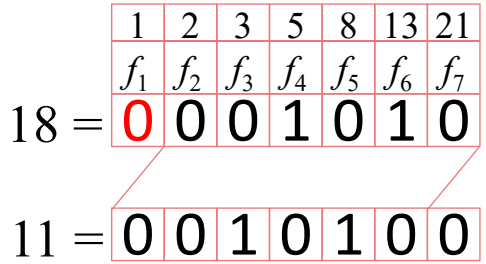
Characterization of BBWT^{-1}(Fibonacci word)

Lemma LF[i] for BBWT^{-1}(F_k) corresponds to bit rotations of $Z_k(i)$.

- $F_k[i] = a$: $LF[i] = |F_k[0..i)|_a$ LF[18]=11

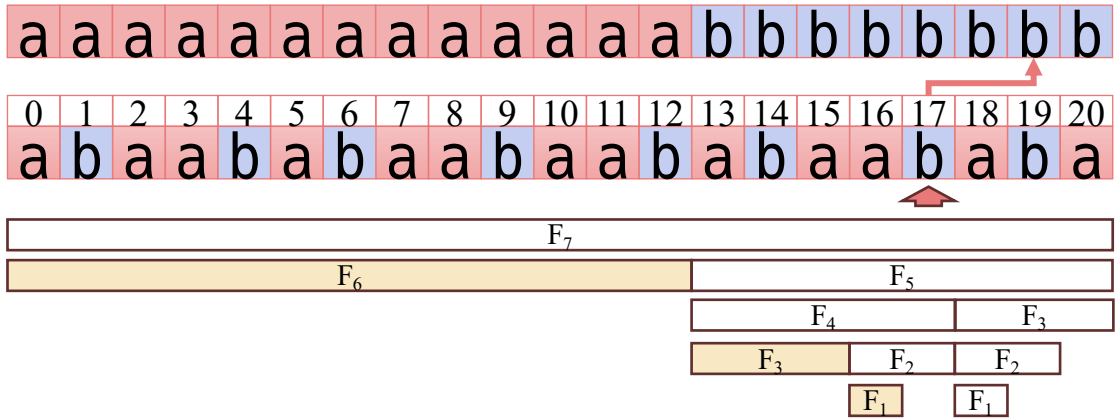


$|F_k|_a = f_{k-1}$

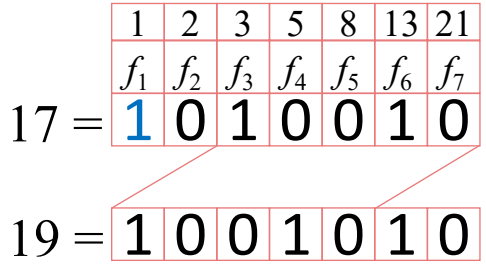


$LF[i]$ is 1-bit rotation of $Z_k(i)$

- $F_k[i] = b$: $LF[i] = f_{k-1} + |F_k[0..i)|_b$ LF[17]=19



$|F_k|_b = f_{k-2}$



$LF[i]$ is 2-bit rotation of $Z_k(i)$

δ of BBWT^{-1}(Fibonacci word)

Lemma LF[i] for $\mathrm{BBWT}^{-1}(\mathrm{F}_k)$ corresponds to bit rotations of $Z_k(i)$.

$\mathrm{F}_k[i]$ = a corresponds to 1 bit rotation
$\mathrm{F}_k[i]$ = b corresponds to 2 bit rotation

Therefore:

- A cycle in LF mapping of $\mathrm{BBWT}^{-1}(\mathrm{F}_k)$
  ⇔ a conjugacy class of k-bit Zeckendorf representation
  ⇔ primitive word of w where $|w|_{\mathsf{a}} + 2|w|_{\mathsf{b}}$ = divisor of k
  If we assume k is prime for simplicity,
  (except for a corresponding to 0^k : LF[0] = 0),
  each w corresponds to Lyndon factor of length < k,
  uniquely occurring in $\mathrm{BBWT}^{-1}(\mathrm{F}_k)$

- Any length $2k$ substring of $\mathrm{BBWT}^{-1}(\mathrm{F}_k)$ is unique
  ⇔ $\delta(\mathrm{BBWT}^{-1}(\mathrm{F}_k)) \geq (\mathrm{f}_k - 2k)/2k = \Omega(n/\log n)$

δ of BBWT^{-1}(Fibonacci word) – Example

Example
BBWT^{-1}(F_7)

1	2	3	5	8	13	21
f_1	f_2	f_3	f_4	f_5	f_6	f_7

	a	a	a	a	a	a	a	a	a	a	a	a	a	b	b	b	b	b	b	b	b
F_7 =	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	a	b	a	a	b	a	b	a	a	b	a	a	b	a	b	a	a	b	a	b	a

a
0000000(0) →
a
b a a a a a
1000000(1) → 0000010(13) → 0000100(8) → 0001000(3) → 0010000(5) → 0100000(2) → aaaaab
b b a a a
1010000(4) → 1000010(14) → 0001010(18) → 0010100(11) → 0101000(7) → aaabb
b a b a a
1001000(6) → 0100010(15) → 1000100(9) → 0010010(16) → 0100100(10) → aabab
b b b a
1010100(12) → 1010010(17) → 1001010(19) → 0101010(20) → abbb

BBWT^{-1}(F_7) = abbb aabab aaabb aaaaab a

Summary and Conclusion

New results on the compressibility of BWT, BBWT

1. Existence of infinite string families s.t. $r_B(T) = o(r(T))$.
2. For any T, $r(T)/r_B(T)$, $r_B(T)/r(T) = O(\text{polylog } n)$
3. On clustering effects of BWT and BBWT

   A) For RLE: BBWT shows similar traits to BWT.
   - $RLE(BBWT(T)) \leq 2 \cdot RLE(T)$

   B) For any measure μ between δ and RLE:
   - BWT, BBWT: increase in μ bounded by polylogarithmic factor.

   C) BBWT can transform a maximally incompressible string to very compressible string

   ➔ BBWT + dictionary compression is **strictly stronger** than dictionary compression

"Clustering effect" of BWT, BBWT is perhaps more powerful than previously perceived.

Future Work / Open problems

- Tighter bounds for 2(B)
- Is there similar example with BWT (not BBWT) for 3(C)?