領域効率のよい文法圧縮のための可変長符号（SPIRE 2012 発表予定）

髙畠嘉将（九工大）
田部井靖生（JST ERATO）
坂本比呂志（九工大）

CFG(文脈自由文法)による圧縮

w={bbacbbacbbacbbac}

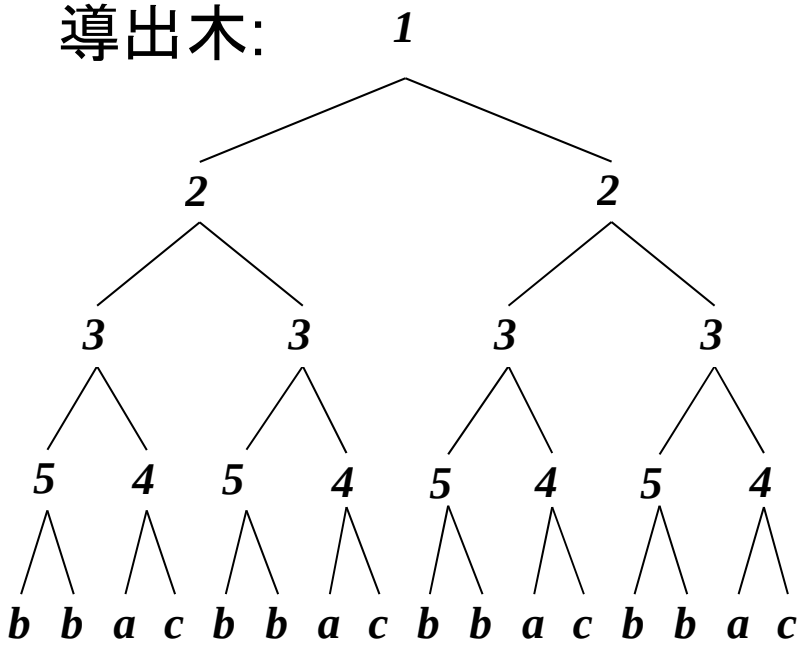
CFG

辞書

Z	X	Y
1	2	2
2	3	3
3	5	4
4	a	c
5	b	b

=

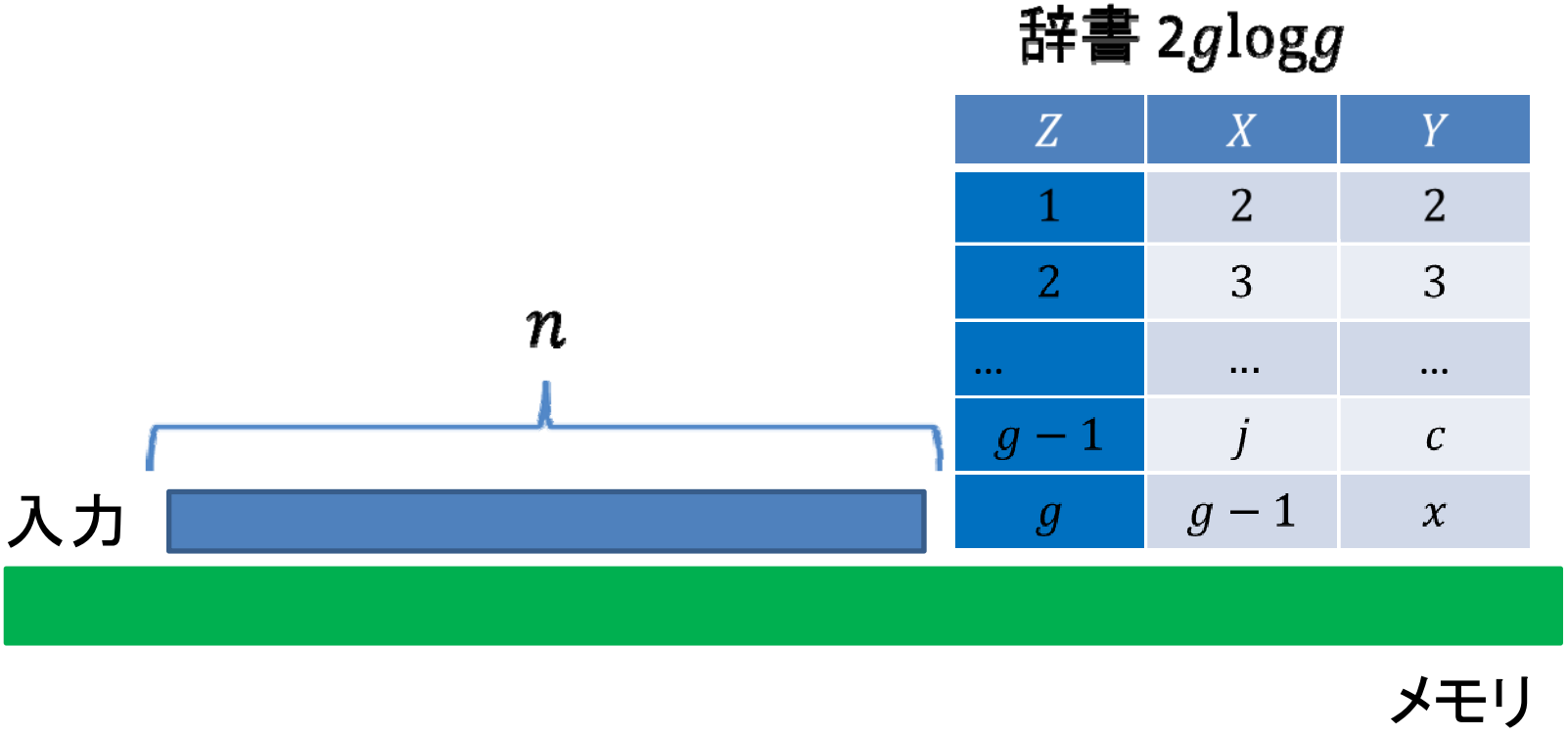
導出木:



最適な辞書を構築するのは、NP-困難　=>近似アルゴリズム

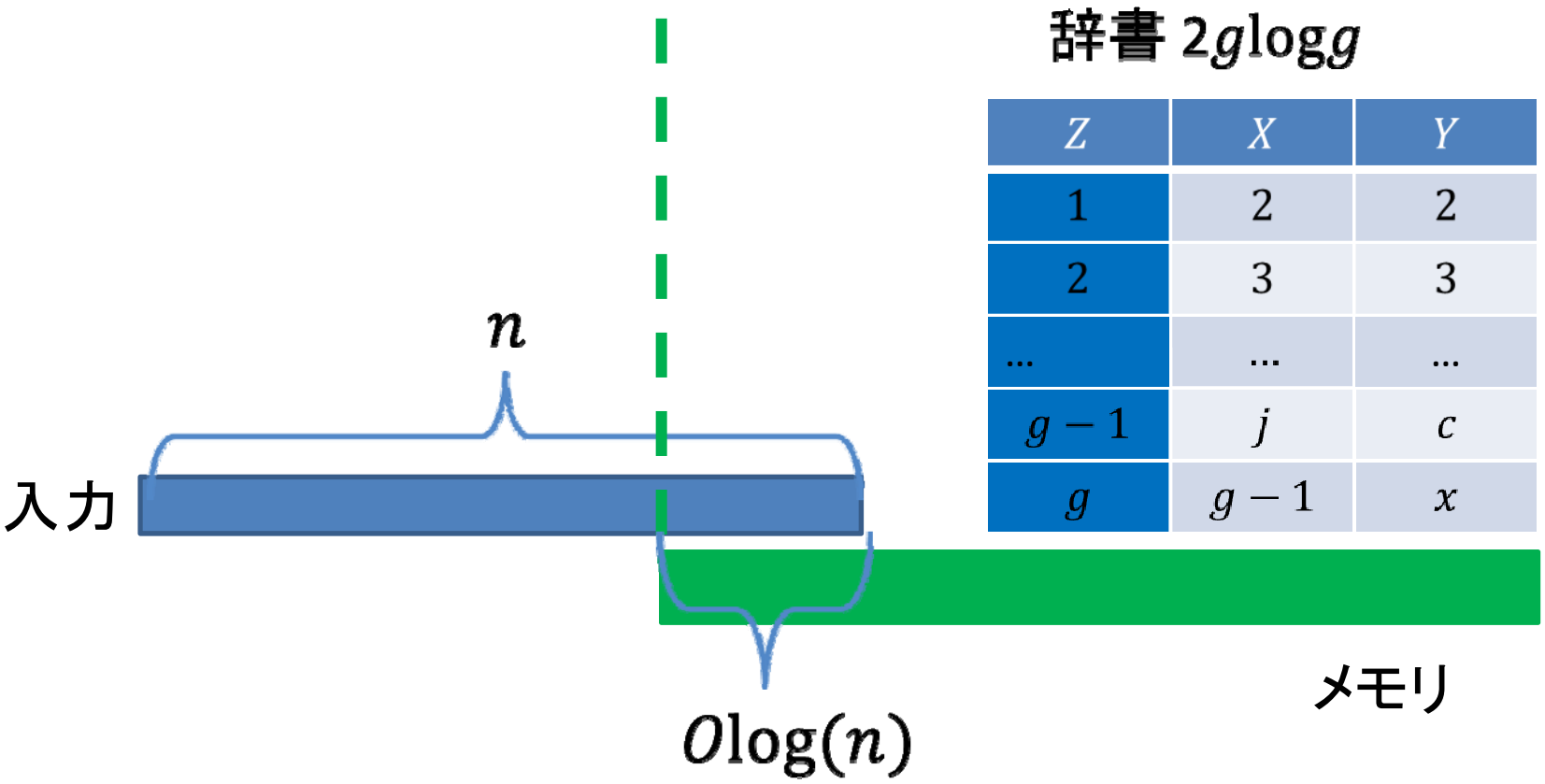
最近の研究

ESP-index(offline)[Maruyama,Kishiue,Nakahara,Sakamoto et el.,SPIRE 2011]



最近の研究

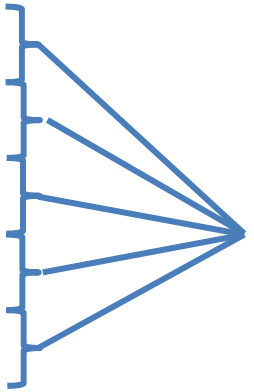
LCA-online[Maruyama,Takeda,Sakamoto et al.,Algorithms 2012]



辞書のサイズの問題

辞書　$2g\log g$

Z	X	Y
1	2	2
2	3	3
…	…	…
$g-1$	j	c
g	$g-1$	x



固定長

入力によっては辞書のサイズが大きい

メモリ使用量

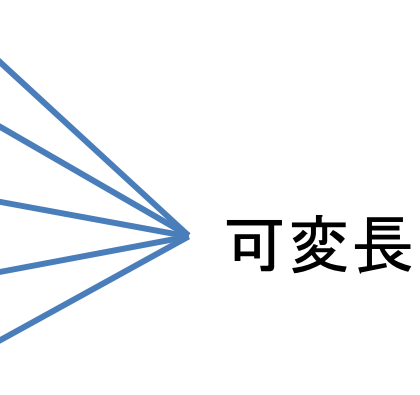
wikipedia

入力	辞書
5533MB	9401MB

本研究

辞書

Z	X	Y
1	2	2
2	3	3
…	…	…
$g-1$	j	c
g	$g-1$	x



- 動的に可変長で構築

＝＞メモリの省スペース化を実現

$$2g\log g \rightarrow \frac{3}{2} g\log g + \frac{15}{2} g$$

メモリ使用量

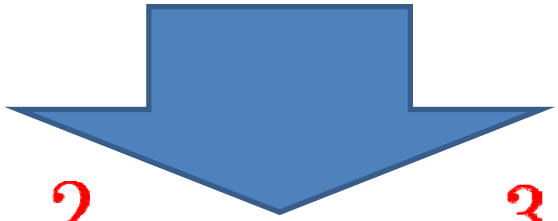
wikipedia

入力	辞書(固定長)	辞書(可変長)
5533MB	9401MB	4776MB

データを可変長で保存

辞書

Z	1	2
X	1: 14	3: 7
Y	2: 1	4: 33



1 byte表現
（C言語でのchar型）

1 2 3 4

00001110 00000001 00000111 01000001

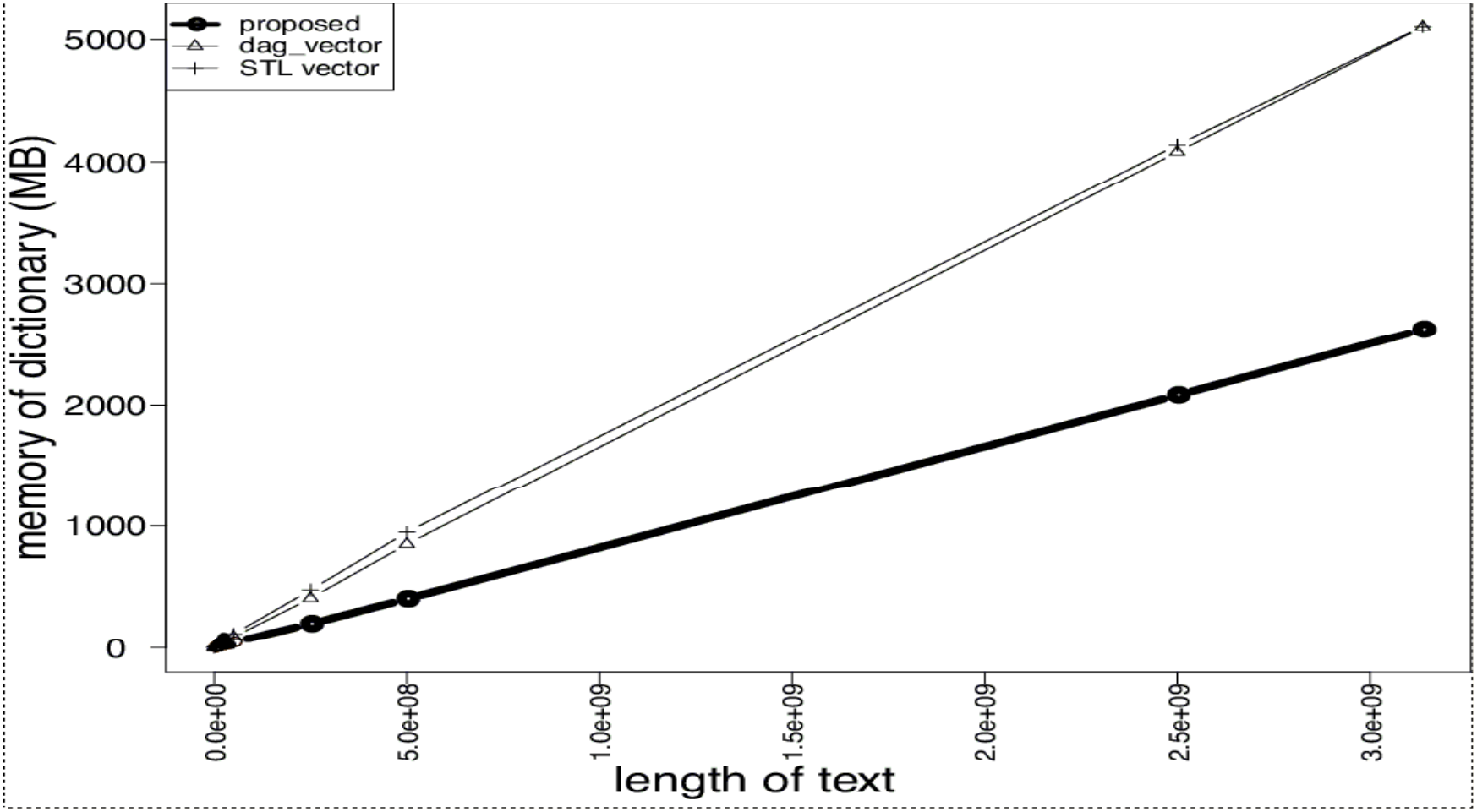
1 2 3 4

1110 1 111 1000001

実験環境

- Linux machine
- CPU:8-Core Intel(R) Xeon(R)CPU E7-8837 2.67GHz
- Memory:1TB
- 比較対象
- STL vector: C++のスタンダードテンプレートクラスのvector
- Dag vector:簡潔データ構造を用いたもの

実験(genome)


proposed
dag_vector
STL vector
memory of dictionary (MB)
5000
4000
3000
2000
1000
0
0.0e+00
5.0e+08
1.0e+09
1.5e+09
2.0e+09
2.5e+09
3.0e+09
length of text

まとめ

- $\frac{3}{2}g\log g+\frac{15}{2}g$ のメモリスペースで辞書を構築
- 動的に可変長で構築可能