Improved ESP-index

a practical self-index for highly repetitive texts

（SEA2014で発表予定）

髙畠嘉将(九州工業大学)

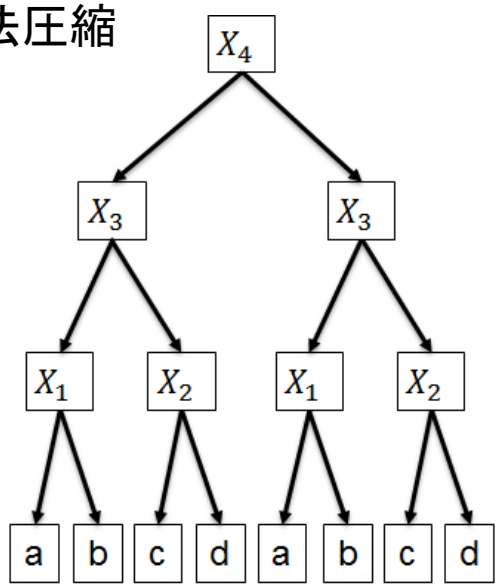
田部井靖生（JST–さきがけ）

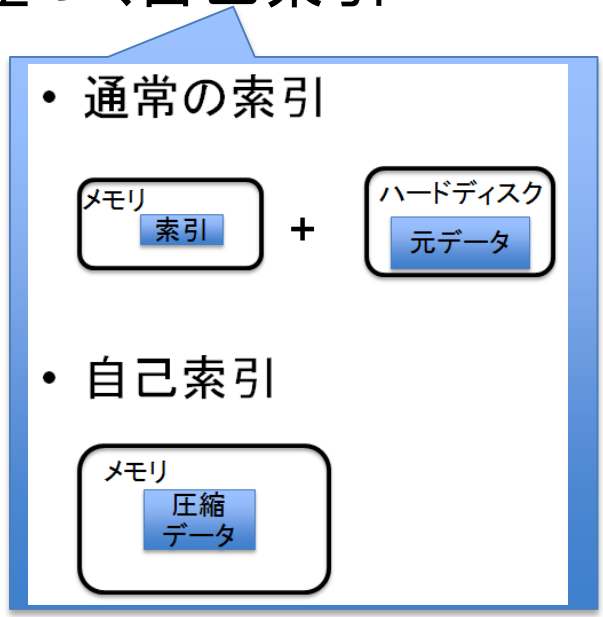
坂本比呂志(九州工業大学)

本研究の成果

- ESP-indexの文字列検索の高速化・大規模化
  - 文法圧縮に基づく自己索引

文法圧縮





n: 生成規則数
$0 < \epsilon < 1$
u:入力長
occ_c:coreの出現数

Searching time: $1/\epsilon(m \lg n + occ_c \lg m \lg u)\lg^* u$

二分探索$O(\lg n)$ →rank/select操作 $O(\lg\lg n)$

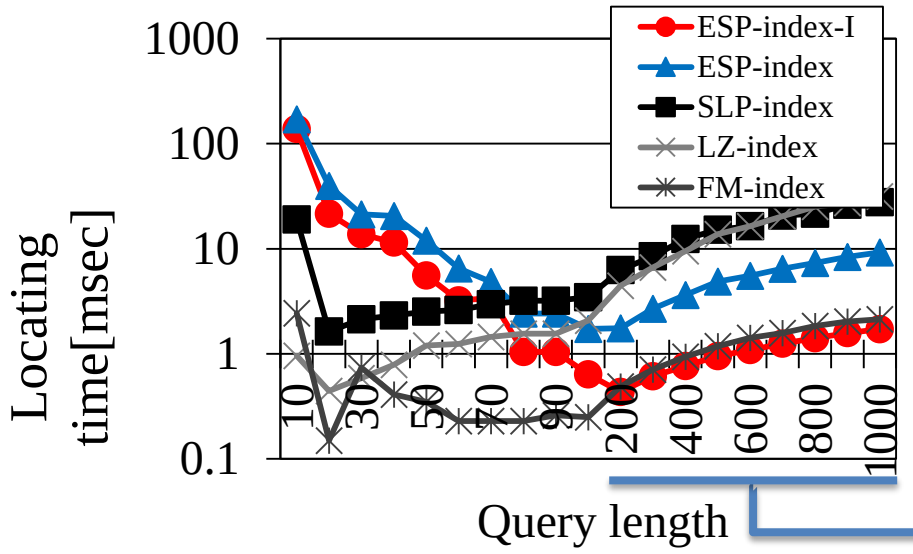
$$\lg\lg n(m + occ_c \lg m \lg u)\lg^* u$$

本研究の成果（実験1）

英文データ200MB(http://pizzachili.dcc.uchile.cl/texts/nlang/english.200MB)

	ESP-index-I	ESP-index	SLP-index	LZ-index	FM-index
Index size(MB)	165	162	209	282	482

ほぼ同じ

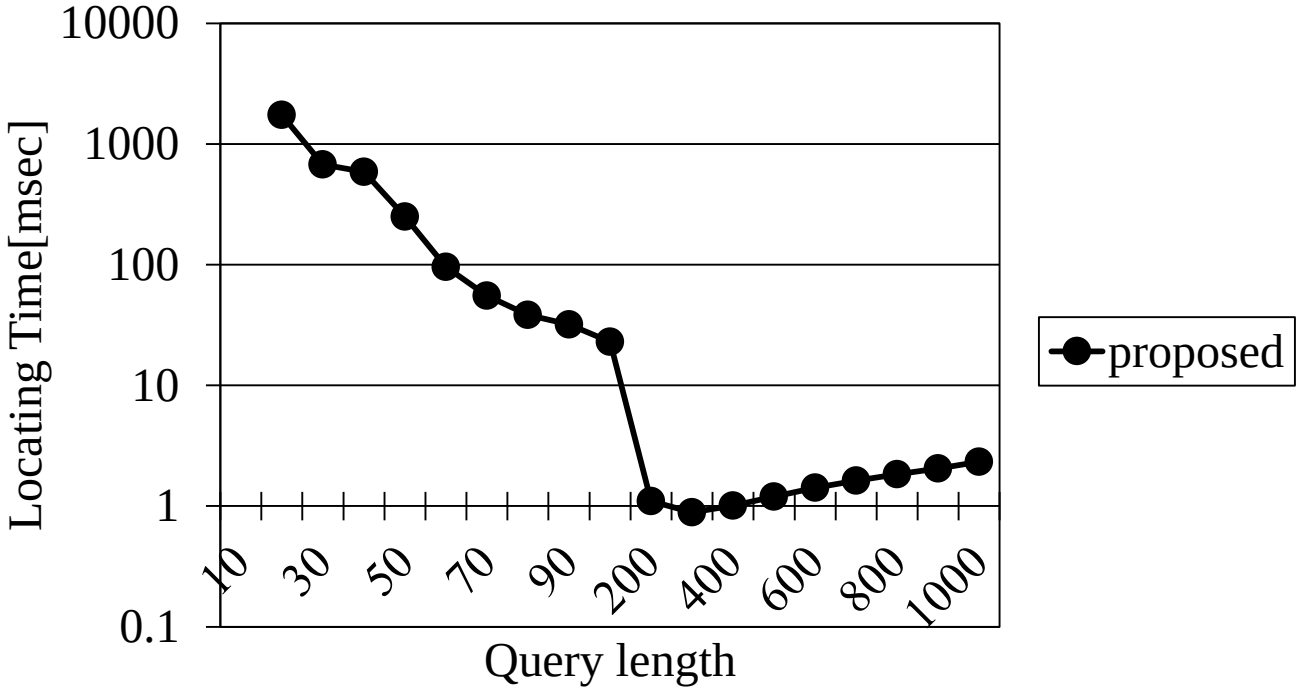


1.4～4.3倍の高速化

長いパターンの高速検索

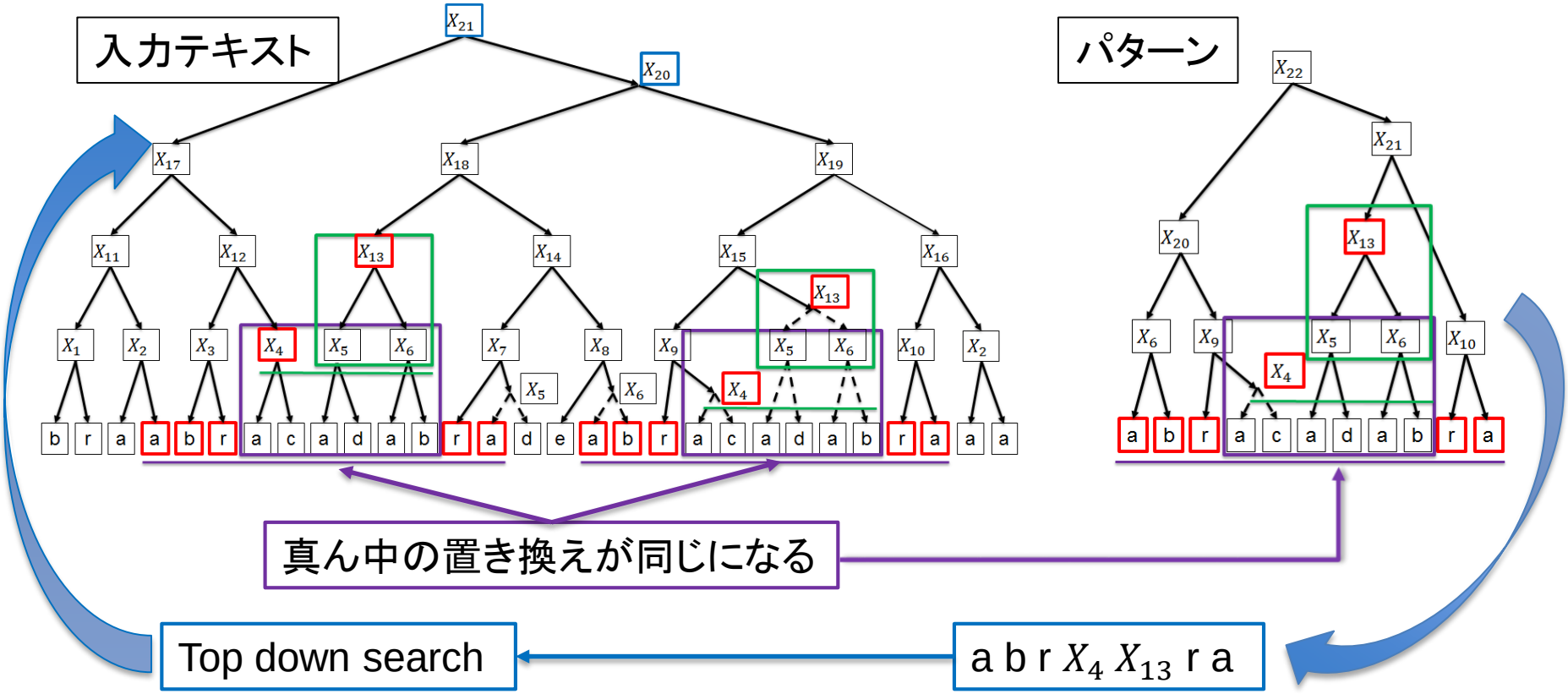
本研究の成果(実験2)

- 大規模テキストへの対応
  - genome(約12GB)



ESP-indexの検索

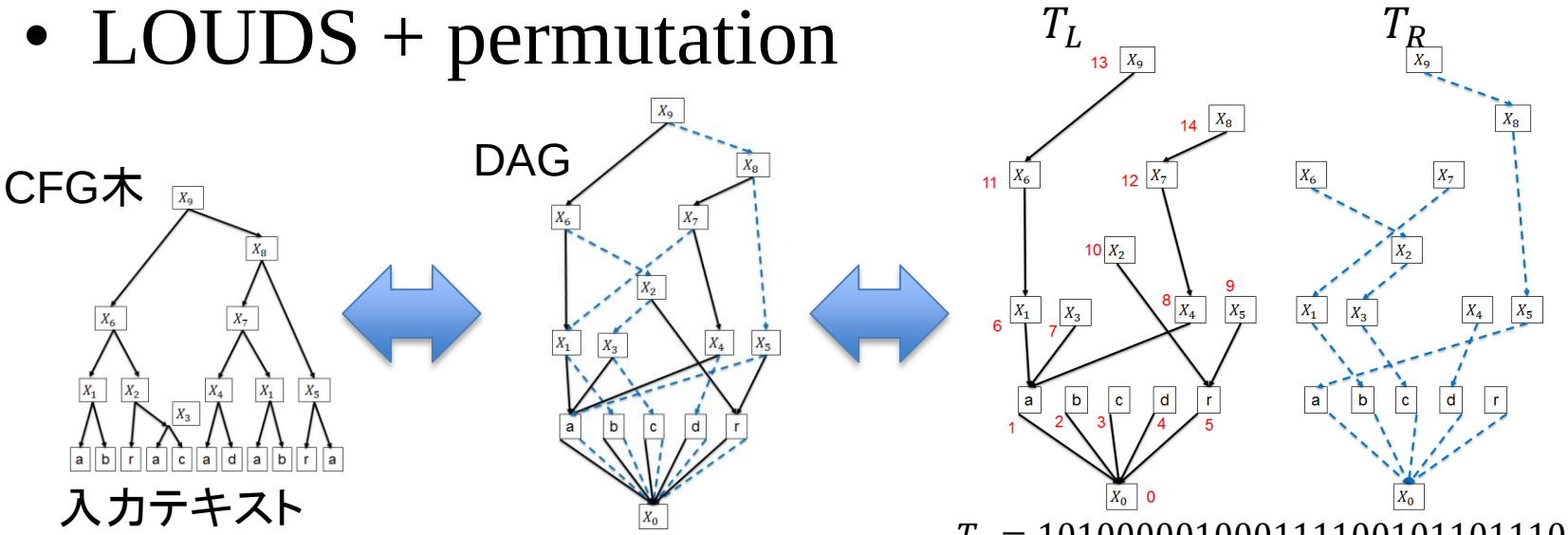
- ESP(Edit sensitive parsing)の性質を利用して検索



本研究では、パターンの圧縮時の$X_iX_j \rightarrow X_k$を高速化

データ構造（従来手法）

- LOUDS + permutation



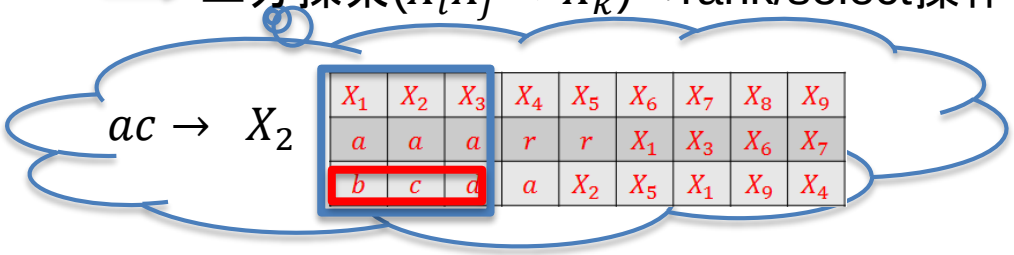
$T_L = 10100000100011110010110111010 11$
$T_R = 10100000101010101101011011010 11$
$\pi = 0,1,2,3,4,5,7,8,9,6,11,13,10,14,12$

Space:$n \lg u + (1+\epsilon)n \lg n + 4n + o(n)$

Searching time:$(1/\epsilon)(m \lg n + occ_c \lg m \lg u)\lg^* u$

二分探索$(X_i X_j \rightarrow X_k)$→rank/select操作で高速化

n: 生成規則数
$0 < \epsilon < 1$
u:入力長
occ_c:coreの出現数

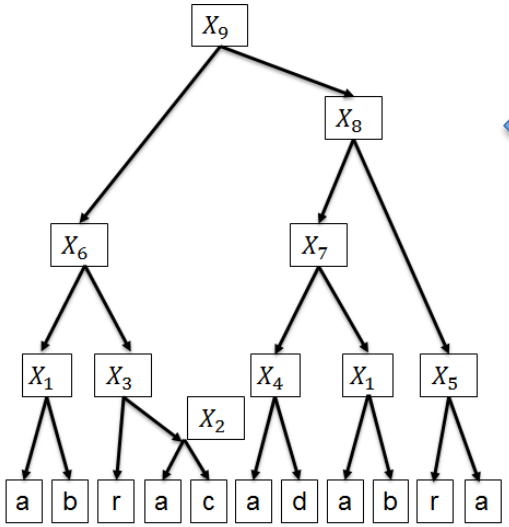


簡潔データ構造

- rank/select辞書 $(n + o(n)\text{bits}, O(1)$時間$)$
  - $rank_b(B, i)$:bit列$\boldsymbol{B}$のi番目までのbの数を返す
  - $select_b(B, i)$:i番目のbの位置を返す
  - 例） $rank_0(B, 7) = 4, select_1(B, 6) = 12$
  - $B = 01001010101110$
    7 12

- GMR $(n \lg n + o(n \lg n)\text{bits}, O(\lg\lg n)$時間$)$
  - 文字列に対するrank/select操作が可能
  - 例）$rank_p(S, 6) = 3, select_e(S, 2) = 7$
  - $S = \text{drpepper}$
    6 7

データ構造（提案手法）



入力テキスト

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
a	a	r	a	r	X_1	X_4	X_7	X_6
b	c	X_2	d	a	X_3	X_1	X_5	X_8

ソート

X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9
a	a	a	r	r	X_1	X_4	X_6	X_7
b	c	d	a	X_2	X_5	X_1	X_8	X_4

$\rightarrow A$ (GMR)

$B = 011100001010001001 01$ →rank/select辞書

size:$n \lg u + 2n + n \lg n + o(n \lg n)$

$X_i X_j \rightarrow X_k$

例）$a\ c \rightarrow X_2$

(1) $p = select_0(B, 1) - 1$ and $q = select_0(B, 2) - 2$

(2) $r = select_j(A, rank_j(A, p) + 1)$

(3) $k = r$ if $r \leq q$ else $X_k \rightarrow X_i X_j \notin$ CFG

searching time: $\lg\lg n(m + occ_c \lg m \lg u)\lg^* u$

実験結果

- Construction time(sec)

	ESP-index-I	ESP-index	SLP-index	LZ-index	FM-index
DNA(200MB)	81.8	82.96	1906.63	64.86	87.7
English(200MB)	93.36	93.58	1906.63	100.624	94.09

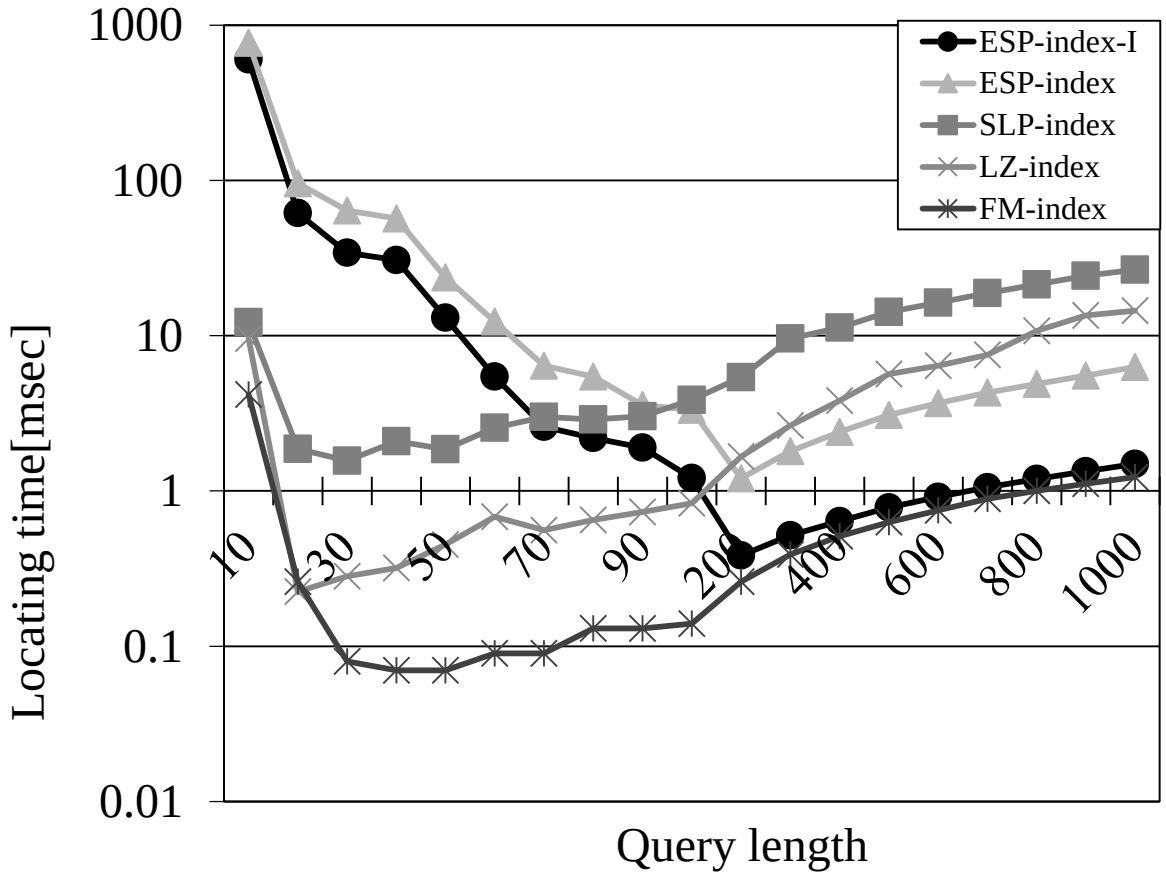
- Index size(MB)

	ESP-index-I	ESP-index	SLP-index	LZ-index	FM-index
DNA(200MB)	156	157	214	208	325
English(200MB)	165	162	209	282	482

- http://pizzachili.dcc.uchile.cl/

実験結果

- DNA(200MB)



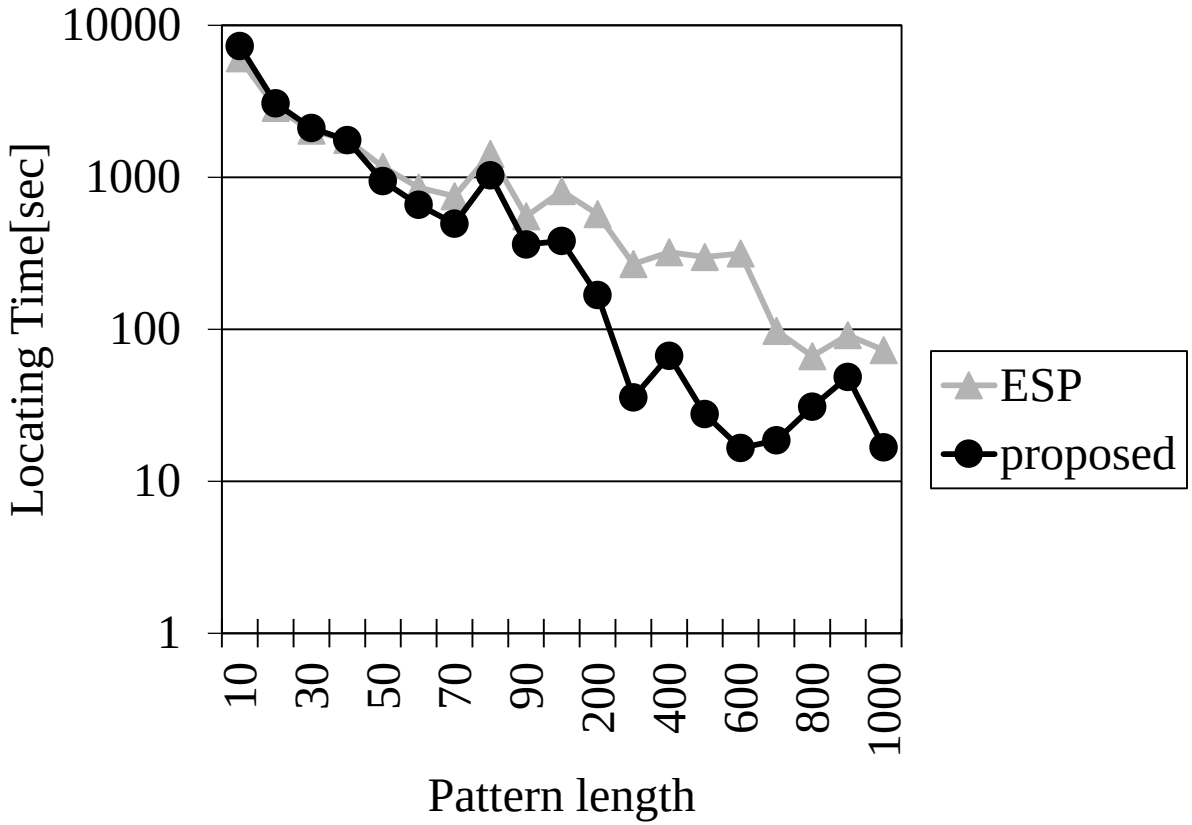
実験結果

- genome(12GB),wikipedia(7.8GB)

	genome		wikipedia	
$\|P\|$	200	1,000	200	1,000
Counting time (msec)	1.06	2.29	139.56	13.04
Locating time (msec)	1.10	2.33	167.40	16.69
Compression time (sec)	4,384		2,347	
Indexing time (sec)	567		74	
Index size (MB)	3,888		594	
Position size (MB)	1,526		246	

実験結果

- wikipedia(7.8GB)



おわりに

- まとめ
  - ESP-indexの高速化・大規模化
  - size:$n \lg u + 2n + n \lg n + o(n \lg n)$
  - searching time:$\lg\lg n(m + occ_c \lg m \lg u)\lg^* u$
  - extraction time:$\lg\lg n(m + \lg u)$
- 今後の課題
  - オンライン化

n: 生成規則数
u:入力長
occ_c:coreの出現数