



Title	木構造の文法圧縮
Author(s)	定兼, 邦彦
Citation	2010年度科学技術振興機構ERATO湊離散構造処理系プロジェクト講究録. p.500-502.
Issue Date	2011-06
Doc URL	<a href="http://hdl.handle.net/2115/48329">http://hdl.handle.net/2115/48329</a>
Type	conference presentation
Note	ERATO湊離散構造処理系プロジェクトシンポジウム(第2回): 第73回情報処理学会全国大会イベント企画. 2011年3月2日(水). 東京工業大学 大岡山キャンパス.
File Information	01.sadakane.pdf



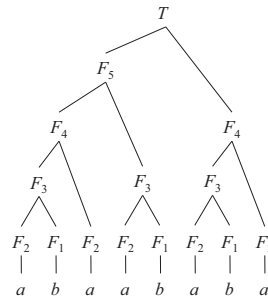
[Instructions for use](#)

### 木構造の文法圧縮

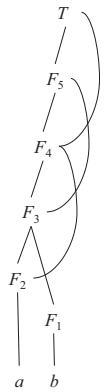
国立情報学研究所  
定兼 邦彦

### 文法圧縮

- 文字列をCFG (文脈自由文法) で表現
  - 入力文字列のみを生成する文法



- $F_1 \rightarrow b$
- $F_2 \rightarrow a$
- $F_3 \rightarrow F_2 F_1$
- $F_4 \rightarrow F_3 F_2$
- $F_5 \rightarrow F_4 F_3$
- $T \rightarrow F_5 F_4$



文字列を表現する木      文字列を表現する文法      DAG表現

### Straight-Line Programs

- 本研究ではstraight-line program (SLP) に限定
  - $X_i \rightarrow c$  ( $c \in A$ )
  - $X_i \rightarrow X_l X_r$  ( $l, r < i$ )
  - 入力文字列  $T[1..M]$  は  $X_n$  で表わされる
- 1つの文字列を表わすCFGはSLPに変換可
- 最短のSLPを求める ( $n$  の最小化) のはNP完全
  - $O(\log M)$  近似は線形時間  $O(M)$  で求まる
- $n$  個の規則のSLPは,  $O(n \log M)$  個の規則, 高さ  $O(\log M)$  のSLPに変換できる
- 本研究では, SLPは与えられているとする
  - (高さの制限は無い)

3

### 既存研究と本研究の結果

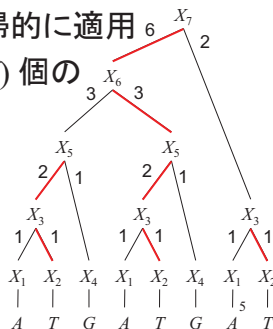
- Claude, Navarro MFCS09
  - 長さ  $m$  の部分文字列の復元:  $O((m+h) \log n)$  時間
  - パターン検索:  $O((m(m+h)+h \text{occ}) \log n)$  時間
  - ( $h$ : SLPの高さ,  $\text{occ}$ : パターンの出現回数)
  - サイズ:  $O(n \log n) + n \log N$  bits
- 本研究
  - 長さ  $m$  の部分文字列の復元:  $O(m + \log N)$  時間
  - サイズ:  $O(n \log N)$  bits
  - 編集距離  $k$  のパターン検索  $O(n(\min\{mk, k^4+m\}) + \text{occ})$  時間

4

### Heavy Path Decomposition

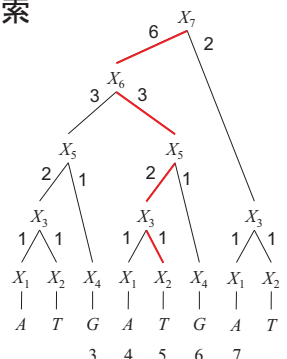
[Harel, Tarjan 84]

- Heavy edge: 大きい部分木への枝
- Heavy path: 根からheavy edgeをたどるパス
- 木からheavy pathを除き, 再帰的に適用
- 根から葉へのパスは  $O(\log N)$  個の heavy pathで表現される
- 各パスで2分探索
  - $\rightarrow O(\log N \cdot \log n)$  time
- 各パスを別々に格納
  - $\rightarrow O(n^2 \log N)$  bit space



### Interval-Biased Search

- $i$  番目の葉を探す場合, heavy path上で左右の部分木のサイズを元に2分探索
  - 左部分木のサイズ (右端の文字の位置)
  - $X_7 - X_6 - X_3 - X_2$
  - 3 4 5
  - 右部分木のサイズ (左端の文字の位置)
  - $X_7 - X_3 - X_5$
  - 7 6 5
- 配列での2分探索
  - $\rightarrow O(\log n)$  time
- Interval-Biased search
  - $\rightarrow O(\log N/x)$  time ( $N$  は木のサイズ)
  - ( $x$  は見つかった部分木のサイズ)



6



