先読み付き正規表現の決定性有限オートマトンによるマッチングの実装

横浜国立大学大学院　千田忠賢　倉光君郎
http://regex-and-pe-to-dfa.com

◆ 背景

- 先読み付き正規表現 (perl準拠)
  - 例 : (?=a). や (?!a).
- 既存の正規表現の処理系では先読みをバックトラックによって実装している
- 先読み付き正規表現をBoolean有限オートマトン(BFA)に変換し、これを決定性有限オートマトン(DFA)に変換する研究(森畑'12)
- 非終端記号を除く解析表現文法(PEG)と先読み付き正規表現を形式的に対応つけDFAに変換

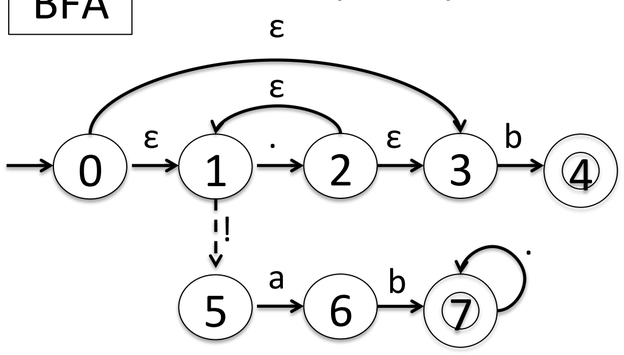
◆ 目的

- 森畑の研究に基づき先読み付き正規表現をBFAに変換し、DFAによるマッチングを実装
- PEGベースのパーサーライブラリを高速化
  - 非終端記号を除く解析表現をDFA化
  - PEGと正規表現の違い
    - 非終端記号
      - A ← 'abc'
    - 優先度付き選択
      - 'a' / 'aa' と a|aa
    - 繰り返し
      - 'a'*'a' と a*a
- PEGと先読み付き正規表現の対応関係
  - 優先度付き選択 $e_1 / e_2 \Rightarrow e_1 \mid (?!e_1)\, e_2$
  - 繰り返し $e^* = e^*(?!e)$

◆ 変換例 : ((?!ab).)*b

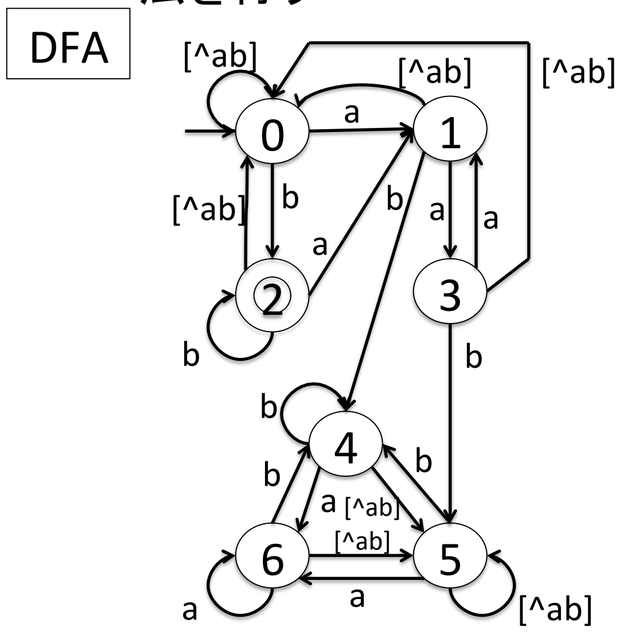
1. 先読み付き正規表現からBFAへ変換
   - 森畑(2012)

BFA



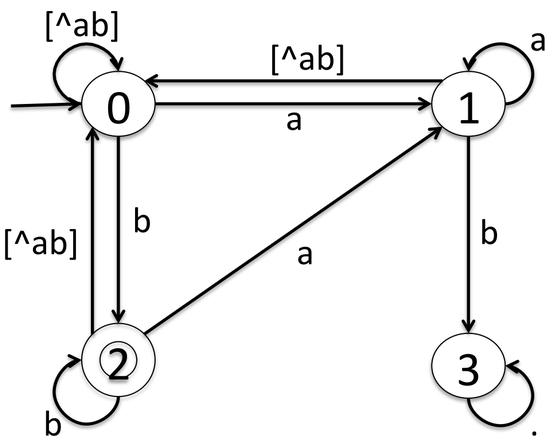
2. BFA から DFA へ変換
   - 論理式の等価判定 : Binary Decision Diagram (BDD)
   - BDDを用いて部分集合構成法を行う

DFA



3. DFAを最小化
   - Brzozowski's algorithm
   - Table-filling algorithm

最小のDFA



◆ コンパイル・実行速度

(1) ((?!(the|and|of|to|I)).)*(the|and|of|to|I)((?!(the|and|of|to|I)).)*
(2) ((?!(Sherlock|Homes)).)*(Sherlock|Holmes).*
(3) .*.*(?=the).*.*(?=.*Project.*).*(?=Gutenberg).*

pcregrep	nez	regex->BFA	BFA->DFA	Minimize
0.06sec	0.022sec	0.004sec	0.114sec	0.310sec
				0.053sec
0.14sec	0.010sec	0.004sec	0.075sec	0.543sec
				0.069sec
1.50sec	0.007sec	0.005sec	8.890sec	2.17sec
				0.116sec

◆ 今後の課題

- Partial matchの実装
  - 計算量の問題について
  - 完全にDFA化すると先読みの情報が消える場合がある
    - 例：a(?=a)
    - (a(?=a)).* から得られるDFAを元に計算できるか
    - そこから得られたDFAにAho-Corasickを用いることができるか
- PEGベースのパーサーライブラリへの応用
  - DFA化する箇所の決定方法