Рецензия к докладу "The Complexity of Compressed Membership Problems for Finite Automata"

1 Содержание доклада

В докладе был предъявлен алгоритм показывающий что FCMP (FULLY COMPRESSED MEMBERSHIP PROBLEM) для NFA лежит в NP, а для DFA лежит в P.

2 Основные определения

Определение. Дана грамматика с термами $X_1, X_2, \ldots, X_n$ и некоторым набором правил. Тогда $val(X_i)$ - множество слов из Σ^* получающиеся из терма X_i.

Определение. Назавем грамматику с термами $X_1, X_2, \ldots, X_s$ хорошей, если все правила имееют вид $X_i \to \omega_1 X_{i_1} \omega_2 X_{i_2} \ldots X_{i_n} \omega_{n+1}$, где $i_k < i$ и $\omega_k \in \Sigma^*$ для каждого k

Определение. Сжатием строки ω является грамматика G с термами $X_1, X_2, \ldots, X_s$ со следующими свойствами:

1. G-хорошая грамматика
2. $val(X_s) = \omega$

Замечание. В хорошей грамматике $val(X_i)$ всегда состоит из одного элемента Σ^*

Определение Дана хорошая грамматика с термамами $X_1, X_2, \ldots, X_s$. Тогда сжатым DFA(NFA) является DFA(NFA) на ребрах которого кроме элементов из Σ написаны термы $X_1, X_2, \ldots, X_s$ которые соответствуют переходам по символам $val(X_1), val(X_2), \ldots, Val(X_s)$

Определение Дана хорошая грамматика с термамами $X_1, X_2, \ldots, X_s$ и содержащая правило $X_i \to \omega_1 X_{i_1} \omega_2 X_{i_2} \ldots X_{i_n} \omega_{n+1}$, где $i_k < i$ и $\omega_k \in \Sigma^*$. Строка s пересекается с X_i если s подстрока в $val(X_i)$, но не является подстрокой $\omega_i, val(X_i)$ ни для какого i.

Определение. Пара ab называется пересекающейся, если ab и X_i пересекаются для любого i

Определение. Буква a пересекающаяся если пара aa пересекающаяся.

3 Ключевой результат

Теорема Даны сжатые при помощи одинаковой грамматики G с термами $X_1, \ldots, X_n$ строка $\omega \in \Sigma^*$ и $N \in DFA(NFA)$, тогда существует алгоритм работающий за полином от G, Σ, N строящий алфавит Σ', $N' \in DFA(NFA)$, $\omega' \in \Sigma'$, со следующим условием $\omega \in N \Leftrightarrow \omega' \in N'(*)$

Схема алгоритма. $While val(X_n) > n$:

1. Сжимаем все непересекающиеся пары, заменяя их на новые буквы.
2. Сжимаем все пересекающиеся пары, заменяя их на новые буквы.
3. Сжимаем все непересекающиеся буквы включая те что были добавлены выше.
4. Сжимаем все пересекающиеся буквы включая те что были добавлены выше.

После заменяем решаем все наивно.

Главная идея. Сжатие каждой непересекающейся пары/буквы является простой и обратимой операцией, а значит следить за сохранением условия $*$ легко. С другой стороны пересекающихся пар/букв довольно мало всего лишь линейное число от суммы длин правил и терминальных символов, а их количество походу работы не растет. А каждую пару/букву за полиномиальное время можно превратить в непересекающуюся пару и сжать.