

# Реализация алгоритма построения представления группы по машине Тьюринга

**Автор:** Шамрай Максим Борисович

**Научный руководитель:** доцент кафедры информатики,  
к. ф.-м.н. С. В. Григорьев

**Рецензент:** ведущий инженер ООО "Ланит-Терком"  
К. К. Смирнов

JetBrains Research, Programming Languages and Tools Lab  
Санкт-Петербургский государственный университет  
Системное программирование

- Кроме иерархии Хомского, есть довольно много классов формальных языков, например, конъюнктивные и булевы
- Не у всех есть критерий непредставимости языка в классе ( $Conj \subseteq Bool$  ?)
- В последнее время все чаще прибегают к смежным дисциплинам для исследования языков
- Предлагается построить представление группы по языку, чтобы в дальнейшем можно было применять аппарат теории групп для исследований

# Представление группы

Пусть  $\Sigma$  — конечный алфавит,  
 $\Sigma^{-1} = \{a^{-1} \mid a \in \Sigma, aa^{-1} = a^{-1}a = 1_G\}$ , тогда

- $\Sigma^+$  — свободная полугруппа
- $\Sigma^*$  — свободный моноид
- $(\Sigma \cup \Sigma^{-1})^*$  — свободная группа

# Представление группы

Пусть  $\Sigma$  — конечный алфавит,  
 $\Sigma^{-1} = \{a^{-1} \mid a \in \Sigma, aa^{-1} = a^{-1}a = 1_G\}$ , тогда

- $\Sigma^+$  — свободная полугруппа
- $\Sigma^*$  — свободный моноид
- $(\Sigma \cup \Sigma^{-1})^*$  — свободная группа

$G = \langle A \mid R \rangle$  — представление группы

- $G = \langle a, b \mid a^3, b^2, (ab)^2 \rangle = \{\varepsilon, a, a^2, b, ab, a^2b\} = S_3$
- $G = \langle a \mid a^5 \rangle = Z_5$
- $G = \langle a, b \mid aba^{-1}b^{-1} \rangle$

# Связь формальных языков с теорией групп

- Представления групп описывают языки, которые могут быть заданы следующим выражением:  
$$L(G) = \{\omega = 1_G \mid \omega \in (A \cup A^{-1})^*\}$$
- Построение представления группы по машине Тьюринга, которая распознает некоторый язык, было описано в статье<sup>1</sup>

## Теорема 1

Пусть  $L \subseteq \Sigma^+$  язык, принимаемый машиной Тьюринга  $M$ , тогда существует конечно представленная группа  $G(M) = \langle A \mid R \rangle$  и инъективное отображение  $K : \Sigma^+ \rightarrow (A \cup A^{-1})^+$  такое что:

$$u \in L \iff K(u) = 1_G$$

---

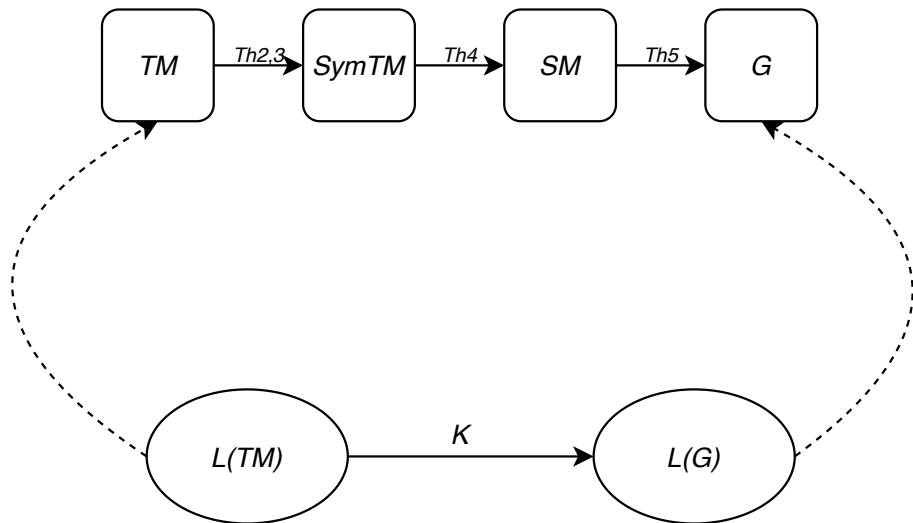
<sup>1</sup>Mark V. Sapir, Jean-Camille Birget and Eliyahu Rips "Isoperimetric and Isodiametric Functions of Groups" (2002)

**Цель:** Предоставить исследователям возможность ставить вычислительные эксперименты по преобразованию формального языка в соответствующее представление группы

**Задачи:**

- 1 Реализовать алгоритм преобразования контекстно-свободной грамматики в машину Тьюринга
- 2 Реализовать алгоритм построения представления группы по машине Тьюринга
- 3 Разработать интерпретаторы промежуточных машин для проверки сохранения языка
- 4 Провести эксперименты

# Схема построения представления группы



# Нотация машины Тьюринга

Машина Тьюринга имеет  $k$  лент и  $k$  голов и может быть описана как шестиместный кортеж:  $M = \langle X, \Gamma, Q, \Theta, \overline{s_1}, \overline{s_0} \rangle$ , где

- $X$  — входной алфавит.
- $\Gamma$  — алфавит лент.
- $Q = \cup_{i=1}^k Q_i$  — множество состояний голов на лентах машины.
- $\Theta$  — множество команд машины.
- $\overline{s_1}$  —  $k$ -вектор начальных состояний машины.
- $\overline{s_0}$  —  $k$ -вектор конечных состояний машины.

Команда одноленточной машины Тьюринга имеет вид:

$$uqv \rightarrow u'q'v'$$

где  $u, v, u', v'$  — ячейки,  $q, q'$  — состояния



# Построение распознавателя контекстно-свободной грамматики

- Алгоритм строит магазинный автомат, написанный в терминах машины Тьюринга, по контекстно-свободной грамматике
  - ▶ На входе — контекстно-свободная грамматика в нормальной форме Хомского
  - ▶ На выходе — машина Тьюринга, где первая лента входная, а вторая эмулирует стек автомата
- Для построения детерминированной машины Тьюринга по детерминированной грамматике при необходимости добавляются команды предпросмотра

# Симметризация недетерминированной машины Тьюринга

## Теорема 2

Для любой машины Тьюринга  $M$  существует недетерминированная машина Тьюринга  $M'$  со следующими свойствами:

- $M'$  симметричная
- Распознает тот же язык, что и  $M$
- Каждая команда действует только на одной ленте
- Для сохранения языка добавляется лента, алфавитом которой являются команды машины Тьюринга
- Получившаяся симметричная машина Тьюринга может иметь много состояний и состоять из многих команд, что говорит о ее сложности
- Но при этом ее можно построить и для детерминированной, и для недетерминированной исходной машины Тьюринга

# Симметризация детерминированной машины Тьюринга

## Теорема 3

Для любой детерминированной машины Тьюринга  $M$  существует эквивалентная симметричная машина Тьюринга, полученная из  $M$  добавлениями команд  $\tau^{-1}$  для каждой команды  $\tau$  из  $M$ .

- Авторы теоремы E. Post и A. A. Markov (1947)
- Получившаяся симметричная машина Тьюринга гораздо легче машины, полученной по предыдущей теореме
- Но ее можно построить только для детерминированной машины Тьюринга

# Построение представления группы

S-машина — система переписывания символов на ленте, которая поддерживает обратный алфавит.

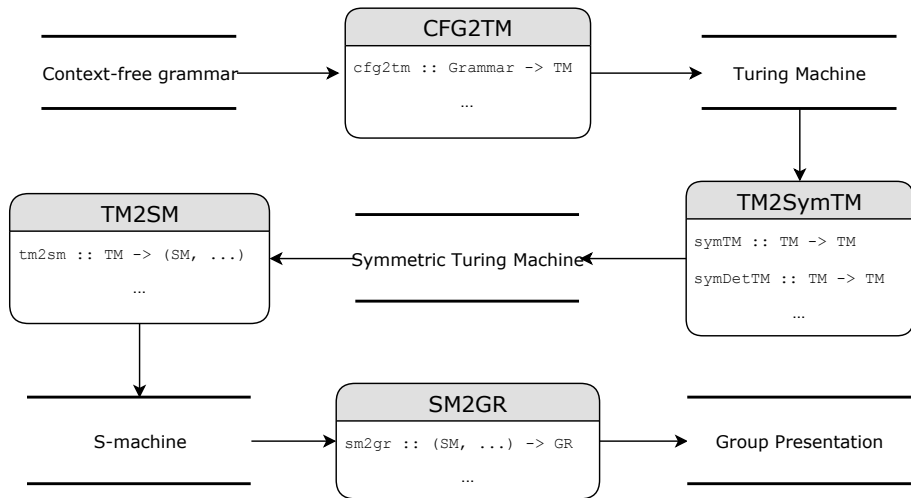
## Теорема 4

Для любой машины Тьюринга  $M'$  существует S-машина, которая симулирует  $M'$

## Теорема 5

Для любой S-машины существует соответствующая конечно-представленная группа

# Архитектура решения<sup>2</sup>



<sup>2</sup><https://github.com/YaccConstructor/LangToGroup>

- Необходимо проверять, сохраняется ли язык после каждого шага преобразования
- Так как авторы статьи используют свою эквивалентную нотацию машины Тьюринга, нами были разработаны интерпретаторы и S-машины, и машины Тьюринга
  - ▶ Дерево вычислений, корень — стартовая конфигурация
  - ▶ Обход дерева в ширину
  - ▶ В интерпретаторе S-машины используется множество пройденных конфигураций и построение дерева определенной высоты с выводом в DOT (graph description language)

Для оценки размера получившегося представления группы, запустили алгоритмы с недетерминированной и детерминированной симметризацией на трех грамматиках:

- one rule:  $S \rightarrow a$

- $a^*$ :  
 $S \rightarrow AS \mid \varepsilon$   
 $A \rightarrow a$

- Dyck:  
 $S \rightarrow AC \mid \varepsilon$   
 $C \rightarrow SD$   
 $D \rightarrow BS$   
 $A \rightarrow a$   
 $B \rightarrow b$

В таблице приведены мощности множеств

	Grammar			TM			
	$\Sigma$	$N$	$R$	$X$	$\Gamma$	$Q$	$\Theta$
1 rule	1	1	1	1	3	6	5
$a^*$	1	2	3	1	4	8	10
Dyck	2	4	6	2	8	13	21

TM'				SM			G	
$X$	$\Gamma$	$Q$	$\Theta$	$Y$	$Q$	$\Theta$	$A$	$R$
1	14	270	206	14	88246	2363	89508	56187
1	26	547	434	26	344118	5741	347370	204903
2	54	1131	900	54	1469136	15064	1478859	957619

<sup>3</sup>Используя алгоритм симметризации недетерминированных машин Тьюринга



В таблице приведены мощности множеств

	Grammar			TM			
	$\Sigma$	$N$	$R$	$X$	$\Gamma$	$Q$	$\Theta$
1 rule	1	1	1	1	3	6	5
$a^*$	1	2	3	1	4	8	10
Dyck	2	4	6	2	8	13	21

TM'				SM			G	
$X$	$\Gamma$	$Q$	$\Theta$	$Y$	$Q$	$\Theta$	$A$	$R$
1	6	39	34	6	6058	501	6410	7637
1	8	73	72	8	15888	1024	16565	17657
2	16	161	158	16	67754	2837	69772	71533

<sup>4</sup>Используя алгоритм симметризации детерминированных машин Тьюринга

- Размер получившихся представлений групп при симметризации в детерминированном случае гораздо меньше
  - ▶ Чем меньше получившееся представление группы, тем проще потом сказать о словах в группе, которые равны единице
  - ▶ В случае детерминированных грамматик, нужно использовать симметризацию для детерминированных машин
- Проведены эксперименты по интерпретации слов представлений групп
  - ▶ Математические пакеты Gap и Maple не справились с задачей проверки слов на равенство групповой единице
  - ▶ Актуальным остается вопрос разработки эффективного алгоритма интерпретации слов представлений групп

- ➊ Реализован алгоритм преобразования контекстно-свободной грамматики в машину Тьюринга
- ➋ Реализован алгоритм построения представления группы по машине Тьюринга
- ➌ Разработаны интерпретаторы промежуточных машин для проверки сохранения языка в ходе преобразований
- ➍ Проведен ряд экспериментов

# Проблема слов

$$G = \langle A \mid R \rangle, \Sigma = A \cup A^{-1}$$

$$\phi : \Sigma^* \rightarrow G$$

$$W(G) = \phi^{-1}(1)$$

- $W(G)$  – регулярна  $\iff G$  – конечна (Anisimov)
- $W(G)$  – контекстно-свободна  $\iff \exists H < G$  – свободная подгруппа конечного индекса (Muller–Schupp)

