5. Регулярные грамматики и языки

Разделы:

- Формальные грамматики
- Право- и леволинейные грамматики
- Свойства замкнутости РЯ
- Доказательство нерегулярности языков

Формальные грамматики

- Формальная грамматика - четверка элементов $G = (V_T, V_N, P, S)$
- V_T – это алфавит **терминальных** символов
- V_N – это алфавит **нетерминальных** символов
- Объединение этих множеств является **алфавитом** языка, а пересечение дает пустое множество
- Символом P обозначается множество продукций, каждый элемент которого состоит из пары (α, β)
  - Здесь α – левая часть продукции, β – правая часть продукции
  - Сама продукция обычно записывается как α -> β
  - При этом α € V^+, β € V^*
- Мы будем использовать также сокращенную запись для одинаковых правых частей
  - α -> β | γ
- Символ S принадлежит V_N - **аксиома**, или **начальный символ** грамматики

Формальные грамматики

- Грамматика используется для генерации (**порождения**) цепочек символов
- Процесс начинается с аксиомы
- Он выполняется как последовательность замен нетерминальных символов из левой части продукции на их правые части
- Получаемые в результате последовательности терминалов и нетерминалов – сентенциальные формы (СФ)
- Процесс **прекращается** при получении СФ, состоящей **только из терминальных символов**
- Языку принадлежат те и только те цепочки (строки), которые можно получить с помощью заданной грамматики

Регулярные грамматики

- Грамматика $G = (V_T, V_N, P, S)$ называется **праволинейной** (ПЛГ), если все ее продукции имеют две формы:
  - $A \text{ -> } xB \mid A \text{ -> } x$
- Грамматика $G = (V_T, V_N, P, S)$ называется **леволинейной** (ЛЛГ), если все ее продукции имеют две формы:
  - $A \text{ -> } By \mid A \text{ -> } y$
- Всякая ЛЛГ и всякая ПЛГ является **регулярной грамматикой** (РГ)

Регулярные грамматики

- Пример ПЛГ:
  - $G_1 = (\{a, b\}, \{S\}, \{S \text{ -> } abS \mid a\}, S)$
  - $S \Rightarrow abS \Rightarrow ababS \Rightarrow ababa$
  - "=>" – это символ порождения терминальной строки с использованием продукций грамматики
- Пример ЛЛГ:
  - $G_2 = (\{a, b\}, \{S, S_1, S_2\}, \{S \text{ -> } S_1ab,$ $S_1 \text{ -> } S_1ab \mid a, S_2 \text{ -> } a\}, S)$
- Грамматика $G_{MMXV} = (\{a, b\}, \{S, A, B\}, \{S \text{ -> } A, A \text{ -> } aB \mid \varepsilon, S \text{ -> } Ab\}, S)$ не является РГ
- **СТУДЕНТАМ**: Почему G_{MMXV} не РГ?

Регулярные грамматики

- Покажем, что язык, генерируемый ПЛГ всегда РЯ
- Для этого конструируем НКА, который имитирует процесс порождения по ПЛГ
- Все СФ ПЛГ имеют такой вид, что в них ровно один нетерминал, и он появляется в крайней правой позиции
- Пусть выполняется такой шаг порождения:
  - *ab...cD => ab...cdE*,
- Это делается применением продукции *D -> dE*
- В НКА это переход из состояния *D* в состояние *E* по символу *d*
- Здесь состоянию автомата соответствует нетерминал в СФ, когда уже обработанная часть входной строки идентична терминальному префиксу СФ

Регулярные грамматики

Теорема 5.1. Если грамматика G является ПЛГ, то $L(G)$ – РЯ.

Доказательство. Мы предполагаем, что $V = \{V_1, V_2, ...\}$, и продукции имеют форму $V_0 \text{ -> } v_1V_i$, $V_i \text{ -> } v_2V_j$, ...,$V_n \text{ -> } v_l$, ... Если w – строка из $L(G)$, то вследствие формы продукций порождение будет иметь следующую форму:

$$V_0 => v_1V_i => v_1v_2V_j =>^* v_1v_2...v_kV_n => v_1v_2...v_kv_l = w$$

Автомат, который мы конструируем, будет репродуцировать это порождение путем поочередного «потребления» каждого из этих v. Стартовое состояние будет помечено нетерминалом V_0, а для каждого нетерминала V_i мы будем иметь состояние V_i. Соответственно, для каждой продукции вида $V_i \text{ -> } a_1a_2...a_mV_j$ у автомата будут переходы, соединяющие V_i и V_j. Значит, расширенная функция переходов может быть определена так:

$$\hat{\delta}(V_i, a_1a_2\ldots a_m) = V_j\,.$$

Для каждой продукции вида $V_i \text{ -> } a_1a_2...a_\mathrm{m}$ соответствующий переход автомата будет определяться функцией:

$\hat{\delta}(V_i, a_1a_2\ldots a_m) = V_f$, где V_f – это заключительное состояние.

Промежуточные состояния, которые нужны, чтобы сделать это, необязательны. Их можно задать произвольными метками. Полный автомат собирается из таких индивидуальных частей.

Теперь предположим, что строка w из $L(G)$ такая, что удовлетворяет выражению (5.1). В НКА есть путь из V_1 в V_i с меткой v_1, путь из V_i в V_j с меткой v_2, и так далее, тогда $V_f \in \hat{\delta}(V_0, w)$, и строка w принимается НКА.

И обратно, предположим, что строка w принимается НКА. Из-за наличия способа конструирования НКА для того, чтобы принять строку автомат должен пройти через последовательность состояний $V_0, V_1,\ldots,V_f$ с метками v_1, v_2, … Следовательно, строка должна иметь форму $w = v_1v_2...v_kv_l$ и порождение

$$V_0 => v_1V_i => v_1v_2V_j =>^* v_1v_2...v_kV_n => v_1v_2...v_kv_l$$

возможно. Следовательно, w принадлежит $L(G)$, и теорема доказана.

Регулярные грамматики

- Создадим КА, принимающий язык, сгенерированный продукциями (V_0->aV_1, V_1 ->abV_0 | b)
  - Начинаем построение графа переходов с вершин V_0, V_1, V_f
  - Первая продукция дает ребро от V_0 к V_1 с меткой a
  - Согласно второй продукции потребуется создать дополнительную вершину графа, чтобы появился путь от V_1 к V_0 с меткой ab
  - Нам нужно ребро с меткой b между V_1 к V_f
- Язык, который принимается полученным КА и генерируется заданной РГ, описывается РВ (*aab*)*(*ab*)

Регулярные грамматики

Теорема 5.2. Если L – РЯ на алфавите Σ, то существует ПЛГ G такая, что $L = L(G)$.

Доказательство. Пусть A – это ДКА, который принимает язык L. Полагаем $Q = \{q_1, q_2, \ldots, q_n\}$ и $\Sigma = \{a_1, a_2, \ldots, q_m\}$. Сконструируем ПЛГ $G = (\Sigma, V, P, S)$, где V = $\{q_1, q_2, \ldots, q_n\}$ и $S = q_0$. Для каждого перехода $\delta(q_i, a_j) = q_k$ мы добавляем в P продукцию

$$q_i \to a_j q_k \qquad (5.2)$$

Если q_k принадлежит F, то в P добавляется продукция

$$q_k \to \varepsilon \qquad (5.3)$$

Покажем, что грамматика G, созданная таким образом, может генерировать любую строку в L. Рассмотрим w из L, причем $w = a_i a_j \ldots a_k a_l$. Автомат A для приема этой строки должен двигаться через

$\delta(q_0, a_i) = q_p$, $\delta(q_p, a_j) = q_r$, …, $\delta(q_s, a_k) = q_t$, $\delta(q_t, a_l) = q_f$ из F.

Согласно нашему построению в грамматике будет одна продукция для каждой из этих δ. Следовательно, мы можем осуществить

$$q_0 \Rightarrow a_i q_p \Rightarrow a_i a_j q_r \Rightarrow^* a_i a_j \ldots a_k q_i \Rightarrow a_i a_j \ldots a_k a_l q_f \Rightarrow a_i a_j \ldots a_k a_l \qquad (5.4)$$

с использованием грамматики G и w из $L(G)$.

И обратно, если w принадлежит $L(G)$, то ее порождение должно иметь форму (5.4). Это означает, что

$\hat{\delta}(q_0, a_i a_j \ldots a_k a_l) = q_f$. Что и требовалось доказать.

Регулярные грамматики

- Создадим ПЛГ для языка $L(aab^*a)$
- Функция переходов для НКА может быть получена по теореме 5.2
- Переходы и продукции:

$\delta(q_0, a) = \{q_1\}$	$q_0 \text{->} aq_1$
$\delta(q_1, a) = \{q_2\}$	$q_1 \text{->} aq_2$
$\delta(q_2, b) = \{q_2\}$	$q_2 \text{->} bq_2$
$\delta(q_2, a) = \{q_f\}$	$q_2 \text{->} aq_f$
q_f из F	$q_f \text{->} \varepsilon$

- Строку *aaba* можно породить следующим образом

    q0 => aq1 => aaq2 => aabq2 => aabaqf => aaba

Регулярные грамматики

Теорема 5.3. Язык L – РЯ тогда и только тогда, когда существует ЛЛГ G такая, что $L = L(G)$.

Доказательство. Мы дадим лишь общую идею. Дана ЛЛГ с продукциями в форме A -> Bv и A -> v. Мы преобразуем ее в ПЛГ G_{rl} путем замены каждой продукции G, соответственно, на A -> $v^R B$ или A -> V^R. Потренировавшись на нескольких примерах, мы поймем, что $L(G) = (L(G_{rl}))^R$. Как известно, обращение любого РЯ так же является РЯ. Поскольку G_{rl} – ПЛГ, то $L(G_{rl})$ – является РЯ, но тогда такими также будут $L(G_{rl}))^R$ и $L(G)$. Что и требовалось.

Совмещая теоремы 5.2 и 5.3 мы подошли к эквивалентности РЯ и РГ.

Теорема 5.4. Язык L – РЯ тогда и только тогда, когда существует РГ G такая, что $L = L(G)$.

Без доказательства

Замкнутость РЯ

- Есть два РЯ – L и M над алфавитом Σ
- Операция **объединения** двух языков нами рассматривалась ранее
- **Пересечением** называется язык , который содержит все строки, принадлежащие обоим языкам
- **Дополнением** языка L называется язык L_{compl}, который содержит множество тех строк в алфавите Σ^*, которые не принадлежат L

Замкнутость РЯ

Теорема 5.5. Если L и M – РЯ, то их объединение тоже РЯ.

Доказательство. Поскольку оба языка регулярны, то им соответствуют некоторые РВ. Пусть $L = L(R)$ и $M = L(S)$. Тогда $L \cup M = L(R+S)$ согласно определению операции объединения для РВ. Все оказалось не так страшно.

При определении свойства замкнутости относительно дополнения, также используют РВ. Разумеется, легким образом преобразовать РВ так, чтобы оно представляло собой дополнение заданного языка, у нас не получится. Однако это возможно, если следовать простой методике:

1. Преобразовать РВ в ε-НКА.
2. Преобразовать ε-НКА в ДКА с помощью конструкции подмножеств.
3. Дополнить заключительные состояния этого ДКА.
4. Преобразовать полученный ДКА обратно в РВ, используя известные способы.

Теорема 5.6. Если L – РЯ, то язык $L_{compl} = \Sigma^* - L$ тоже РЯ.

Доказательство. Пусть $L = L(A)$ для некоторого ДКА $A(Q, \Sigma, \delta, q_0, F)$. Тогда $L_{compl} = L(B)$, где B – это ДКА $(Q, \Sigma, \delta, q_0, Q - F)$. Иначе говоря, оба автоматы отличаются только тем, что заключительные состояния A стали незаключительными состояниями в B, и наоборот. Тогда w принадлежит $L(B)$, если и только если $\hat{\delta}(q_0, w)$ принадлежит $Q - F$, т.е. w не принадлежит $L(A)$.

Замкнутость РЯ

- Язык, содержащий строки из 0 и 1, которые всегда заканчиваются на 01, определяется РВ (0+1)*01
- Дополнением является язык, содержащий строки из 0 и 1, которые **не заканчиваются** на 01
- Он описывается РВ (1+00*1(00*1)*1)*
- После построения ДКА по РВ оба автомата отличаются только тем, что заключительные состояния стали незаключительными и наоборот

Замкнутость РЯ

- **СТУДЕНТАМ**: как выразить операцию пересечения через объединение и дополнение?
- Мы можем непосредственно построить ДКА для пересечения двух РЯ

Теорема 5.7. Если L и M – РЯ, то язык $L \cap M$ тоже РЯ.

Доказательство. Пусть ДКА $A_L(Q_L, \Sigma, \delta_L, q_L, F_L)$ и $A_M(Q_M, \Sigma, \delta_M, q_M, F_M)$ – это ДКА для заданных языков. Ограничение на детерминированность – несущественно, можно строить и НКА. Мы должны сконструировать ДКА A, который является комбинацией $A = (Q_L \times Q_M, \Sigma, \delta, (q_L, q_M), (F_L \times F_M))$, где $\delta((p,q),a) = (\delta_L(p,a), \delta_M(q,a))$.

Достаточно легко с помощью индукции показать, что любая строка w принимается таким объединенным автоматом, если и только если ее допускают оба исходных автомата, т.е. $\hat{\delta}((q_L, q_M), w) = (\hat{\delta}_L(q_L, w), \hat{\delta}_M(q_M, w))$.

Замкнутость РЯ

- Если L и M – языки, то **разностью** L – M называется множество строк, которые принадлежат L и не принадлежат M
- РЯ замкнуты относительно разности
- **Теорема 5.8.** Если L и M – РЯ, то язык L - M тоже РЯ
- **Доказательство.** Заметим, что $L - M = L \cap M_{compl}$
- По теореме 5.6 язык M_{compl} – РЯ
- По теореме 5.7 – $L \cap M_{compl}$
- Значит, (L – M) – РЯ
- Свойства замкнутости РЯ относительно **конкатенации** и **итерации** доказываются также, как и доказательство замкнутости относительно объединения, т.е. через операции над РВ
- **СТУДЕНТАМ**: Приведите эти доказательства

Замкнутость РЯ

- **Обращением строки** $a_1a_2\dots a_n$ называется строка, записанная в обратном порядке, и для произвольной строки w обозначается как w^R
  - **СТУДЕНТАМ**: чему равно 0010^R и ε^R?
- **Обращение языка** L, обозначаемое через L^R, состоит из всех строк, обратных строкам языка L
  - **СТУДЕНТАМ**: чему равно L^R для $L = \{001, 10, 111\}$?

Замкнутость РЯ

- Если для автомата A есть язык L такой, что $L=L(A)$, то можно построить КА для L^R следующим образом
  1. Обратить все дуги на диаграмме переходов автомата A
  2. Сделать начальное состояние A единственным заключительным состоянием нового автомата
  3. Создать начальное состояние p_0 с ε-переходами во все заключительные состояния автомата A
- Будет получен КА, имитирующий A в обратном порядке, а значит, допускающий строку w тогда и только тогда, когда A допускает w^R

Замкнутость РЯ

- Есть другое доказательство – через РВ
- Оно сводится к базисным правилам и индукции по трем РВ-операторам
- **Базис**: Для РВ $E = \varepsilon, \emptyset, a$ (из алфавита), обращение $E^R = E$
- **СТУДЕНТАМ**: проведите индукцию
- Пусть язык L определяется РВ (0+1)0*
- Тогда по правилу конкатенации L^R – это язык, описываемый выражением $(0^*)^R(0+1)^R$
- Если применять правила итерации и объединения к двум частям этого выражения, а потом использовать базисное правило, то получим, что язык L^R определяется РВ (0+1)0*

Замкнутость РЯ

- **Гомоморфизм строк** – это такая функция h на множестве строк, которая подставляет определенную строку вместо каждого ее символа
  - Пример: если $h(0)=ab$ и $h(1)=\varepsilon$, тогда $h(1100)=abab$
- **Гомоморфизм языка** определяется с помощью его применения к каждой строке языка
- Иными словами, если L – язык в алфавите Σ, а h – гомоморфизм на Σ, то $h(L) = \{h(w) \mid w$ принадлежит $L\}$
  - Пример: для языка $L(10^*1)$ и нашего h, $h(L) = (ab)^*$

Замкнутость РЯ

- Гомоморфизм можно применять в обратном направлении (**обратный гомоморфизм**)
- Пусть h – это гомоморфизм над алфавитом Σ в строки, заданные в другом алфавите T
- Пусть L – язык в алфавите T
- Тогда $h^{-1}(L)$, читаемое как «обратное h от L», - это множество строк w из Σ^*, для которых $h(w)$ принадлежит L

Замкнутость РЯ

- Пусть L – язык РВ (00+1)*, т.е. все строки из 0 и 1, где нули встречаются парами, и пусть h – это гомоморфизм $h(a) = 01$, $h(b)=10$
- Тогда $h^{-1}(L)$ – это язык РВ $(ba)^*$
- **Теорема 5.9.** Если L – РЯ в заданном алфавите, и h – гомоморфизм на этом алфавите, то язык $h(L)$ – также РЯ
- **Теорема 5.10.** Если h – гомоморфизм из алфавита Σ в алфавит T, L – РЯ в алфавите T, то язык $h^{-1}(L)$ – также РЯ

Лемма о разрастании РЯ

Теорема 5.11 («Лемма о разрастании для РЯ»). Пусть L – РЯ, и существует константа n, для которой каждую строку w из L, удовлетворяющую неравенству $|w| \geq n$, можно разбить на три строки $w = xyz$ так, что выполняются условия:

1. $y \neq \varepsilon$.
2. $|xy| \leq n$.
3. Для любого $k \geq 0$ строка xy^kz также принадлежит L.

Это значит, что всегда можно найти такую строку y недалеко от начала строки w, которая может разрастись. Если строку y повторить любое количество раз или удалить ее (k=0), то результирующая строка все равно будет принадлежать языку L.

Доказательство. Пусть L – РЯ, тогда L=$L(A)$ для некоторого A. Пусть A имеет n состояний. Рассмотрим произвольную строку w длиной не менее n, например, $w = a_1a_2\ldots a_m$, где $m \geq n$ и каждый a_i есть входной символ. Для $i = 0,1,\ldots,n$ определим состояние p_i как $\hat{\delta}(q_0, a_1a_2...a_i)$, причем p_0=q_0.

Рассмотрим n+1 состояний p_i при $i = 0,1,\ldots,n$. Поскольку у КА n различных состояний, то всегда найдутся два разных числа i и j ($0 \leq i < j \leq n$) при которых p_i=p_j.

Теперь разобьем строку w на xyz.

1. $x = a_1a_2\ldots a_i$
2. $y = a_{i+1}a_{i+2}\ldots a_j$
3. $z = a_{j+1}a_{j+2}\ldots a_m$

Таким образом, x приводит КА в состояние p_i, y – из p_i обратно в p_i, а z – остаток строки w. Строка x может быть пустой при i=0, строка z – при $j = n = m$. А вот y не может быть пустой строкой из-за строгого неравенства.

Что же происходит, когда на вход поступает строка xy^kz для любого неотрицательного k. При k=0 наш автомат переходит из q_0 в p_i, прочитав x. Поскольку $p_i = p_j$, то z переводит A из p_i в заключительное состояние.

Если $k < 0$, то по x автомат переходит из q_0 в p_i, затем, читая y, он k раз циклически проходит через p_i , а затем по z переходит в заключительное состояние. Иначе говоря, для любого неотрицательного k строка xy^kz также принимается автоматом A, т.е. принадлежит языку L.

Дополнительные источники

- Гилл, А. Введение в теорию конечных автоматов / А. Гилл. – М.: Наука, 1966. – 272 с.
- Кузнецов, А.С. Теория вычислительных процессов [Текст] : учеб. пособие / А. С. Кузнецов, М. А. Русаков, Р. Ю. Царев ; Сиб. федерал. ун-т. - Красноярск: ИПК СФУ, 2008. – 184 с.
- Короткова, М.А. Математическая теория автоматов. Учебное пособие / М.А. Короткова. – М.: МИФИ, 2008. – 116 с.
- Молчанов, А. Ю. Системное программное обеспечение. 3-е изд. / А.Ю. Молчанов. – СПб.: Питер, 2010. – 400 с.
- Регулярная грамматика - http://ru.wikipedia.org/wiki/Регулярная_грамматика

Дополнительные источники

- Теория автоматов / Э. А. Якубайтис, В. О. Васюкевич, А. Ю. Гобземис, Н. Е. Зазнова, А. А. Курмит, А. А. Лоренц, А. Ф. Петренко, В. П. Чапенко // Теория вероятностей. Математическая статистика. Теоретическая кибернетика. — М.: ВИНИТИ, 1976. — Т. 13. — С. 109–188. — URL http://www.mathnet.ru/php/getFT.phtml?jrnid=intv&paperid=28&what=fullt&op
- Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. Теория и реализация языков программирования — М.: МЗ-Пресс, 2006 г., 2-е изд. - http://trpl7.ru/t-books/TRYAP_BOOK_Details.htm
- Введение в схемы, автоматы и алгоритмы - http://www.intuit.ru/studies/courses/1030/205/info