Семейство лемм о накачке.
Нормальная форма Грейбах.
Теорема Хомского–Шутценберже

Теория формальных языков

2022 г.

Связь КС-грамматик и АПГ

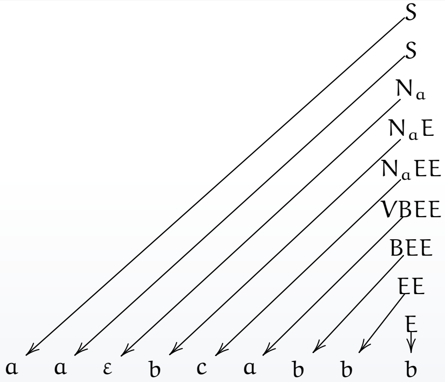
Рассмотрим язык сентенциальных форм, но только без учёта терминальных символов. Получим алфавитную префиксную грамматику (АПГ, см. лекцию 4). При этом каждое применение правила переписывания такой грамматики выбрасывает не более чем один терминальный символ слева, а стирающее правило — ровно один.

Рассмотрим КС-грамматику, соответствующую ей АПГ и путь вывода слова, получаемый с помощью АПГ.

$$
\begin{array}{lll@{\qquad}lll}
S & \to & aS \mid bS \mid aN_a \mid bN_b & S & \to & S \mid N_a \mid N_b \\
N_a & \to & N_aE \mid bN_aE \mid cVB & N_a & \to & N_aE \mid VB \\
N_b & \to & aN_bE \mid bN_bE \mid VA & N_b & \to & N_bE \mid VA \\
V & \to & aV \mid bV \mid a \mid b & V & \to & V \mid \varepsilon \\
\multicolumn{2}{l}{A \to a} & B \to b \quad E \to a \mid b & \multicolumn{3}{l}{A \to \varepsilon \quad B \to \varepsilon \quad E \to \varepsilon}
\end{array}
$$

Связь КС-грамматик и АПГ

S	$\rightarrow$	$aS \mid bS \mid aN_a \mid bN_b$	S	$\rightarrow$	$S \mid N_a \mid N_b$
N_a	$\rightarrow$	$N_aE \mid bN_aE \mid cVB$	N_a	$\rightarrow$	$N_aE \mid VB$
N_b	$\rightarrow$	$aN_bE \mid bN_bE \mid VA$	N_b	$\rightarrow$	$N_bE \mid VA$
V	$\rightarrow$	$aV \mid bV \mid a \mid b$	V	$\rightarrow$	$V \mid \varepsilon$
$A \rightarrow a$		$B \rightarrow b \quad E \rightarrow a \mid b$	$A \rightarrow \varepsilon$		$B \rightarrow \varepsilon \quad E \rightarrow \varepsilon$



Основная теорема

Следствие теоремы Турчина

Пусть G — алфавитная префиксная грамматика, в которой N правил с непустой правой частью, и максимальная длина правой части правила равна M. Тогда любая последовательность порождаемых ею слов

$a_1 \dots a_n$

$\dots$

ε

длиной не менее $N^M \cdot (n+1)$ содержит пару вида $\tau_1 = \Phi\Theta_0$, $\tau_2 = \Phi\Psi\Theta_0$, такую, что $|\Phi| \leqslant M$ и на отрезке $[\tau_1, \tau_2]$ нет слов длины меньше $|\Theta_0| + 1$.

Первая лемма о накачке

Классическая лемма о накачке

Пусть G — КС-язык. Тогда существует $p \in \mathbb{N}$ такое, что любое слово $w \in L(G)$ длины не меньше p имеет представление вида $x_1 y_1 z y_2 x_2$, где $|y_1 y_2| \geqslant 1$, $|y_1 z y_2| \leqslant p$, и все слова вида $x_1 y_1^k z y_2^k x_2$ также принадлежат $L(G)$.

Доказательство: при левостороннем разборе выбираем самую последнюю пару сентенциальных форм.

Вторая лемма о накачке

Пусть G — КС-язык. Тогда существует $p \in \mathbb{N}$ такое, что любое слово $w \in L(G)$ длины не меньше p имеет представление вида $x_1 y_1 z y_2 x_2$, где $|y_2| \geqslant 1$, $|x_1 y_1| \leqslant p$, $|y_2| \leqslant p$ либо y_2 накачивается отдельно, $|x_2| \leqslant p$ либо x_2 накачивается отдельно, и все слова вида $x_1 y_1^k z y_2^k x_2$ также принадлежат $L(G)$.

Варианты лемм о накачке

- Хотелось бы сдвигать начало отрезка накачки вперёд на любое константное количество букв, аналогично — конец отрезка накачки (используя свойство реверса).
- Понятие накачки может быть применено рекурсивно к некоторым достаточно длинным подсловам выбранного слова. Т.е. можно выкидывать из слова подслова, накачиваемые отдельно, без риска выйти из языка.

Для первого нужна гарантия того, что если порождается k букв слова, то длина сентенциальных форм увеличивается не более чем в $f(k)$ раз (где f — хотя бы полином).

Бонус: лемма Огдена

Пусть L — КС-язык. Тогда существует такое число n, что в любом слове w, $|w| \geqslant n$, при отметке n или более букв, w представляется в виде $x_1 y_1 z y_2 x_2$, причём либо во всех трех из x_1, y_1, z есть отмеченные буквы, либо они есть во всех трех из z, y_2, x_2, в слове $y_1 z y_2$ отмечено не более n букв, и $\forall k (x_1 y_1^k z y_2^k x_2 \in L)$.

Исследуем «плохой» язык $\{a^m b^n c^n d^n \mid m > 0\} \cup \{b^i c^j d^k\}$ с помощью леммы Огдена. Абеляр (антагонист) выбирает n. Элоиза (т.е. мы) строит слово $ab^{2n}c^{2n}d^{2n}$ и отмечает n последних букв d. Абеляр может разбить слово $ab^{2n}c^{2n}d^{2n}$ на $x_1 y_1 z y_2 x_2$ двумя способами:

- отмечены x_1, y_1, z, накачиваться может только d^{2n}.
- отмечены x_2, y_2, z, накачивается либо d^{2n}, либо d^{2n} совместно с c^{2n}, b^{2n} или a.

Оба типа накачки выводят из языка, поскольку при любой положительной накачке число вхождений букв b или c расходится с числом вхождений d в слово.

Н.ф. Хомского и левосторонний вывод

- Могут быть непродуктивные левосторонние цепочки: $A \to AB \to \ldots AB^n \to \ldots$.
- Есть гарантия роста слова при развёртке, но нет определённости, по какому префиксу.

Нормальная форма Грейбах

Определение

Грамматика G ($\varepsilon \notin \mathsf{L}(\mathsf{G})$) находится в GNF (н.ф. Грейбах) $\Leftrightarrow$ каждое её правило имеет вид $A_i \to a_j\alpha$, где $A_i \in \mathsf{N}$, $\alpha \in \mathsf{N}^*$, $a_j \in \Sigma$.

- Левосторонний разбор по грамматике в GNF на каждом шагу переписывания порождает терминальный символ.

Нормальная форма Грейбах

Определение

Грамматика G ($\varepsilon \notin L(G)$) находится в GNF (н.ф. Грейбах) $\Leftrightarrow$ каждое её правило имеет вид $A_i \to a_j\alpha$, где $A_i \in N$, $\alpha \in N^*$, $a_j \in \Sigma$.

- Левосторонний разбор по грамматике в GNF на каждом шагу переписывания порождает терминальный символ.
- Для приведения к GNF нужно «вытащить из рекурсии» возможные first-терминалы, порождаемые нетерминалами грамматики.

Нормальная форма Грейбах

Определение

Грамматика G ($\varepsilon \notin L(G)$) находится в GNF (н.ф. Грейбах) $\Leftrightarrow$ каждое её правило имеет вид $A_i \to a_j\alpha$, где $A_i \in N$, $\alpha \in N^*$, $a_j \in \Sigma$.

- Левосторонний разбор по грамматике в GNF на каждом шагу переписывания порождает терминальный символ.
- Для приведения к GNF нужно «вытащить из рекурсии» возможные first-терминалы, порождаемые нетерминалами грамматики.
    - явно найти все завершающиеся цепочки вывода;

Нормальная форма Грейбах

Определение

Грамматика G ($\varepsilon \notin L(G)$) находится в GNF (н.ф. Грейбах) $\Leftrightarrow$ каждое её правило имеет вид $A_i \to a_j\alpha$, где $A_i \in N$, $\alpha \in N^*$, $a_j \in \Sigma$.

- Левосторонний разбор по грамматике в GNF на каждом шагу переписывания порождает терминальный символ.
- Для приведения к GNF нужно «вытащить из рекурсии» возможные first-терминалы, порождаемые нетерминалами грамматики.
  - явно найти все завершающиеся цепочки вывода;
  - рассмотреть язык-реверс сентенциальных форм.
- По умолчанию считаем, что к GNF приводится CNF (н.ф. Хомского).

Первый способ приведения к GNF

1. Нумеруем нетерминалы в правых частях правил в порядке их вхождения;
2. (по исчерпанию по i, начиная с $i = 1$) Если имеется правило вида $A_i \to B_j\beta$, где $j < i$, тогда подставляем вместо B_j все правые части α_k правил вида $B_j \to \alpha_k$.
3. Если после этого все правила имеют вид либо $A_i \to a\alpha$, $a \in \Sigma$, либо $A_i \to B_j\beta$, причём $i < j$, тогда GNF получается последовательной развёрткой B_j.

Первый способ приведения к GNF

1. Нумеруем нетерминалы в правых частях правил в порядке их вхождения;
2. (по исчерпанию по i, начиная с $i = 1$) Если имеется правило вида $A_i \to B_j\beta$, где $j < i$, тогда подставляем вместо B_j все правые части α_k правил вида $B_j \to \alpha_k$.
3. Если после этого все правила имеют вид либо $A_i \to a\alpha$, $a \in \Sigma$, либо $A_i \to B_j\beta$, причём $i < j$, тогда GNF получается последовательной развёрткой B_j. Существует лексикографический порядок на функциональных символах из N.

Первый способ приведения к GNF

1. Нумеруем нетерминалы в правых частях правил в порядке их вхождения;
2. (по исчерпанию по i, начиная с $i = 1$) Если имеется правило вида $A_i \to B_j \beta$, где $j < i$, тогда подставляем вместо B_j все правые части α_k правил вида $B_j \to \alpha_k$.
3. Если после этого все правила имеют вид либо $A_i \to a\alpha$, $a \in \Sigma$, либо $A_i \to B_j \beta$, причём $i < j$, тогда GNF получается последовательной развёрткой B_j. Существует лексикографический порядок на функциональных символах из N.
4. Если есть правила вида $A_i \to A_i \alpha$, тогда устраняем левую рекурсию.
5. Увеличиваем i, если ещё остались нетерминалы, не приведённые к ГНФ.

Устранение левой рекурсии

1. Предположим, для A_i нашлось n леворекурсивных правил и m упорядоченных лексикографически:

$$\begin{array}{ll} A_i \to A_i\alpha_1 & A_i \to \beta_1 \\ \dots & \dots \\ A_i \to A_i\alpha_n & A_i \to \beta_m \end{array}$$

2. Вводим новый нетерминал A_i' такой, что его вес меньше всех прочих, и заменяем правила на:

$$\begin{array}{ll} A_i' \to \alpha_1 A_i' \mid \alpha_1 & A_i \to \beta_1 \mid \beta_1 A_i' \\ \dots & \dots \\ A_i' \to \alpha_n A_i' \mid \alpha_n & A_i \to \beta_m \mid \beta_m A_i' \end{array}$$

3. После всех таких замен грамматика лексикографически упорядочена по левому разбору, и GNF получается последовательной левой развёрткой.

Второй способ приведения к GNF

Алгоритм Блюма–Коха (1999).

Неформальное описание

- Рассмотрим язык сентенциальных форм с переписыванием только по левому разбору. Он регулярен, и в конечное состояние его НКА ведут стрелки, помеченные терминалами.
- Для такого языка легко построить инверсный $\Rightarrow$ множество терминалов-префиксов, которые может породить данный нетерминал.

Второй способ: порождение НКА

1. По каждому нетерминалу B строим автомат $M_B = \langle N_B \cup \{S_B\}, \Sigma \cup N, B_B, \{S_B\}, \delta\rangle$ (S_B — новое состояние, N_B — множество нетерминалов CFG, индексированное нетерминалом B). Правила перехода δ:
   - $\langle C_B, E, M\rangle \Leftrightarrow M = \{D_B \mid C \to DE \in P\}$;
   - $\langle C_B, a, \{S_B\}\rangle \Leftrightarrow C \to a \in P$.
2. Строим реверс к M_B, получаем НКА M_B^R.
3. Строим грамматику $G_B' = \langle N_B \cup \{S_B\}, \Sigma \cup N, R_B', S_B\rangle$ для M_B^R с правилами переписывания:
   - $S_B \to aC_B \Leftrightarrow \langle S_B, a, C_B\rangle \in \delta^R$ и $C_B \neq B_B$ либо из B_B есть стрелки в M_B^R;
   - $S_B \to a \Leftrightarrow \langle S_B, a, B_B\rangle \in \delta^R$;
   - $D_B \to EC_B \Leftrightarrow \langle D_B, E, C_B\rangle \in \delta^R$ и $C_B \neq B_B$ либо из B_B есть стрелки в M_B^R;
   - $C_B \to E \Leftrightarrow \langle C_B, E, B_B\rangle \in \delta^R$.

Окончание конструкции

Теперь по всем G'_i строим окончательный вариант грамматики $G_B = \langle N_B \cup \{S_B\}, \Sigma, R_B, S_B \rangle$ с правилами:

- $S_B \to aC_B,\ S_B \to aC_B \in R'_B$;
- $S_B \to a\ S_B \to a \in R'_B$;
- $D_B \to \alpha C_B \Leftrightarrow D_B \to EC_B \in R'_B$ & $S_E \to \alpha$ (по всем таким α и E);
- $D_B \to \alpha \Leftrightarrow D_B \to E \in R'_B$ & $S_E \to \alpha$ (по всем таким α и E).

Грамматика $\bigcup\limits_{i \in N} G_i$ со стартовым символом S_S — это искомая GNF для исходной грамматики G.

Последние два правила разворачивают неразмеченные нетерминалы в стартовые правила грамматик их сентенциальных форм, поэтому автоматы M_B имеет смысл строить только для тех нетерминалов, которые встречаются в исходной грамматике не только первыми в правых частях правил.

Пример преобразования грамматики по Блюму–Коху

Привести к GNF грамматику некорректных сумм двоичных чисел (почему некорректных?)

$$S \to S + S \mid \mathsf{D} \qquad \mathsf{D} \to \mathsf{D0} \mid \mathsf{D1} \mid \mathsf{1} \mid (S)$$

Пример преобразования грамматики по Блюму–Коху

Привести к GNF грамматику некорректных сумм двоичных чисел (почему некорректных?)

$$S \to S + S \mid D \qquad D \to D0 \mid D1 \mid 1 \mid (S)$$

Сначала избавляемся от цепного правила $S \to D$. Потом строим порождающую структуру A_V сентенциальных форм по левостороннему разбору с финальным состоянием N_V и стартовым V_V. Каждому нетерминалу V соответствует своя структура.

Для A_S : $\quad S_S \xrightarrow{+S} S_S \quad S_S \xrightarrow{0} D_S \quad S_S \xrightarrow{1} D_S \quad S_S \xrightarrow{1} N_S$

$S_S \xrightarrow{(S)} N_S \quad D_S \xrightarrow{0} D_S \quad D_S \xrightarrow{1} D_S \quad D_S \xrightarrow{1} N_S \quad D_S \xrightarrow{(S)} N_S$

Пример преобразования грамматики по Блюму–Коху

Привести к GNF грамматику некорректных сумм двоичных чисел (почему некорректных?)

$$\mathsf{S} \to \mathsf{S} + \mathsf{S} \,|\, \mathsf{D} \qquad \mathsf{D} \to \mathsf{D0} \,|\, \mathsf{D1} \,|\, \mathsf{1} \,|\, (\mathsf{S})$$

Сначала избавляемся от цепного правила $\mathsf{S} \to \mathsf{D}$. Потом строим порождающую структуру $\mathcal{A}_{\mathsf{V}}$ сентенциальных форм по левостороннему разбору с финальным состоянием N_{V} и стартовым V_{V}. Каждому нетерминалу V соответствует своя структура.

Для $\mathcal{A}_{\mathsf{S}}$: $\quad \mathsf{S}_{\mathsf{S}} \xrightarrow{+\mathsf{S}} \mathsf{S}_{\mathsf{S}} \quad \mathsf{S}_{\mathsf{S}} \xrightarrow{0} \mathsf{D}_{\mathsf{S}} \quad \mathsf{S}_{\mathsf{S}} \xrightarrow{1} \mathsf{D}_{\mathsf{S}} \quad \mathsf{S}_{\mathsf{S}} \xrightarrow{1} \mathsf{N}_{\mathsf{S}}$

$\mathsf{S}_{\mathsf{S}} \xrightarrow{(\mathsf{S})} \mathsf{N}_{\mathsf{S}} \quad \mathsf{D}_{\mathsf{S}} \xrightarrow{0} \mathsf{D}_{\mathsf{S}} \quad \mathsf{D}_{\mathsf{S}} \xrightarrow{1} \mathsf{D}_{\mathsf{S}} \quad \mathsf{D}_{\mathsf{S}} \xrightarrow{1} \mathsf{N}_{\mathsf{S}} \quad \mathsf{D}_{\mathsf{S}} \xrightarrow{(\mathsf{S})} \mathsf{N}_{\mathsf{S}}$

Для $\mathcal{A}_{\mathsf{D}}$: $\quad \mathsf{D}_{\mathsf{D}} \xrightarrow{0} \mathsf{D}_{\mathsf{D}} \quad \mathsf{D}_{\mathsf{D}} \xrightarrow{1} \mathsf{D}_{\mathsf{D}} \quad \mathsf{D}_{\mathsf{D}} \xrightarrow{1} \mathsf{N}_{\mathsf{D}} \quad \mathsf{D}_{\mathsf{D}} \xrightarrow{(\mathsf{S})} \mathsf{N}_{\mathsf{D}}$

Пример преобразования грамматики по Блюму–Коху

Превращаем структуры в праволинейные (меняя местами нетерминалы левых и правых частей правил и стартовые состояния с финальными):

$$\text{Для } A_S: \quad S_S \xrightarrow{+S} S_S \quad D_S \xrightarrow{0} S_S \quad D_S \xrightarrow{1} D_S \quad N_S \xrightarrow{1} S_S$$
$$N_S \xrightarrow{(S)} S_S \quad D_S \xrightarrow{0} D_S \quad D_S \xrightarrow{1} D_S \quad N_S \xrightarrow{1} D_S \quad N_S \xrightarrow{(S)} D_S$$

$$\text{Для } A_D: \quad D_D \xrightarrow{0} D_D \quad D_D \xrightarrow{1} D_D \quad N_D \xrightarrow{1} D_D \quad N_D \xrightarrow{(S)} D_D$$

Извлекаем праволинейные (почти) грамматики. В правой части правила может не быть нетерминалов, если там стоял нетерминал V_V.

$$
\begin{array}{lllll}
\text{q-RLG } G_S: & S_S \to +SS_S & D_S \to 0S_S & D_S \to 1D_S & N_S \to 1S_S \\
N_S \to (S)S_S & D_S \to 0D_S & D_S \to 1D_S & N_S \to 1D_S & N_S \to (S)D_S \\
S_S \to +S & D_S \to 0 & D_S \to 1 & N_S \to 1 & N_S \to (S)
\end{array}
$$

Извлекаем праволинейные (почти) грамматики. В правой части правила может не быть нетерминалов, если там стоял нетерминал V_V.

q-RLG G_S :	$S_S \to +SS_S$	$D_S \to 0S_S$	$D_S \to 1D_S$	$N_S \to 1S_S$
$N_S \to (S)S_S$	$D_S \to 0D_S$	$D_S \to 1D_S$	$N_S \to 1D_S$	$N_S \to (S)D_S$
$S_S \to +S$	$D_S \to 0$	$D_S \to 1$	$N_S \to 1$	$N_S \to (S)$

q-RLG G_D :	$D_D \to 0D_D$	$D_D \to 1D_D$	$N_D \to 1D_D$	$N_D \to (S)D_D$
$D_D \to 0$	$D_D \to 1$	$N_D \to 1$	$N_D \to (S)$	

Извлекаем праволинейные (почти) грамматики. В правой части правила может не быть нетерминалов, если там стоял нетерминал V_V.

q-RLG G_S :	$S_S \to +SS_S$	$D_S \to 0S_S$	$D_S \to 1D_S$	$N_S \to 1S_S$
$N_S \to (S)S_S$	$D_S \to 0D_S$	$D_S \to 1D_S$	$N_S \to 1D_S$	$N_S \to (S)D_S$
$S_S \to +S$	$D_S \to 0$	$D_S \to 1$	$N_S \to 1$	$N_S \to (S)$

q-RLG G_D :	$D_D \to 0D_D$	$D_D \to 1D_D$	$N_D \to 1D_D$	$N_D \to (S)D_D$
$D_D \to 0$	$D_D \to 1$	$N_D \to 1$	$N_D \to (S)$	

Заменяем неразмеченные нетерминальные символы V исходной грамматики на N_V. В данном случае нет правил, в которых неразмеченные нетерминалы стояли бы первыми в правых частях, поэтому достаточно просто заменить их на N_V. Иначе пришлось бы заменять их на все возможные правые части α правил вида $N_V \to \alpha$. Стартовый символ — N_S. GNF почти построена!

Извлекаем праволинейные (почти) грамматики. В правой части правила может не быть нетерминалов, если там стоял нетерминал V_V.
Заменяем неразмеченные нетерминальные символы V исходной грамматики на N_V. В данном случае нет правил, в которых неразмеченные нетерминалы стояли бы первыми в правых частях, поэтому достаточно просто заменить их на N_V. Иначе пришлось бы заменять их на все возможные правые части α правил вида $N_V \to \alpha$. Стартовый символ — N_S. GNF почти построена!

q-GNF для G :	$S_S \to +N_S S_S$	$D_S \to 0S_S$	$D_S \to 1D_S$	$N_S \to 1S_S$
$N_S \to (N_S)S_S$	$D_S \to 0D_S$	$D_S \to 1D_S$	$N_S \to 1D_S$	$N_S \to (N_S)D_S$
$S_S \to +N_S$	$D_S \to 0$	$D_S \to 1$	$N_S \to 1$	$N_S \to (N_S)$

Извлекаем праволинейные (почти) грамматики. В правой части правила может не быть нетерминалов, если там стоял нетерминал V_V.

Заменяем неразмеченные нетерминальные символы V исходной грамматики на N_V. В данном случае нет правил, в которых неразмеченные нетерминалы стояли бы первыми в правых частях, поэтому достаточно просто заменить их на N_V. Иначе пришлось бы заменять их на все возможные правые части α правил вида $N_V \to \alpha$. Стартовый символ — N_S. GNF почти построена!

q-GNF для G :	$S_S \to +N_S S_S$	$D_S \to 0S_S$	$D_S \to 1D_S$	$N_S \to 1S_S$
$N_S \to (N_S)S_S$	$D_S \to 0D_S$	$D_S \to 1D_S$	$N_S \to 1D_S$	$N_S \to (N_S)D_S$
$S_S \to +N_S$	$D_S \to 0$	$D_S \to 1$	$N_S \to 1$	$N_S \to (N_S)$

Осталось обернуть в delay-нетерминалы терминальные символы правых частей правил, кроме первого. Здесь это символ).

Теорема Хомского–Шутценберже

Пусть PAREN_n — язык из $4 * n$ элементов $\{[_1,]_1, \ldots, [_n,]_n, (_1,)_1, \ldots, (_n,)_n\}$.

Теорема

Любой CF-язык получается гомоморфизмом из языка $L' = \mathrm{PAREN}_n \cap R$, где R — регулярный.

Пусть G — грамматика L в нормальной форме Хомского. Пронумеруем правила G и поставим им в соответствие следующие.

Теорема Хомского–Шутценберже

Пусть PAREN_n — язык из $4 * n$ элементов $\{[_1,]_1, \dots, [_n,]_n, (_1,)_1, \dots, (_n,)_n\}$.

Теорема

Любой CF-язык получается гомоморфизмом из языка $L' = \mathrm{PAREN}_n \cap R$, где R — регулярный.

Пусть G — грамматика L в нормальной форме Хомского. Пронумеруем правила G и поставим им в соответствие следующие.

1. Если правило n имеет вид $A \to BC$, тогда порождаем правило $A \to [_n B]_n (_n C)_n$.
2. Если правило n имеет вид $A \to a$, тогда порождаем правило $A \to [_n]_n (_n)_n$.

Свойства языка $L(G')$

- Все $]_n$ строго предшествуют $(_n$.

Свойства языка $L(G')$

- Все $]_n$ строго предшествуют $(_n$.
- Ни одна $)_n$ не предшествует непосредственно левой скобке.

Свойства языка $L(G')$

- Все $]_n$ строго предшествуют $(_n$.
- Ни одна $)_n$ не предшествует непосредственно левой скобке.
- Если правило n — это $A \to BC$, тогда $[_n$ непосредственно предшествует некоторой $[_p$, так же как и $(_n$.

Язык R

$R = \{x \in$
$\{[_j,]_j, (_j,)_j\}^* \mid x$ начинается с $[_n$ для некоторого правила $n :$
$A \to \cdots$ & все $]_n$ предшествуют $(_n\}$.

Язык R

$R = \{x \in \{[_j,]_j, (_j,)_j\}^* \mid x$ начинается с $[_n$ для некоторого правила $n : A \to \cdots$ & все $]_n$ предшествуют $(_n\}$.

Можно убедиться, что $L(G') = R \cap PAREN_n$.

Язык R

$R = \{x \in \{[_j,]_j, (_j,)_j\}^* \mid x$ начинается с $[_n$ для некоторого правила $n : A \to \cdots$ & все $]_n$ предшествуют $(_n\}$.

Можно убедиться, что $L(G') = R \cap PAREN_n$.

Осталось определить h. Если n — нефинальное правило, то $h([_n) = h(]_n) = h((_n) = h()_n) = \varepsilon$. Иначе $h([_n) = a$, для остальных скобок так же.

Значение теоремы Х.-Ш.

Возможно разделить парсинг любого КС-языка на две стадии: лексический анализ (проверка условия R) и разбор правильных скобочных структур.

Замечание: поскольку гомоморфизм h не обязан быть инъективным, разбор ПСП не всегда можно определить однозначно. Пример: $\{\mathsf{a}^n\mathsf{b}^n\} \cup \{\mathsf{a}^n\mathsf{b}^{2n}\}$ (полностью неоднозначность устранить нельзя, т.к. этот язык не является детерминированным). Однако Т.Х.Ш. даёт подсказку, как строить КС-грамматики: надо найти в языке все скрытые «скобочные структуры».

Построение грамматики по Х.-Ш.

Построить КС-грамматику для языка $\{a^n b^m c^k \mid n = 2 * m - k\}$.

Ищем возможную скобочную структуру. Для этого сначала избавимся от вычитания: $n + k = 2 * m$. Значит, буквы a должны балансироваться буквами b справа (т.е. буквы b являются «закрывающими скобками» для a), а буквы c — буквами b слева (т.е. буквы b являются «открывающими» для c). Возможны два случая: n и k оба чётны либо оба нечётны. Построим соответствующие им разбиения: $\{a^{2*n'} b^{n'} b^{k'} c^{2*k'}\}$ и $\{a a^{2*n'} b^{n'} b b^{k'} c^{2*k'} c\}$. Дальнейшее построение грамматики уже очевидно. Заметим, что гомоморфизм подразумевает минимум четыре вида скобок: пара $(_{2a},)_b$, пара $(_b,)_{2c}$, внешняя пара $[_a,]_c$ (для нечётного варианта) и $[_b,]_\varepsilon$ для него же, чтобы породить внутреннюю букву b.

Построение грамматики по Х.-Ш.

Построить КС-грамматику для языка $\{a^n b^m c^k \mid n = 2 * m - k\}$.

Как итог, получаем язык, гомоморфно порождаемый языком Дика над $\{(_{2a},)_b, (_b,)_{2c}, [_b,]_\varepsilon, [_a,]_c\}$ со следующим лексером:

1. До $(_{2a}$ может идти лишь единственная $[_a$.
2. После $)_b$ распознаётся одна $[_b$, если распозналась $[_a$.
3. После $)_b$ или $]_\varepsilon$ не может идти ничего другого, кроме $(_b$ или $]_c$ (последняя — только после $]_\varepsilon$).
4. После $)_{2c}$ не может быть ничего, кроме $)_{2c}$ или $]_c$. Дополнительное условие на существование $[_a$ уже не требуется — оно следует из сбалансированности ПСП.

Конструкция выше отличается от используемой в доказательстве теоремы — в целях экономии, в ней почти нет скобок, гомоморфно отображаемых в пустое слово.

Дополнительный пример

Построить КС-грамматику для $\mathsf{L}_{\neq} = \{w_1 c w_2 \,|\, w_i \in \{\mathsf{a}, \mathsf{b}\}^+ \ \&\ w_1 \neq w_2\}$.

Классический пример грамматики с не-КС дополнением. Чтобы расшифровать неравенство, раскроем его в дизъюнкцию: «слово w_1 короче, чем w_2; либо w_2 короче, чем w_1; либо существует такое i, что w_1 и w_2 различаются в i-й позиции». Здесь условия не взаимоисключающие: достаточно одного из них, чтобы слово принадлежало $\mathsf{L}_{\neq}$, но могут выполняться и два сразу.
Перепишем первое условие: $w_1 c w_1' w_2$, где $|w_2| > 0$ и $|w_1| = |w_1'|$. Очевидно, что «открывающими скобками» будут буквы из w_1, «закрывающими» — из w_1', а «скобки» для w_2 замкнуты на самом w_2. Чтобы обеспечить четыре вида соответствий букв по счёту, придётся ввести четыре пары скобок для w_1 и w_1': $\{(_\mathsf{a},)_\mathsf{a}, (_\mathsf{b},)_\mathsf{b}, [_\mathsf{a},]_\mathsf{b}, [_\mathsf{b},]_\mathsf{a}\}$. И две пары скобок для w_2: $\{\{_\mathsf{a}, \}_\varepsilon, \{'_\mathsf{b}, \}'_\varepsilon\}$ и пара скобок для порождения c: $(_\mathsf{c}$ и $)_\varepsilon$(в нижних индексах — гомоморфные образы скобок).

Дополнительный пример

Построить КС-грамматику для $L_{\neq} = \{w_1 c w_2 \mid w_i \in \{a, b\}^+ \ \&\ w_1 \neq w_2\}$.

Осталось построить регулярные условия. Для языка $w_1 c w_1' w_2$, где $|w_2| > 0$ и $|w_1| = |w_1'|$, их можно описать следующим образом:

1. После $(_a$, $(_b$, $[_a$, $[_b$ всегда идёт либо опять одна из таких скобок, либо $(_c$. Скобка $(_c$ единственна.
2. После $)_\varepsilon$, а также скобок $)_a$, $)_b$, $]_b$, $]_a$, могут идти либо $)_a$, $)_b$, $]_b$, $]_a$, либо фигурные скобки.
3. В каждом слове есть хотя бы одна фигурная скобка. После открывающей фигурной скобки обязательно сразу идёт закрывающая, и после первой встреченной фигурной скобки все остальные скобки — тоже фигурные.

Представление Хомского–Шутценбергера для языка $w_1 w_2 c w_1'$, где $|w_2| > 0$ и $|w_1| = |w_1'|$, строится симметрично.

Дополнительный пример

Построить КС-грамматику для $L_{\neq} = \{w_1 c w_2 \mid w_i \in \{a, b\}^+ \ \&\ w_1 \neq w_2\}$.

Осталось разобрать $\{w_1 t_1 w_2 c w_3 t_2 w_4 \mid |w_1| = |w_3| \ \&\ t_1 \neq t_2\}$. Очевидно, что в нём «открывающими» будут элементы w_1, закрывающими — элементы w_3, t_1 и t_2 — уникальные скобки, отображающиеся в разные элементы алфавита, а w_2 и w_4 закрываются сами собой (как w_2 в предыдущем языке). Для $t_1 + t_2$-скобок назначим пары $[^t_a,]^t_b$ и $[^t_b,]^t_a$, остальные обозначения сохраним те же. Лексер языка:

1. После $(_a, (_b, [_a, [_b$ идёт либо опять одна из таких скобок, либо $[^t_{\_}$-скобка. $[^t_{\_}$ единственна, за ней следует либо $\{_a$, либо $\{'_b$, либо $(_c$.
2. За открывающей фигурной скобкой следует закрывающая, и после первой встреченной фигурной скобки все остальные скобки — тоже фигурные, до конца строки либо до чтения единственной $(_c$.
3. После $)_\varepsilon$, а также скобок $)_a,)_b,]_b,]_a$, могут идти либо $)_a,)_b,]_b$, $]_a$, либо скобка $]^t_{\_}$. За $]^t_{\_}$ следует EOL или $\{_a$ или $\{'_b$.

Чтобы получить прообраз языка $L_{\neq}$, объединим все три лексера.