8. Контекстно-свободные грамматики и языки

Разделы:

- Нормальные формы КСГ
- Лемма о разрастании КСЯ
- Свойства замкнутости КСЯ

Нормальные формы КСГ

- Любой КСЯ без ε порождается грамматикой, все продукции которой имеют вид
  - A -> BC и A -> d
  - A, B, C – нетерминалы, d – терминал
- Если в RHS любой продукции два нетерминала, либо один терминальный символ, то это **нормальная форма Хомского** (НФХ)
- Грамматика с продукциями S->AS | a, A -> SA | b находится в НФХ
- Грамматика S->AS | AAS, A -> SA | aa – не в НФХ
- **СТУДЕНТАМ**: поясните почему она не в НФХ?

Удаление бесполезных символов и продукций

- Для получения НФХ-грамматики нужны три действия:
  - удаление бесполезных символов;
  - удаление ε-продукций;
  - удаление цепных продукций
- Символ X **полезный**, если существует некоторое порождение вида $S \Rightarrow^* \alpha X \beta \Rightarrow^* w$ из VT^*
- Символ X может быть нетерминалом или терминалом, а выводимая строка $\alpha X \beta$ может быть первой или последней в порождении
- Символ X **бесполезный**, если он не является полезным

Удаление бесполезных символов и продукций

- Символ X **порождающий**, если $X =>^* w$ для некоторой терминальной строки w
- Символ X **достижимый**, если существует порождение $S =>^* \alpha X\beta$ для некоторых α и β
- Полезный символ одновременно порождающий и достижимый
- В грамматике с продукциями
  – $S \to AB \mid a$, $A \to b$
- все символы, кроме B являются порождающими

Удаление бесполезных символов и продукций

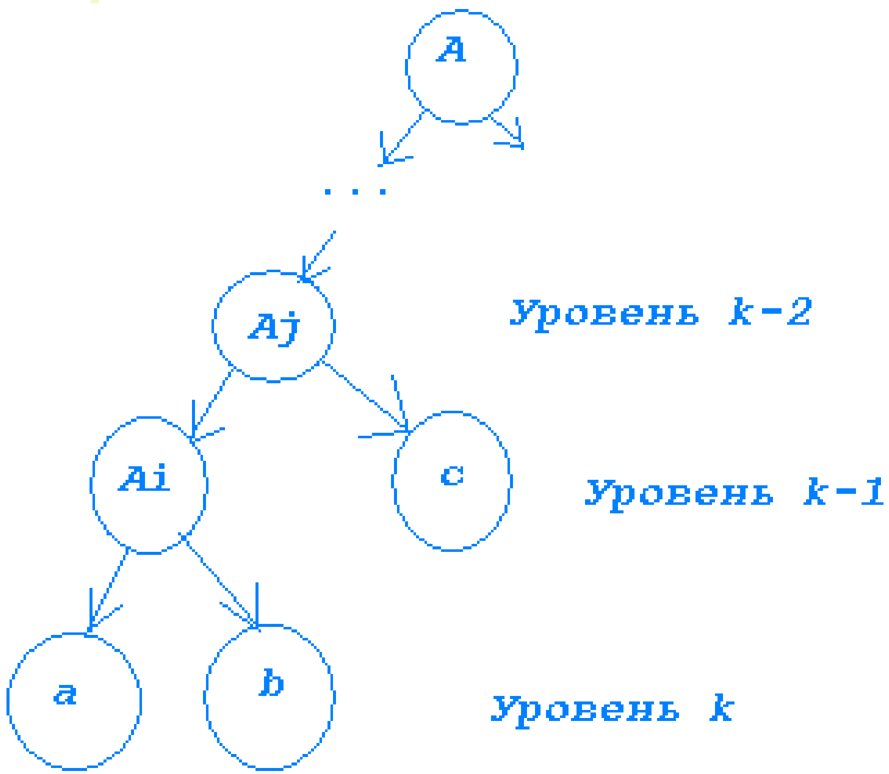
- Продукция **бесполезна**, если в ней есть хотя бы один бесполезный символ
- **Теорема 8.1.** Пусть G – КСГ и $L(G)$ – непустой язык, т.е. эта грамматика порождает хотя бы одну непустую строку
- Тогда существует эквивалентная ей КСГ G_1, не содержащая бесполезных символов или продукций
- **Доказательство**: (по построению)
- Выполняется алгоритм из двух частей (см. следующий слайд)

Удаление бесполезных символов и продукций

- **Доказательство теоремы 8.1.**
- Часть 1. Конструируется промежуточная грамматика $G_2 = (V_{T,2}, V_{N,2}, P_2, S)$, где $V_{N,2}$ содержит такие нетерминалы A, что $A =>^* w$ из T^*

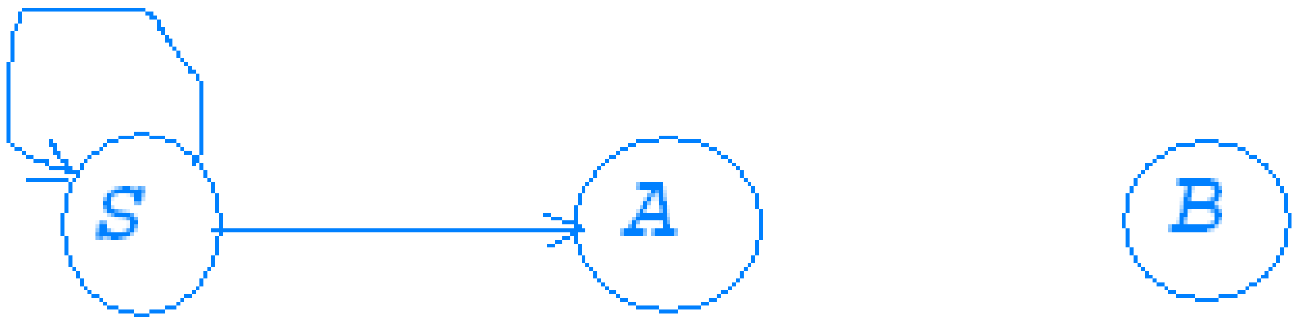
1. Установить $V_{N,2} = \varnothing$
2. Пока не останется символов для попадания в $V_{N,2}$, повторять добавление такого нетерминала A из V_N, у которых продукции имеют следующую форму:
   $A \text{ -> } x1 \ldots x_n$, все x_i из $V_{N,2}$ U V_T
3. Принять P_2 как все продукции из P, чьи символы полностью в $(V_{N,2}$ U $V_T)$

Удаление бесполезных символов и продукций



- Во второй части доказательства теоремы 8.1 из G_2 нужно получить грамматику G_1 без бесполезных символов и продукций, их содержащих
- Для этого строится **граф зависимостей** нетерминалов грамматики G_2

Удаление бесполезных символов и продукций



- Для КСГ граф зависимостей содержит узлы *C* и *D*, помеченные нетерминалами, которые соединяются дугой, если и только если существует продукция вида *C* -> *xDy*

Удаление бесполезных символов и продукций

- **Доказательство теоремы 8.1.** Часть 2

Для всех w из $L(G)$ мы имеем порождение $S =>^* xAy =>^* w$. Поскольку построение G_1 сохраняет нетерминал A и все связанные с ним продукции, то у нас есть все, чтобы с помощью G_1 получить то же самое порождение. Из произвольности нетерминала A и строки w следует $L(G) \subseteq L(G_1)$

Грамматика G_1 получена из G путем удаления продукций, так что $P_1 \subseteq P$. Значит, $L(G_1) \subseteq L(G)$. Объединив оба результата, получаем доказательство эквивалентности исходной и результирующей грамматик.

Удаление ε-продукций

- Если язык L задается КСГ, то L-{ε} имеет КСГ без ε-продукций
- Такой язык - **ε-свободный**
- Продукция вида A -> ε - **ε-продукция**
- Нетерминал A, для которого возможно порождение A =>* ε, называется **допускающим пустоту**, или **ε-порождающим** (*nullable*)
- Пример:
  - Грамматика $S \to aS_1b$, $S_1 \to aS_1b \mid \varepsilon$ генерирует ε-свободный язык $\{a^nb^n : n \geq 1\}$
  - ε-продукцию из КСГ можно удалить после добавления правил, полученных заменой ε на S_1 там, где он появляется справа
  - Результат: $S \to aS_1b \mid ab$, $S_1 \to aS_1b \mid ab$

Удаление ε-продукций

Теорема 8.2. Пусть G – КСГ и $L(G)$ не содержит ε. Тогда существует эквивалентная ей КСГ G_1, не содержащая ε-продукций.

Доказательство. Сначала нужно найти множество NN всех ε-порождающих нетерминалов в G. Для этого выполняются два следующих шага.

1. Для всех продукций $A \text{ -> } \varepsilon$ добавляем A во множество NN.

2. Пока не останется нетерминалов для добавления в NN повторять для всех продукций вида $B \text{ -> } A_1 \ldots A_N$, где $A_1, \ldots, A_N$ есть в NN, положить B в это множество.

Когда NN найдено, мы готовы к конструированию P_1. Чтобы это сделать, мы взглянем на такие продукции в P, которые имеют вид $A \text{ -> } x_1 \ldots x_m$, $m > 0$, для каждого x_i из $(V_N \text{ U } V_T)$. Каждую такую продукцию мы размещаем в P_1 также как все продукции, сгенерированные заменой ε-порождающих нетерминалов с ε во всех возможных комбинациях. Например, если x_i и x_j – ε-порождающие нетерминалы, то в P_1 будет одна продукция с x_i, замененным на ε, одна – с x_j, замененным на ε, и еще одна где оба символа заменены на ε.

Есть одно **исключение из этого правила**: если все x_i – ε-порождающие нетерминалы, то продукция $A \text{ -> } \varepsilon$ не добавляется в множество P_1.

Доказательство эквивалентности грамматик оставляем в качестве самостоятельного упражнения.

Удаление ε-продукций

- **СТУДЕНТАМ**: устраните ε-порождающие нетерминалы из грамматики с продукциями $S \to ABaC$, $A \to BC$, $B \to b \mid \varepsilon$, $C \to D \mid \varepsilon$, $D \to d$

Удаление цепных продукций

- **Цепная продукция** – это продукция вида *A* -> *B*, где и *A*, и *B* являются нетерминалами
- Для удаления можно использовать правило подстановки, сформулированное в виде теоремы
- **Теорема 8.3.** Продукция вида $A \to x_1 B x_2$ может быть удалена из грамматики при условии $A \neq B$, если заменим ее набором продукций, в которых *B* заменяется всеми строками, порождаемыми ею за один шаг

Удаление цепных продукций

- Пусть в грамматике *G* есть продукции *A* -> *a* | *aaA* | *abBc*, *B* -> *abbA* | *b*
- Используя правило подстановки для нетерминала *B*, мы получим грамматику G_1 с продукциями *A* -> *a* | *aaA* | *ababbA* | *abbc*, *B* -> *abbA* | *b*

Удаление цепных продукций

Теорема 8.4. Пусть G – КСГ и $L(G)$ не содержит ε. Тогда существует эквивалентная ей КСГ G_1, не содержащая цепных продукций.

Доказательство. Очевидно, что любая продукция вида A -> A может быть безопасно удалена из грамматики, и нам остается рассматривать случай A -> B. На первый взгляд может показаться, что можно применять теорему 8.3 напрямую, допустив $x_1 = x_2 = \varepsilon$, чтобы заменить A -> B чем-то вроде A -> $y_1 \mid \ldots \mid y_n$. Однако это не всегда дает нужный эффект. Скажем, из продукций A -> B, B -> A нельзя удалить цепные продукции. Чтобы обойти эту проблему, мы должны для всех нетерминалов A найти нетерминалы B такие, что:

$$A \Rightarrow^* B. \qquad (8.1)$$

Мы можем сделать это с помощью графа зависимостей добавлением в него ребра (C, D) всякий раз, когда есть продукция C -> D. Это распространяется и на (8.1), т.к. это выражение хранит путь от A к B. Грамматика G_1 конструируется из G во-первых, копированием в P_1 всех нецепных продукций из P, и во-вторых, для всех A и B, удовлетворяющих условию (8.1), мы добавляем в P_1

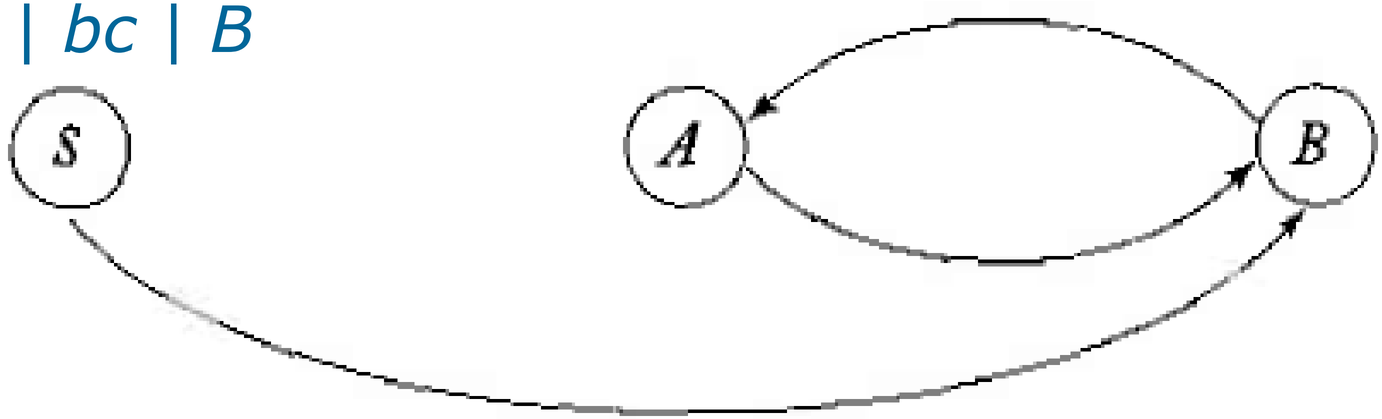
продукции A -> $y_1 \mid \ldots \mid y_n$,

где B -> $y_1 \mid \ldots \mid y_n$ – это набор продукций в P_1, имеющих нетерминал B в LHS. Важно отметить, что эти продукции из P_1, поэтому ни один из y_i не может быть одиночным нетерминалом, так что на последнем шаге не создается ни одной цепной продукции.

Эквивалентность результирующей и исходной грамматик может быть доказана использованием теоремы 8.3.

Удаление цепных продукций

- Удалим все цепные продукции из грамматики с продукциями
  S -> Aa | B, B -> A | bb,
  A -> a | bc | B



- Из ГЗ видно что
  S => A, S =>* B, B =>* A, A =>* B*
- К первоначальным нецепным продукциям
  S -> Aa, B -> bb, A -> a | bc добавлены правила
  S -> a | bc | bb, A -> bb, B -> a | bc
- Результат: *S -> Aa | a | bc | bb,*
  A -> a | bb | bc, B -> a | bb | bc

Удаление цепных продукций

Теорема 8.5. Пусть L – это КСЯ, который не содержит ε. Тогда существует КСГ G такая, $L = L(G)$, и в G нет бесполезных, ε- и цепных продукций.

Доказательство. Процедуры, сформулированные в теоремах 8.4, 8.2 и 8.1, поочередно удаляют все названные типы продукций. Единственный момент, который нуждается в пояснении, заключается в том, что удаление одного типа продукций может приводить к появлению продукций другого типа. Например, устранение ε-продукций может давать в результате цепные продукции. Кроме того, теорема 8.4 требует, чтобы в КСГ не было ε-продукций. С другой стороны, легко проверить, что удаление цепных продукций не создает ε-продукций, а удаление бесполезных продукций не приводит к созданию ε- и цепных продукций. Следовательно, мы можем избавиться от этих нежелательных элементов, используя следующий порядок выполняемых шагов.

1. Удаляются ε- продукции.
2. Удаляются цепные продукции.
3. Удаляются бесполезные продукции.

После третьего шага КСГ не будет содержать ни одну из этих продукций, и теорема доказана.

Удаление цепных продукций

Теорема 8.6. Любая КСГ G для языка L такого, что $L(G)$ не содержит ε, имеет эквивалентную ей КСГ G_{cnf} в НФХ.

Доказательство. По теореме 8.5 мы можем допустить без потери общности, что G не имеет ε- и цепных продукций. Конструирование G_{cnf} можно выполнить за два шага.

Шаг 1. На основе G создадим грамматику $G_1 = (V_T, V_{N,1}, S, P_1)$ путем рассмотрения в P всех продукций вида

$$A \text{ -> } x_1 \ldots x_n, \qquad (8.2)$$

где каждый x_i – это символ из множества терминальных или нетерминальных символов. Если $n = 1$, то x_i является терминалом, т.к. в нашей грамматике нет цепных продукций. В этом случае копируем эту продукцию в P_1. Если n > 1, то вводим новый нетерминал B_a для всех a из V_T. Для каждой продукции в P, имеющей форму (8.2), мы размещаем в P_1 продукцию $A \text{ -> } C_1 \ldots C_n$, где $C_i = x_i$, если последний является нетерминалом, и $C_i = B_a$, если $x_i = a$.

Для всех B_a мы также создаем в P_1 продукции вида $B_a \text{ -> } a$.

В этой части алгоритма будут удалены все терминалы из тех продукций, чьи RHS имеет длину более одного символа. Они заменяются на вновь создаваемые нетерминалы. Иначе говоря, по окончании первого шага будет получена G_1, все продукции которой имеют форму

$$A \text{ -> } a, \qquad (8.3)$$

либо

$$A \text{ -> } C_1 \ldots C_n, \qquad (8.4)$$

где C_i принадлежит $V_{N,1}$.

Это следует из теоремы 8.3, что $L(G_1) = L(G)$.

Шаг 2. В этой части алгоритма мы вводим дополнительные нетерминалы, чтобы уменьшить длину RHS продукций, если это необходимо. Первым делом мы копируем в P_{cnf} все продукции вида (8.3), а также все продукции вида (8.4) для $n = 2$. Если $n > 2$, то вводятся новые нетерминалы $D_1, D_2, \ldots$ и в P_{cnf} вводятся продукции $A \text{ -> } C_1D_1$, $D_1 \text{ -> } C_2D_2$, ..., $D_{n-2} \text{ -> } C_{n-1}C_n$. Очевидно, результирующая грамматика G_{cnf} находится в НФХ. Повторное применение теоремы (8.3) покажет, что $L(G_1) = L(G_{cnf})$, так что $L(G) = L(G_{cnf})$. Этот в какой-то степени неформальный аргумент можно уточнить, но это мы оставляем на самостоятельное изучение.

Нормальная форма Грейбах

- В НФГ ограничения накладываются не на длину RHS, а на позиции символов в ней
- Аргументы, обосновывающие НФГ, немного сложнее и менее прозрачны для понимания, чем НФХ
- КСГ находится в НФГ, если все продукции имеют вид *A* -> *ax*, где *a* – терминал, а *x* – это нетерминал либо *ε*
- Если КСГ не находится в НФГ, то ее можно переписать

Нормальная форма Грейбах

- Преобразуем в НФГ грамматику с продукциями
  $S \text{ -> } AB,\ A \text{ -> } aA \mid bB \mid b,\ B \text{ -> } b$
  - Она не в НФГ
  - По теореме 8.3, мы немедленно получим нормализованную грамматику
    $S \text{ -> } aAB \mid bBb \mid bB,\ A \text{ -> } aA \mid bB \mid b,$
    $B \text{ -> } b$

Нормальная форма Грейбах

- Преобразуем в НФГ грамматику с продукциями
  $S \text{ -> } abSb \mid aa$
  - Воспользуемся теоремой 8.6, и введем новые нетерминалы A и B, которые всего лишь синонимы для терминалов a и b
  - Заменив терминалы, ассоциированными с ними нетерминалами, мы придем к нужному виду грамматики
  - Продукции: $S \text{ -> } aBSB \mid aA$, $A \text{ -> } a$, $B \text{ -> } b$
- **Теорема 8.7.** Любая КСГ G для языка L такого, что $L(G)$ не содержит ε, имеет эквивалентную ей КСГ G_{gnf} в НФГ

Лемма о разрастании КСЯ

Теорема 8.8. Пусть дано дерево разбора, соответствующее КСГ G в НФХ, и пусть кроной дерева является терминальная строка w. Если n – наибольшая длина пути от корня к листьям, то $|w| \leq 2^{n-1}$.

Доказательство. Используем индукцию по n. В базисной части предполагаем, что $n = 1$. Поскольку длина пути отличается от количества ребер на 1, то дерево максимальной длиной пути 1 состоит из корня и листа, отмеченного терминалом. Строка w является этим терминалом, и $|w|= 1$. $2^{n-1} = 2^0 = 1$, значит, базис доказан.

В индуктивной части доказательства предполагаем, что самый длинный путь имеет длину n, и $n > 1$. Корень дерева использует продукцию, которая должна иметь вид A -> BC, поскольку $n > 1$. Ни один из путей в поддеревьях с корнями в B и C не может иметь длину больше n-1, т.к. в этих путях нет ребра от корня к потомку, помеченному B или C. По предположению индукции эти два дерева имеют кроны длины не более 2^{n-2}. Крона всего дерева представляет собой конкатенацию этих двух крон, поэтому имеет длину не более $2^{n-2} + 2^{n-2} = 2^{n-2}$. Доказана и индуктивная часть утверждения.

- **СТУДЕНТАМ**: в этих рассуждениях есть фактическая ошибка. Найдите ее
- Hint: See lecture notes

Лемма о разрастании КСЯ

Теорема 8.9. «Лемма о разрастании для КСЯ». Пусть L – КСЯ, тогда существует такое число n, что если z – произвольная строка из L, длина которой не меньше n, то можно записать $w = uvwxy$, причем выполняются следующие условия.

1. $|vwx| \le n$. Таким образом, средняя часть не слишком длинная.
2. $vx \ne \varepsilon$. По причине того, что v и x подстроки, которые должны разрастись, одна из них не может быть пустой.
3. uv^iwx^iy принадлежит L для всех $i \ge 0$. Две строки v и x могут разрастаться произвольное число раз, включая 0, и полученная при этом строка также будет принадлежать L.

- **СТУДЕНТАМ**: в постановочной части теоремы есть опечатка. Найдите ее.
- Hint: See lecture notes

Лемма о разрастании КСЯ

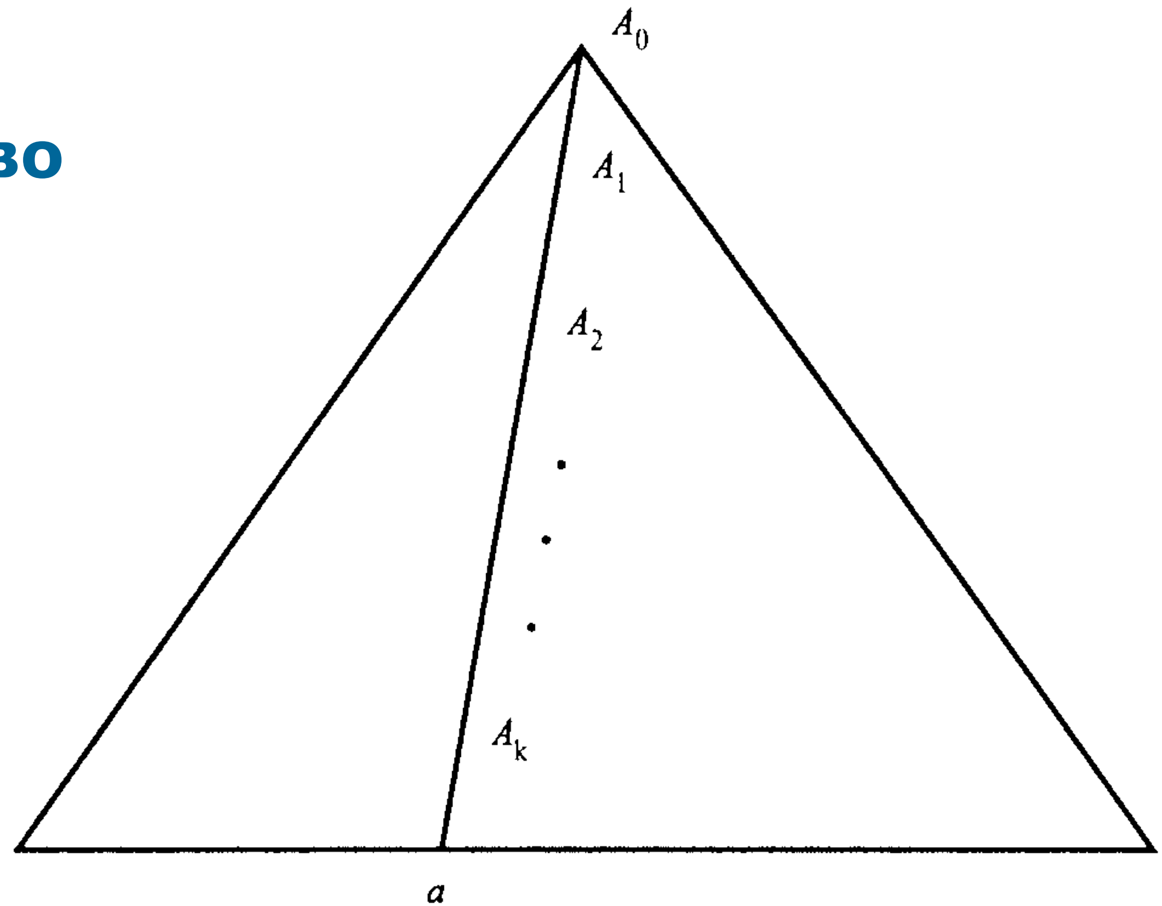
- **Доказательство (начало)**

Сначала для L найдем грамматику G в НФХ. Это невозможно если $L = \{\varepsilon\}$ или $L = \varnothing$. Во втором из этих случаев утверждение леммы не может быть нарушено, т.к. строки z нет в $\varnothing$. НФХ грамматики G в действительности порождает L-$\{\varepsilon\}$, но это не имеет значения, т.к. выбирается $n > 0$, и z не может быть ε.

Итак, пусть в НФХ-грамматика G имеет m нетерминалов и порождает язык $L(G) = L$-$\{\varepsilon\}$. Выбираем $n = 2^m$. Предположим, что z из L имеет длину не менее n. По теореме 8.8 любое дерево разбора, наибольшая длина путей в котором не превышает m, должно иметь крону длиной не более $2^{m-1} = n/2$. Такое дерево разбора не может иметь крону z, т.к. для этого она слишком длинная. Таким образом, любое дерево разбора с кроной z имеет путь длины не менее m+1.

Лемма о разрастании КСЯ

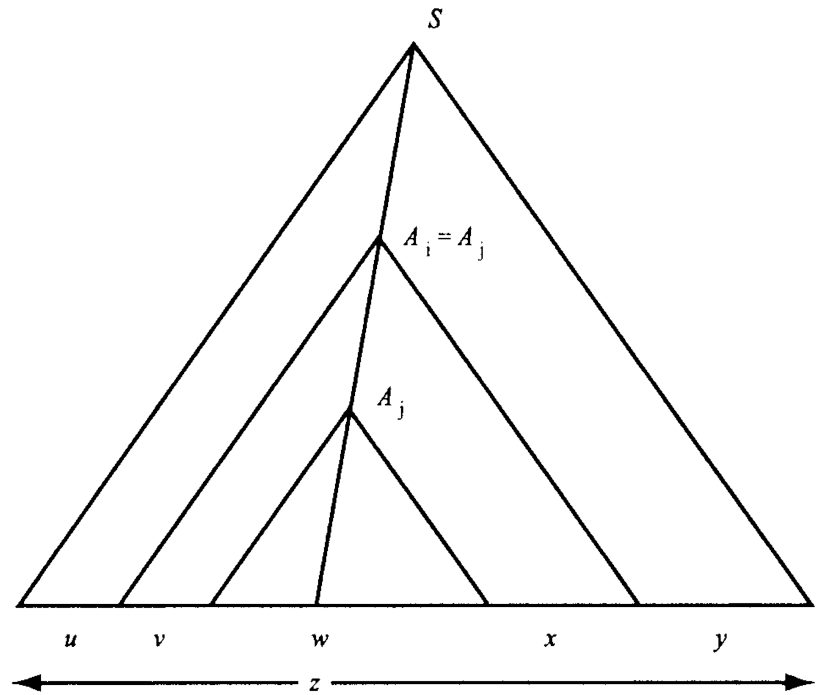
- **Доказательство** (продолжение)



Самый длинный пусть в дереве для z. Его длина равна k+1, где $k \geq m$, поэтому на пути встречается не менее m+1 нетерминалов A_0, A_1, …, A_k. Однако V_N содержит, как мы условились, всего m различных символов, поэтому хотя бы два из m+1 последних нетерминалов на пути от A_{k-m} до A_k (включительно) должны совпадать. Пусть $A_i = A_j$, где k-$m \leq i \leq j \leq k$.

Лемма о разрастании КСЯ

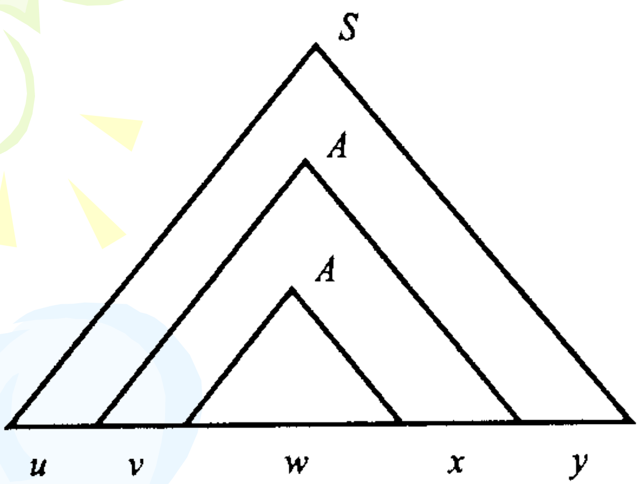
- **Доказательство** (продолжение)
- Дерево можно разделить



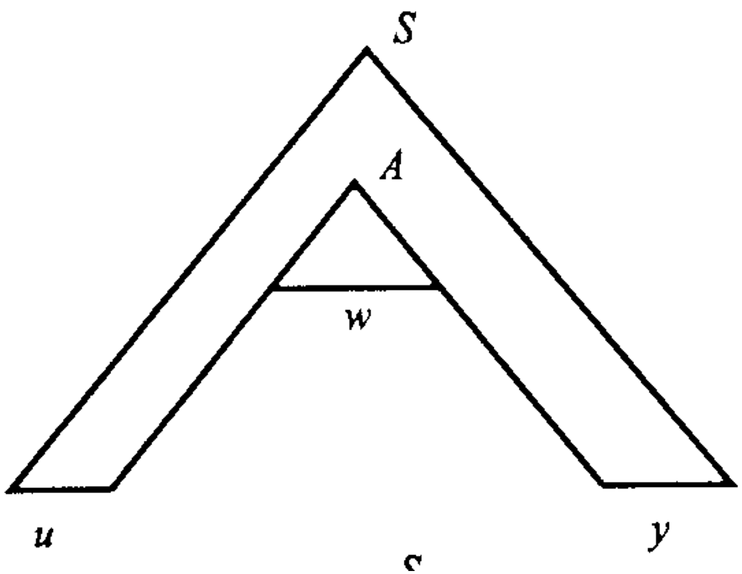
Строка w является кроной поддерева с корнем A_j. Строки v и x – это цепочки соответственно слева и справа от w в кроне большего поддерева с корнем в A_i. В G нет цепных продукций, поэтому v и x не могут быть одновременно пустыми, хотя одна из них может. Цепочки u и y образуют части z, лежащие слева и справа от поддерева с корнем A_i.

Лемма о разрастании КСЯ

- **Доказательство** (продолжение)



а)



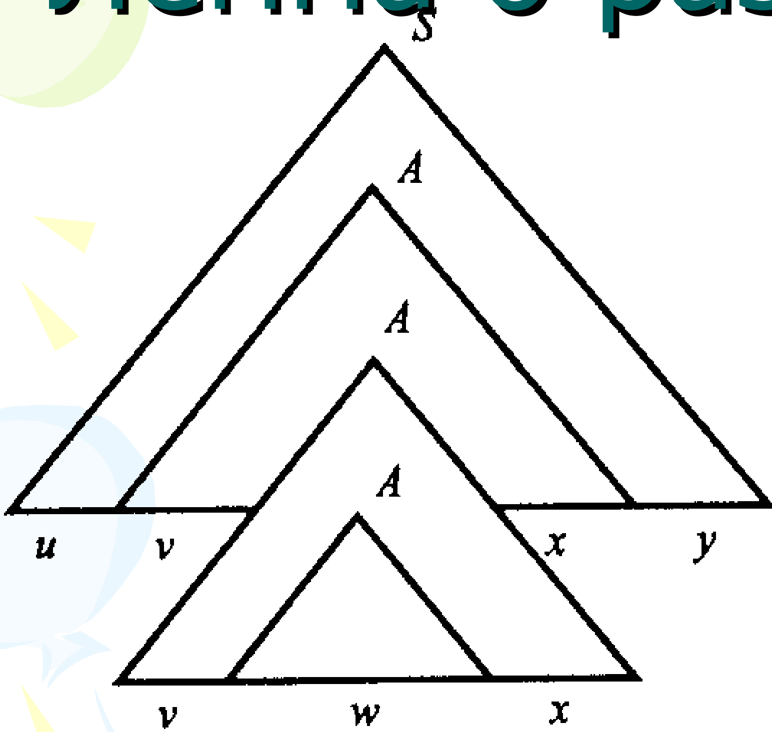
б)

Сначала можно заменить поддерево с корнем A_i, имеющее крону vwx, поддеревом с корнем A_j с кроной w. Это допустимо, т.к. корни помечены одним и тем же символом A. Оно имеет крону и соответствует случаю $i = 0$ в строке uv^iwx^iy.

Лемма о разрастании КСЯ

- **Доказательство** (продолжение)

в)



поддерево с корнем A_j заменено поддеревом с корнем A_i. Это допустимо по той же причине, что и в части (а). Крона поддерева – uv^2wx^2y. Если бы мы потом заменили поддерево с кроной w б льшим поддеревом с кроной vwx, то получили бы дерево с кроной uv^3wx^3y. Этот процесс можно продолжать для любого i.

Лемма о разрастании КСЯ

- **Доказательство** (окончание)

В нашей грамматике имеются деревья разбора для всех строк указанного вида (uv^iwx^iy).

Мы можем также успешно расправиться с условием 1, где $|vwx| \leq n$. Мы выбирали A_i как можно ближе к кроне дерева, поэтому $k\text{-}i \leq m$. Тогда самый длинный путь в поддереве с корнем A_i имеет длину не более m+1. Согласно теореме 8.8 поддерево с корнем A_i имеет крону, длина которой не больше $2^m = n$. Лемма о разрастании доказана.

Лемма о разрастании КСЯ

- Лемму о разрастании КСЯ можно использовать в виде игры с противником
  1. Мы выбираем язык L, желая доказать, что он не КСЯ
  2. Противник выбирает заранее неизвестное n, поэтому мы можем рассчитывать на любое возможное значение
  3. Мы выбираем z и при желании используем n как параметр
  4. Противник разбивает z на 5 частей, соблюдая ограничения $|vwx| \le n$, $vx \ne \varepsilon$
  5. Мы выигрываем, если смогли, выбрав i, показать, что uv^iwx^iy не принадлежит языку L

Лемма о разрастании КСЯ

Пусть $L = \{0^n1^n2^n \mid n \geq 1\}$. Предполагаем, что L – КСЯ, тогда существует n из леммы о разрастании. Выбираем $z = 0^n1^n2^n$.

Наш противник разбивает z на пять частей с соблюдением всех условий. Тогда нам известно, что vwx не может включать одновременно нули и двойки, т.к. последний нуль и первая двойка разделены n+1 позициями. Докажем, что L содержит некоторую строку, которая не может быть в L. Возможны следующие случаи.

1. vwx не имеет двоек, т.е. vx состоит только из 0 и 1 и содержит хотя бы один из этих символов. Тогда строка uwy, которая по лемме должна быть в L, имеет n двоек, но меньше, чем n нулей и единиц. Значит, она не принадлежит L, и в этом случае L – не КСЯ.
2. vwx не имеет нулей. Аналогично, uwy имеет n нулей, но меньше двоек или единиц, поэтому не принадлежит L.

В любом случае мы приходим к выводу, что L содержит строку, которая не может ему принадлежать. Мы пришли к противоречию, которое позволяет заключить, что наше предположение ложно. Следовательно, L не является КСЯ.

Свойства замкнутости КСЯ

- Пусть Σ – это алфавит, и для каждого символа a из алфавита выбран язык L_a
- Выбранные языки могут быть в любых алфавитах, не обязательно одинаковых и не обязательно совпадающих с Σ
- Выбор языков определяет функцию s (**подстановка**) на Σ, и L_a обозначается как $s(a)$ для всех a
- Если $w = a_1 \ldots a_n$ – строка из Σ^*, то $s(w)$ - язык всех строк $x_1 \ldots x_n$, где x_i принадлежит языку $s(a_i)$.
- $s(w)$ - **конкатенация** языков $s(a_1) \ldots s(a_n)$
- $s(L)$ – это объединение $s(w)$ по всем строкам из L

Свойства замкнутости КСЯ

- Пусть $s(0)=\{a^n b^n : n > 0\}$, $s(1) = \{aa, bb\}$
- Подстановка определяется на алфавите $\{0,1\}$
- Пусть $w = 01$, тогда $s(w) = s(0)s(1)$
- Более точно, $s(w)$ состоит из всех строк вида $a^n b^n aa$ и $a^n b^{n+2}$, где $n > 0$
- Пусть $L = L(0^*)$, тогда $s(L) = (s(0))^*$
- Этот набор цепочек вида $a^{n1} b^{n1} \ldots a^{nk} b^{nk}$ для некоторого $k \geq 0$ и произвольной последовательности положительных целых чисел $n_1, \ldots, n_k$

Свойства замкнутости КСЯ

Теорема 8.10. Если L – КСЯ в алфавите Σ, а s – подстановка на Σ, при которой $s(a)$ является КСЯ для каждого a из Σ, то $s(L)$ также является КСЯ.

Доказательство. Дадим общую идею. Берется КСГ для L, и каждый терминал a заменяется аксиомой грамматики для языка $s(a)$. В результате получится единственная КСГ, порождающая $s(L)$.

Пусть грамматика $G = (\Sigma, V_N, P, S)$ задает язык L, а КСГ $G_a = (V_{T,a}, V_{N,a}, P_a, S_a)$ – язык, подставляемый вместо каждого a из Σ. Поскольку для нетерминалов можно выбирать любые имена, то обеспечим непересечение множеств имен символов в V_N и $V_{N,a}$. Цель – гарантировать, что при сборе продукций из разных грамматик в одном множестве случайно не смешаются продукции двух грамматик, и таким образом, получить порождения, невозможные в данных грамматиках.

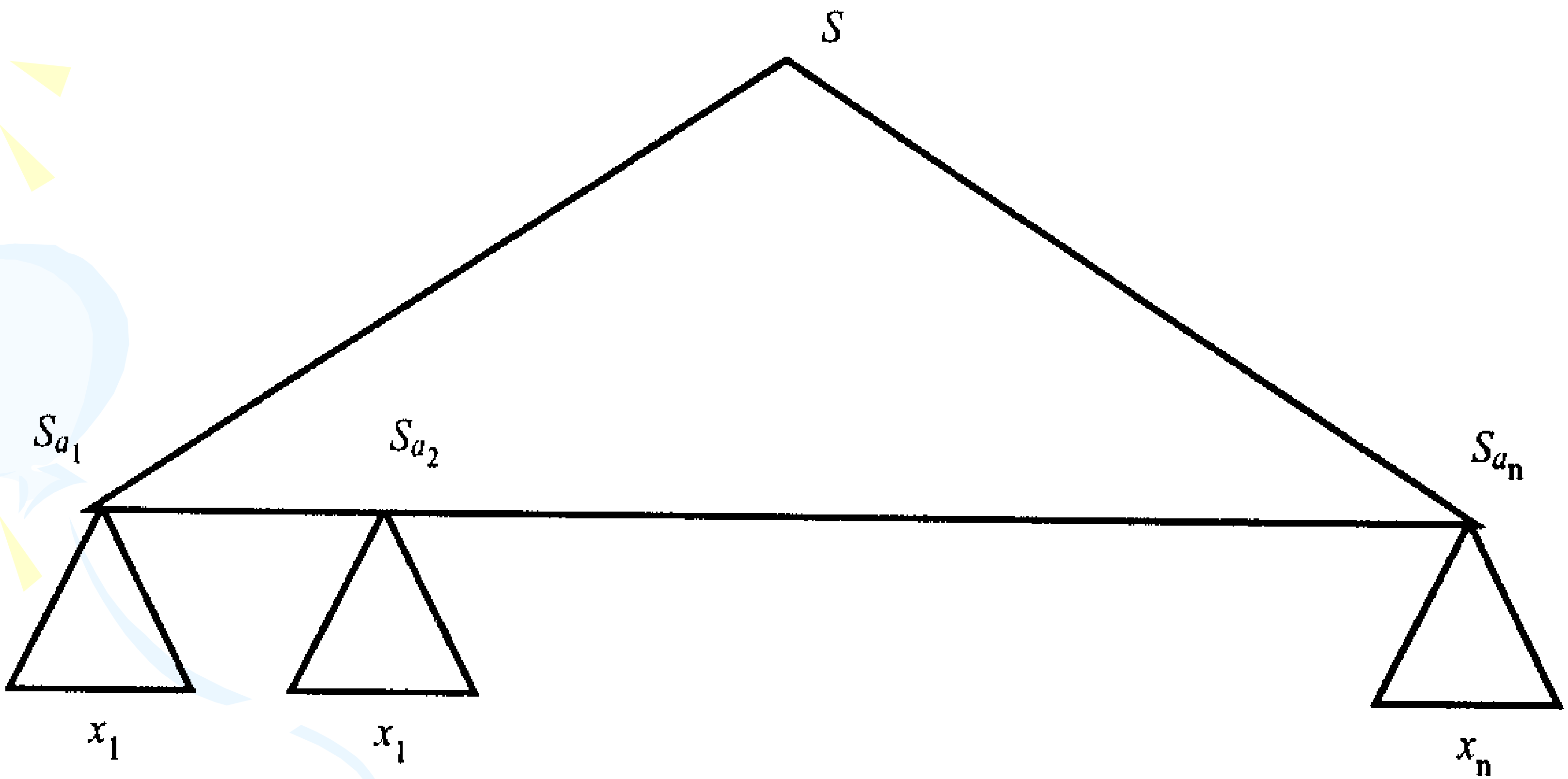
Новая грамматика $G' = (V_T', V_N', P', S)$ для $s(L)$ по следующим правилам.

1. V_N' представляет собой объединение V_N и $V_{N,a}$ по всем a.
2. V_T' является объединением $V_{T,a}$ по a из Σ.
3. P' состоит из всех продукций каждого из P_a для a из Σ и всех продукций P с заменой в их RHS каждого терминала a на S_a.

Таким образом, все деревья разбора в G' начинаются как деревья разбора в G, но вместо порождения кроны в Σ^* он содержат границу, на которой все узлы отмечены нетерминалами S_a вместо a. Каждый такой узел является корнем дерева в G_a, крона которого представляет собой терминальную строку из $s(a)$.

Требуется доказать, что эта конструкция правильна в том смысле, что G' порождает язык $s(L)$. Мы оставим это для самостоятельного изучения.

Свойства замкнутости КСЯ



Свойства замкнутости КСЯ

Теорема 8.11. КСЯ замкнуты относительно операций объединения, конкатенации, замыкания, транзитивного замыкания и гомоморфизма.

Доказательство. Для каждой операции требуется определение соответствующей подстановки и использование теоремы 8.10.

Пусть L_1 и L_2 – КСЯ. Тогда $L_1 \cup L_2$ – является языком $s(L)$, где L – язык $\{1, 2\}$, а s – подстановка, определяемая как $s(1) = L_1$ и $s(2) = L_2$.

L_1L_2 – также является языком $s(L)$, где L – язык $\{12\}$, а s – подстановка, определяемая как $s(1) = L_1$ и $s(2) = L_2$.

Если L_1 – КСЯ, L – язык $\{1\}^*$, а s – подстановка, определяемая как $s(1) = L_1$, то $L_1^* = s(L)$.

Если L_1 – КСЯ, L – язык $\{1\}^+$, а s – подстановка, определяемая как $s(1) = L_1$, то $L_1^+ = s(L)$.

Пусть L – КСЯ над алфавитом Σ, а h – гомоморфизм на алфавите. Пусть s – это подстановка, заменяющая каждый символ a из Σ языком, состоящим из единственной строки $h(a)$. Иначе говоря, $s(a)=\{h(a)\}$ для всех a. Тогда $h(L) = s(L)$.

Свойства замкнутости КСЯ

Теорема 8.12. Если L – КСЯ, то L^R – тоже КСЯ.

Доказательство. Снова даем только общую идею. Пусть $L = L(G)$ для КСГ $G = (V_T, V_N, P, S)$. Построим $G^R = (V_T, V_N, P^R, S)$, где продукции P^R представляют собой обращения продукций из P. Таким образом, если A -> α – продукция в G, то A -> α^R – продукция G^R. Используя индукцию по длине порождений в G и в G^R, нетрудно показать, что $L(G^R) = L^R$. По сути, все выводимые в G^R строки являются обращениями строк, выводимых в G и наоборот.

Теорема 8.13. Если L – КСЯ, а R – РЯ, то их пересечение – КСЯ.

Доказательство. Здесь можно исходить из моделирования КСЯ с помощью МПА, а РЯ – с помощью КА. Они запускаются «параллельно», и в результате должен быть получен новый МПА.

Теорема 8.14. Если L – КСЯ, а h – гомоморфизм. Тогда $h^{-1}(L)$ – КСЯ.

Приводим без доказательства, которое также сводится к моделированию поведения МПА и индукции по количеству переходов, совершаемых исходным и результирующим автоматами.

Дополнительные источники

- Контекстно-свободная грамматика - http://ru.wikipedia.org/wiki/Контекстно-свободная_грамматика
- Короткова, М.А. Математическая теория автоматов. Учебное пособие / М.А. Короткова. – М.: МИФИ, 2008. – 116 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.
- Молчанов, А. Ю. Системное программное обеспечение. 3-е изд. / А.Ю. Молчанов. – СПб.: Питер, 2010. – 400 с.

Дополнительные источники

- 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
- 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

Дополнительные источники

- Нормальная форма Хомского - http://ru.wikipedia.org/wiki/Нормальная_форма_Хомского
- Свойства контекстно-свободных языков - http://www.williamspublishing.com/PDF/978-5-8459-1347-0/part7.pdf
- Контекстно-свободные грамматики - http://www.math.spbu.ru/user/mbk/PDF/Ch-4.pdf