Частичное угадывание сверхсобытий, порожденных простыми $LL(1)$-грамматиками

А. А. Мастихина

Автомат угадывает символ входной последовательности, если он выдает этот символ на выходе в предыдущий момент времени. В работе рассмотрен вопрос, можно ли с помощью детерминированного автомата частично предвосхитить некоторое сверхсобытие, то есть угадать некоторую ненулевую долю символов каждого сверхслова из этого сверхсобытия. Получены критерий и алгоритм, позволяющие это установить для сверхсобытий, образованных сверхитерацией языков, порожденных простыми $LL(1)$-грамматиками.

Ключевые слова: угадывающие автоматы, автоматы с магазинной памятью, частичное угадывание, сверхсобытия, простые $LL(1)$-грамматики.

1. Введение

В статье рассматривается предвосхищение сверхсобытий над алфавитом $\{0, 1\}$ определенного вида с помощью детерминированных автоматов. Понятие автомата с предвосхищением возникло еще в начале развития теории автоматов [1].

Это означает, что автомат использует не только данные, уже поступившие на вход, но и некоторые значения, которые должны поступить в будущем.

В данной работе рассматриваются машины, использующие только информацию, которая уже поступила на вход, и пытающиеся предсказать на выходном канале то, что поступит.

В статье [2] рассматривались конечные автоматы, угадывающие сверхслово или множество сверхслов. Это значило, что через некоторое конечное время автомат начинает угадывать каждый следующий символ, то есть на выходе в момент времени t выдавать элемент входной последовательности с номером $t+1$.

Было показано, что угадываемо только множество периодических сверхслов с фиксированным периодом.

В статье [3] было введено понятие частичного угадывания, которое имеет место, если некоторый автомат угадывает ненулевую долю символов сверхслова.

Вводится степень угадывания как предел отношения числа верно предсказанных символов за время t от начала слова к длине префикса t при t, устремленном к бесконечности.

В зависимости от того, брать ли нижний или верхний предел в случае, если они не равны, меняется смысл задачи. В первом случае угадывание должно происходить регулярно, вне зависимости от того, когда измерять долю угаданных символов. Случай же верхнего предела иллюстрирует ситуацию, когда у угадывающего есть возможность самостоятельно выбирать, когда подсчитывать эту долю. Кстати, в этом случае несущественно, что последовательность бесконечная, так как можно вычислять долю в момент окончания слова. Частичная угадываемость исследуется независимо для случаев верхней степени и нижней степени.

Рассматривается частичная угадываемость множеств сверхслов. Степень угадывания множества определяется как точная нижняя грань степени по всем сверхсловам этого множества. Таким образом, удачным считается случай, если на вход некоторой машины подается неизвестная последовательность из нулей и единиц из известного множества, и машина гарантированно предсказывает некоторую ненулевую долю символов.

В статье [4] был получен критерий частичной угадываемости для общерегулярных сверхсобытий.

Множества, рассматриваемые в данной работе — сверхитерации некоторых контекстно-свободных языков, а именно порожденных простыми $LL(1)$-грамматиками.

Для них получен критерий, определяющий по грамматике, есть ли частичная угадываемость, в случае положительного ответа строится угадывающий алгоритм, реализуемый автоматом с магазинной памятью. Частичная угадываемость рассматривается в смысле верхнего и нижнего пределов доли верно предсказанных символов. Критерии и алгоритмы для этих двух случаев различаются.

Получены оценки доли угадывания, причем они достижимы, то есть для множеств с наименьшей возможной степенью предложенные алгоритмы угадывают именно столько.

Результаты данной работы анонсированы в [6]. Автор выражает благодарность профессору Э. Э. Гасанову за постановку задачи и помощь в работе.

2. Основные понятия и формулировка результатов

Будем использовать следующие обозначения:

- A^* — множество всех слов в алфавите A, по определению будем считать, что пустое слово Λ принадлежит A^*,
- A^∞ — множество всех сверхслов в алфавите A,
- $|a|$ — *длина* слова $a \in A^*$. По определению $|\Lambda| = 0$,
- $a(n)$ — n-ый элемент слова или сверхслова a,
- *префикс* слова или сверхслова a некоторой длины n — $a(1)\dots a(n)$,
- *событие* — подмножество $A^* \setminus \{\Lambda\}$, *сверхсобытие* — подмножество A^∞,
- ab — *конкатенация* слов a и b,
- R_1R_2 — *произведение* события R_1 и события или сверхсобытия R_2, то есть все слова (сверхслова) вида ab, где $a \in R_1, b \in R_2$,
- $a^n = \underbrace{a\dots a}_{n}$, здесь a может быть словом или событием, n — натуральное и может быть равно ∞,
- R^* — *итерация* события R, то есть $R^* = R \cup R^2 \cup R^3 \cup \dots \cup R^i \dots$,

- R^∞ — *сверхитерация* события R, то есть $R^\infty = \{a_1a_2a_3\ldots | a_i \in R, i = 1,2,3,\ldots\}$.

Будем рассматривать класс детерминированных, но, возможно, бесконечных автоматов $\mathfrak{M}$.

Выходное сверхслово некоторого автомата $\mathfrak{A}$ при подаче на его вход сверхслова α будем обозначать через $y_\alpha^\mathfrak{A}$. Детерминированность автомата означает, что после подачи на его вход t первых символов сверхслова α он однозначно выдает некоторый выходной символ $y_\alpha^\mathfrak{A}(t)$.

Автомат $\mathfrak{A}$ *угадывает сверхслово* $\alpha \in \{0,1\}^\infty$ *с (нижней) степенью* $\sigma \in [0,1]$, если

$$c^\mathfrak{A}(\alpha) = \underline{\lim}_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} (1 - |y_\alpha^\mathfrak{A}(i) - \alpha(i+1)|) = \sigma.$$

$\mathfrak{A}$ *угадывает сверхслово* $\alpha \in \{0,1\}^\infty$ *с верхней степенью* $\sigma \in [0,1]$, если

$$\overline{c}^\mathfrak{A}(\alpha) = \overline{\lim}_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} (1 - |y_\alpha^\mathfrak{A}(i) - \alpha(i+1)|) = \sigma.$$

Автомат $\mathfrak{A}$ *частично угадывает* множество сверхслов A (*частично угадывает* в смысле верхнего предела), если для любого $\alpha \in A$ найдется такое $\sigma > 0$, что выполнено $c^{\mathfrak{A}(\alpha)} > \sigma > 0$ ($\overline{c}^\mathfrak{A}(\alpha) > \sigma > 0$). Множество A *частично угадываемо*, если найдется частично угадывающий его автомат $\mathfrak{A}$.

Степень угадывания множества A

$$c^\mathfrak{A}(A) = \inf_{\alpha \in A} c^\mathfrak{A}(\alpha).$$

Грамматикой называется четверка $(N,T,P,S) = G$, где N — множество нетерминальных символов (далее НС), T — множество терминальных символов, причем $N \cap T = \varnothing$, множество правил P — конечное подмножество множества $(N\cup T)^*N(N\cup T)^* \times (N\cup T)^*$, выделенный начальный символ $S \in N$ (аксиома).

Цепочка символов $\delta\alpha\gamma$ из $(N\cup T)^*$ *непосредственно выводима* из $\delta\beta\gamma$ (обозначим $\delta\alpha\gamma \Rightarrow \delta\beta\gamma$), если $\beta \to \alpha \in P$.

Цепочка μ *выводима* из α (обозначим $\alpha \Rightarrow^* \mu$), если существуют такие $\alpha_1, \ldots, \alpha_k$, что $\alpha = \alpha_1 \Rightarrow \alpha_2 \Rightarrow \ldots \Rightarrow \alpha_k = \mu$.

Множество выводимых из аксиомы S цепочек, не содержащих нетерминальных символов, есть *язык, порожденный грамматикой* G (обозначается $L(G)$).

Грамматика называется *контекстно-свободной*, если каждое ее правило имеет вид $A \to \alpha$, где $A \in N, \alpha \in (N \cup T)^*$.

Пусть для каждого нетерминального символа грамматики есть либо одно правило вывода, либо не более $|T|$ правил, таких что

$A \to a_1\alpha_1, A \to a_2\alpha_2, \ldots, A \to a_n\alpha_n$, где $a_i \in T, a_i \neq a_j$ при $i \neq j$, $i, j = 1, \ldots, n, n \leqslant |T|, \alpha_i \in (T \cup N)^*$.

Такая грамматика называется *простой* $LL(1)$*-грамматикой* или *разделенной грамматикой.*

Будем рассматривать множества сверхслов, образованные сверхитерацией контекстно-свободного языка $L(G)$, порожденного простой $LL(1)$-грамматикой.

Грамматика находится в *нормальной форме Грейбах*, если все правила имеют вид $A \to aB_1 \ldots B_k$, где $B_1, \ldots, B_k \in N, a \in T$,или $A \to a$, или $S \to \Lambda$. Причем, если грамматика простая $LL(1)$-, для каждого нетерминального символа не более $|T|$ команд (с разными терминальными символами).

Символ A *достижим из символа* B, если найдутся такие $\alpha \in T^*$, $\beta \in (T \cup N)^*$, что $B \Rightarrow^* \alpha A \beta$.

Символ A грамматики G называется *достижимым*, если найдутся такие $\alpha \in T^*, \beta \in (T \cup N)^*$, что $S \Rightarrow^* \alpha A \beta$. Иначе символ называется *недостижимым.*

Нетерминальный символ A называется *тупиковым*, если не найдется такого слова $\beta \in T^*$, что $A \Rightarrow^* \beta$.

Далее рассматриваем грамматики в нормальной форме Грейбах и считаем, что недостижимые и тупиковые символы отсутствуют. Также считаем, что аксиома не встречается в правой части правил. Будем тогда говорить, что грамматика находится в *нормальной форме.*

Ниже показано, что при приведении грамматики к нормальной форме она остается простой $LL(1)$-грамматикой.

Автомат с магазинной памятью (далее — МП-автомат) — это семерка $(Q, \Sigma, \Gamma, Z, \Phi, q_0, F)$, где Q — множество состояний, выделенное начальное состояние $q_0 \in Q$ и некоторое множество $F \subseteq Q$ заключительных состояний, Σ — входной алфавит, Γ — магазинный алфавит, Φ — множество команд, каждая команда — отображение $(Q \times \Sigma \times \Gamma \to Q \times \Gamma^*)$, Z — символ для обозначения пустого магазина.

Конфигурацией будем называть пару $(q, \xi), q \in Q, \xi \in \Gamma^*$: текущее состояние и содержимое стека, глубиной стека — количество НС в стеке в данный момент.

МП-автомат работает потактово. Автомат детерминированный, то есть из каждой конфигурации под воздействием одной входной буквы выводится не более одной конфигурации.

МП-автомат *допускает язык* L, если только после подачи слов $\alpha \in L$ на начальную конфигурацию (q_0, S) автомат приходит в конфигурацию $(q, Z), q \in F$.

Известно, что язык, порожденный простой $LL(1)$-грамматикой, допускается МП-автоматом с одним состоянием. Построим такой автомат с магазинной памятью для такой грамматики $(N, \{0,1\}, S, P)$.

$Q = \{q\} = F = q_0$

$\Sigma = \{0,1\}, \Gamma = N \cup \{Z\}$, Z — символ для обозначения пустого стека, в начальный момент времени в стеке содержится символ S.

$\Phi = \{(q, a, A \to q, \alpha) : A \to a\alpha \in P\}$

Слово $\alpha \in \{0,1\}^*$ допускается автоматом $(\{q_0\}, \{0,1\}, \Gamma, Z, \Phi)$, если после его подачи на начальную конфигурацию (q_0, S) автомат приходит в конфигурацию (q_0, Z).

Определим, как автомат допускает сверхитерацию языка, порожденного той же грамматикой.

Расширим множество команд Φ. Для каждого правила $S \to a\alpha \in P$, такого что α не пустое слово, добавим команду $(q_0, Z, a \to q_0, \alpha)$ и добавим команду $(q_0, Z, a \to q_0, Z)$ для каждого $S \to a \in P$.

МП-автомат $(\{q_0\}, \{0,1\}, \Gamma, Z, \Phi)$ *допускает сверхитерацию* $L(G)$, если при подаче любого сверхслова из этого множества он оказывается в конфигурации Z бесконечное число раз.

Теорема 1. *Множество сверхслов, образованное сверхитерацией языка* $L(G)$*, порожденного простой* $LL(1)$*-грамматикой* G *в нормальной форме, частично угадываемо в смысле нижней степени* *тогда и только тогда, когда хотя бы для одного нетерминального символа грамматики существует только одно правило вывода.*

Рассмотрим грамматику $G = (N, \{0,1\}, P, S)$ в нормальной форме.

Введем параметры грамматики: число нетерминальных символов $|N| = n$ и L — максимальная длина правила (длиной правила назовем число НС в правой части команды).

Обозначим за $\widetilde{G}(n, L)$ класс простых $LL(1)$-грамматик с этими параметрами.

За $\widetilde{G}^0(n, L)$ обозначим класс таких грамматик из $\widetilde{G}(n, L)$, которые являются частично угадываемыми.

Теорема 2. *Для произвольных $n \geqslant 2, L \geqslant 1$*

$$\inf_{G \in \widetilde{G}^0(n,L)} \sup_{\mathfrak{A} \in \mathfrak{M}} c^{\mathfrak{A}}((L(G))^\infty) = \frac{L-1}{2L^{n-1} - L^{n-2} - 1},$$
$$\inf_{G \in \widetilde{G}^0(1,L)} \sup_{\mathfrak{A} \in \mathfrak{M}} c^{\mathfrak{A}}((L(G))^\infty) = 1.$$

Назовем правило грамматики *регулярным*, если оно имеет вид $A \to aB$ (глубина стека соответствующего МП-автомата не увеличивается), $A, B \in N$. Назовем правило *стирающим*, если правило имеет вид $A \to a$ (глубина стека уменьшается). Остальные правила назовем *удлинняющими*.

НС *B достижим по регулярным правилам по слову α из НС A*, если существуют регулярные правила

$$\begin{aligned}
&A \to \alpha(1)A_1,\\
&A_1 \to \alpha(2)A_2,\\
&\dots\\
&A_{|\alpha|} \to \alpha(|\alpha|)B.
\end{aligned}$$

Если существует такое слово α, то *B достижим из A по регулярным правилам.*

Назовем *регулярной сильно-связной компонентой* $\widetilde{R}$ множество НС, попарно достижимых друг из друга по регулярным правилам.

Выходом из регулярной сильно-связной компоненты $\widetilde{R}$ назовем правило для $A \in \widetilde{R}$, не являющееся регулярным или имеющее вид $A \to aB$, где $B \notin \widetilde{R}$.

Опишем алгоритм $\mathfrak{D}$, разделяющий НС грамматики $G = (N, \{0,1\}, S, P)$ на классы следующим образом.

1-й шаг.

Рассмотрим регулярные сильно-связные компоненты, все выходы из которых являются стирающими, занесем их в класс D_1.

Далее удалим из грамматики НС из класса D_1. То есть преобразуем грамматику G в грамматику $G' = (N', \{0,1\}, S, P')$, где $N' = N \setminus D_1$, а P' получается из P удалением правил, начинающихся с элементов D_1 и вычеркиванием элементов D_1 из правых частей других содержащих их правил.

Например, если $D_1 = \{A\}$, то правило $B \to aAC$ из P в P' будет иметь вид $B \to aC$.

Пусть сделано i шагов.

Опишем $(i+1)$-й шаг.

В грамматике $G^{(i)}$ рассмотрим сильно-связные регулярные компоненты, все выходы из которых являются стирающими. Все НС из таких компонент занесем в класс D_{i+1}.

Удалим из правил грамматики правила, начинающиеся с НС из класса D_{i+1}, а также вычеркнем элементы D_{i+1} из других правил. Получившуюся грамматику назовем $G^{(i+1)}$.

Алгоритм останавливается, когда некоторый класс D_k оказывается пуст или на k-ом шаге объединение классов $\cup_{i=1}^{k} D_i$ есть множество всех НС N.

Теорема 3. *Множество сверхслов $(L(G))^\infty$, где $G = (N, \{0,1\}, S, P)$ — простая $LL(1)$-грамматика в нормальной форме, частично угадываемым в смысле верхней степени тогда и только тогда, когда после работы алгоритма выполнено $\cup_{i=1}^{k} D_i \neq N$, где k — номер шага, на котором алгоритм $\mathfrak{D}$ остановился.*

Обозначим l — минимальная длина правила грамматики.

Пусть $R = \max_{i=0,1,\ldots,k} r_i$, где r_i — максимальное количество НС в регулярной сильно-связной компоненте в грамматике $G^{(i)}$.

Обозначим r_c^i наибольшую длину цепи регулярных правил в грамматике $G^{(i)}$, то есть наибольшее r, такое что существуют $A_1 \to a_1A_2, A_2 \to a_2A_3, \ldots, A_r \to a_rA_{r+1} \in P^{(i)}$, причем $A_1, \ldots, A_{r+1}$ попарно различны.

$R_c = \max_{i=0,\ldots,k} r_c^i$.

Обозначим за $\widetilde{G}(l, L, R_c)$ класс простых $LL(1)$-грамматик с параметрами l, L, R_c.

За $\widetilde{G}^0(l, L, R_c)$ обозначим класс таких грамматик из $\widetilde{G}(l, L, R_c)$, которые являются частично угадываемыми.

Теорема 4. *Для произвольных $L \geqslant 1, R_c, l \geqslant 1$*

$$\inf_{G \in \widetilde{G}(l,L,R_c)} \sup_{\mathfrak{A} \in \mathfrak{M}} \overline{c}^{\mathfrak{A}}((L(G))^\infty) =$$
$$= \min\left\{\frac{l}{l+1}, \frac{1}{R_c+1}, \frac{L-1}{L(R_c(L-1)+L)^k - 1}\right\}.$$

3. Угадывание в смысле нижнего предела

Лемма 1. *Для простой $LL(1)$-грамматики существует грамматика в нормальной форме, порождающая тот же язык и являющаяся простой $LL((1)$-грамматикой.*

Доказательство. Удалим недостижимые и тупиковые НС. Число правил могло только уменьшиться. Введение дополнительного НС вместо аксиомы в правой части также не влияет на число правил для каждого символа.

Покажем, что любая простая $LL(1)$-грамматика при приведении к нормальной форме Грейбах также остается простой $LL(1)$-грамматикой.

Для этого используем алгоритм, описанный в [5].

Введем линейный порядок на множестве НС $N = A_1, \ldots, A_n$: $A_1 < A_2 < \ldots < A_n$, причем если $A \to B\alpha \in P, \alpha \in (T \cup N)^*$, то $A < B$. Это можно сделать, так как в простой $LL(1)$-грамматике не может быть правила $A \to A\alpha$, так как это было бы единственное правило для A, а тогда он тупиковый. Также из определения простой

$LL(1)$-грамматики следует, что невозможен случай $A < B$ и $B < A$ одновременно, так как тогда A и B тупиковые.

Последовательно рассмотрим НС $A_{n-1}, A_{n-2}, \ldots, A_1$.

Если есть правило вида $A_i \to A_j\alpha, j > i$ (оно может быть одно, иначе правила должны начинаться с терминального символа), то заменим его на одно или два правила $A_i \to \beta\alpha$, если $A_j \to \beta$, $\beta \in T(T \cup N)^*$(правила $A_j \to \beta$ удалим). Получившаяся грамматика также является простой $LL(1)$-грамматикой.

После этой процедуры все правила начинаются с терминальных символов либо с Λ.

Рассмотрим каждое правило $A \to aX_1 \ldots X_k$, $a \in T, X_i \in T \cup N$. Если $X_i \in T$, то заменим в этом правиле X_i на дополнительный $X_i' \in N$ и добавим $X_i \to X_i'$.

Полученная грамматика находится в нормальной форме и является простой $LL(1)$-грамматикой. Лемма 1 доказана.

Лемма 2. *В грамматике* $G \in \widetilde{G}(n, L), n \geqslant 2, L \geqslant 1$ *для любых нетерминальных символов* A *и* B*, таких, что* A *достижим из* B*, найдется слово* $\alpha \in \{0, 1, A\}^*$ *, такое, что* $B \Rightarrow^* \alpha$*, и*
$|\alpha| \leqslant \frac{L^{n-1}-1}{L-1} + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1})$*, если* B *— аксиома,*
$|\alpha| \leqslant \frac{L^{n-2}-1}{L-1} + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1})$ *иначе,*
где k *— число вхождений* A *в* α*, а* i_A *— число НС, из которых недостижим* A*.*

Доказательство. Докажем утверждение леммы сначала для случая, когда $k = 0$, то есть $\alpha \in \{0, 1\}^*$.

Рассмотрим $B \in N$ и α наименьшей длины, такое что $B \Rightarrow^* \alpha$. Вывод α можно представить как последовательность непосредственных выводов $B \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \ldots \Rightarrow \alpha$. Сопоставим выводу $B \Rightarrow^* \alpha$ ориентированное от корня дерево следующим образом.

B — корень дерева. Дерево достраивается после каждого непосредственного вывода.

Если $B \Rightarrow \alpha_1 = aA_1 \ldots A_k$, то из корня выходит $k+1$ ребро, а вершинам (пока — листьям) приписаны слева направо $a, A_1, \ldots, A_k$.

Пусть построено дерево до вывода α_i. Пусть $\alpha_i = \beta_1 A\beta_2, \alpha_{i+1} = \beta_1 bC_1 \ldots C_t\beta_2$, то есть в грамматике есть правило $A \to bC_1 \ldots C_t$. Тогда из листа, соответствующему данному вхождению символа A (если считать слева направо, то его номер $|\beta_1| + 1$) проведем $t+1$ ребро, а новым вершинам припишем слева направо $b, C_1, \ldots, C_t$.

В результате построения листьями дерева будут терминальные символы.

Назовем такое дерево *деревом вывода*.

На рисунке 1 приведен пример построения дерева вывода.

Высотой дерева будем называть наибольшее количество вершин в ориентированном пути от корня до листа включительно. Будем говорить, что вершина или приписанный ей символ находится на высоте h от корня, если количество вершин на пути от корня до этой вершины есть h.

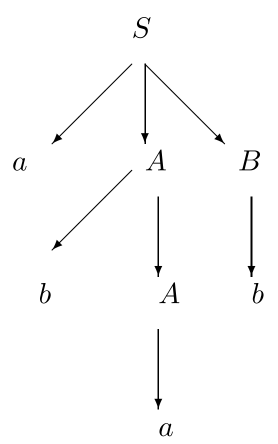

Рис. 1. Дерево вывода для $S \Rightarrow aAB \Rightarrow abAB \Rightarrow abaB \Rightarrow abab$.

Покажем, что в дереве вывода кратчайшего слова α ни в одном пути от корня до вершины не может встречаться какой-либо $D \in N$ более одного раза.

Действительно, пусть $B \Rightarrow^* \alpha_1 D, \alpha_2$, где $\alpha_1, \alpha_2 \in \{0,1\}^*$. Пусть также $D \Rightarrow^* \beta_1 D \beta_2, \beta_1, \beta_2 \in \{0,1\}^*$, и $\alpha = \alpha_1 \beta_1 \beta_0 \beta_2 \alpha_2$, где $D \Rightarrow^* \beta_0$. Но тогда слово $\alpha_1 \beta \alpha_2$ также выводимо из B, и его длина $|\alpha_1 \beta_0 \alpha_2| \leqslant |\alpha|$, что противоречит тому, что α — наименьшей длины.

Таким образом, дерево вывода не может быть высоты большей, чем один плюс число НС без учета A, причем, так как аксиома не может быть выведена ни из какого НС, если B — не аксиома, то дерево не может быть высоты, большей $n-1$.

Каждому непосредственному выводу соответствует одна буква слова α, а так как для каждого НС в выводе есть непосредственный вывод, длина слова равна числу НС в дереве вывода.

Таким образом, наименьшая длина слова α не превосходит $1+L+\ldots+L^{n-2} = \frac{L^{n-1}-1}{L-1}$, если B — аксиома, и $|\alpha| \leqslant 1+L+\ldots+L^{n-3} = \frac{L^{n-2}-1}{L-1}$ иначе.

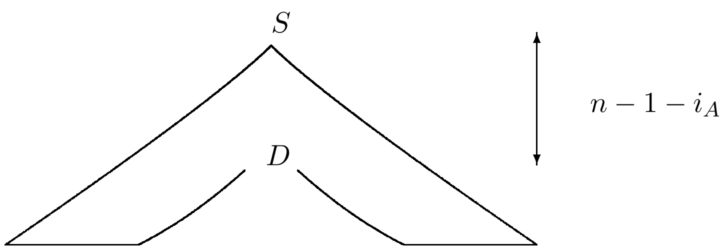

Рис. 2. Вывод слова в алфавите $\{0, 1, D\}$.

Предположим теперь, что символ A встретился в слове α 1 раз.

Рассмотрим вывод слова наименьшей длины из аксиомы. Если он содержит D, такой, что $D \to a\alpha A\beta \in P, \alpha, \beta \in N^*, a \in \{0,1\}$, то на высоте от корня не более $n-1-i_A$, потому что на пути от корня к D не встречались ни A, ни символы, из которых A недостижим. Выведем из одного символа D НС A. Также в правой части правила $D \to a\alpha A\beta$ могут встретиться любые нетерминальные символы, кроме S и A, так же, как и в кратчайших выводах из них. Дерево вывода из D содержит, самое большее, $1+1+(L-1)(1+L+\ldots+L^{n-3})$ НС.

Поэтому длина α не превосходит $1+L+\ldots+L^{n-2} - \sum_{i=0}^{i_A} L^i + 1 + (L-1)(1+\ldots+L^{n-3}) + 1 = \sum_{i=0}^{n-2} L^i + (L^{n-2} + 1 - \sum_{i=0}^{i_A} L^i) = \frac{L^{n-1}-1}{L-1} + L^{n-2} - \frac{L^{i_A+1}-L}{L-1}$.

Если вывод слова наименьшей длины не содержит D, такой что $D \to a\alpha A\beta \in P$, он содержит F_j, $j = 1, \dots, n-2$, такой что существуют правила

$F_m \to a_m\alpha_m F_{m+1}\beta_m \in P$, $\alpha_m, \beta_m \in N^*$, $a_m \in \{0,1\}$, $m = j, \dots, n-2$,

$F_{n-1} = D \to a\alpha A\beta \in P$.

Такие НС в выводе всегда есть, так как A достижим из аксиомы, и в крайнем случае $F_j = S$.

Выберем наибольшее j, такое что в выводе кратчайшего слова присутствует F_j, но отсутствуют $F_{j+1}, \dots, F_{n-1}$.

Тогда НС F_j появляется на высоте от корня не большей $j - i_A$, а все дерево вывода высоты не большей $n - (n-j) = j$.

Таким образом, длина слова $\beta_0 \in \{0, 1, F_j\}^*$, такого что $B \Rightarrow^* \beta_0$, $|\beta_0| \leqslant 1 + L + \ldots + L^{j-1} + 1 - \sum_{i=0}^{i_A} L^i$.

Рассмотрим вывод $F_j \to aC_1 \dots C_i F_{j+1} C_{i+1} \dots C_t, t \leqslant L-1$, и слова наименьшей длины, выводимые из $C_1, \dots, C_t, F_{j+1}$. Последовательно проверим, участвуют ли в этих выводах НС $A, F_{n-1}, \dots, F_{j+2}$. Пусть есть $F_{j+i_2}, i_2 > 1$, но нет $A, F_i, i > j + i_2$. Высота каждого дерева не больше $j + i_2 - 1$ (нет также аксиомы), а F_{j+i_2} не мог появиться на высоте, большей $(j + i_2 - 1) - i_A - j$, так как из j НС, выводимых в кратчайшем слове из S, не выводились ни A, ни $F_i, i = j, \dots, n-2$, следовательно, они не могут встретиться на пути к F_{j+i_2}. Если в выводе слов наименьшей длины не встретился ни один из этих символов, то объявим $F_{j+i^2} = F_{j+1}$.

Поэтому наименьшая длина такого слова β_1 в алфавите $\{0, 1, F_{j+i_2}\}$, что $F_j \Rightarrow^* \beta_1$, не превосходит $1 + L(1 + L + \ldots + L^{j+i_2-2}) + 1 - \sum_{i=0}^{i_A+j-1} L^i \leqslant 1 + L^j + \ldots + L^{j+i_2-1}$. Здесь 1 соответствует F_j, который уже был посчитан на предыдущем шаге в длине слова β_0, выведенного из S, поэтому в общей длине слова единица учитываться не будет.

На следующем шаге переходим к НС F_{j+i_2}, строим для него, как и для F_j, слово β_2 в алфавите $\{0, 1, F_{F_j+i_3}\}, i_3 > i_2$, такое что $F_{j+i_2} \Rightarrow^* \beta_2$, и его длина $L^{j+i_2} + \ldots + L^{j+i_3-1} + 1$.

Число шагов не больше $n - j - 1$.

Таким образом, суммарная длина слова $|\alpha| \leqslant 1 + L^{i_A+1} + L^{i_A+2} + \ldots + L^{n-2} + L^{n-2}$.

Пусть теперь в слове α содержится k букв A.

Символ A непосредственно выводится из некоторых $D^1, \ldots, D^t$, $t \leqslant k$. Для слова $\beta \in \{0, 1, D^1, \ldots, D^t\}^*$, такого что $S \Rightarrow^* \beta$, выполнено $|\beta| \leqslant 1 + L + \ldots + L^{n-2} - t(\frac{L^{i_A+1}-L}{L-1})$, так как они не могут появиться на высоте, большей $n - 1 - i_A$.

Пусть в каждом использованном в α правиле $D^i \to a\alpha_i A\beta_i, \alpha_i, \beta_i \in N^*$ в правой части содержится k_i букв A, $k_1 + \ldots + k_t = k$. Из остальных $L - k_i$ НС выведем слова наименьшей длины ($\leqslant 1 + L + \ldots + L^{n-3}$).

Тогда

$|\alpha| \leqslant 1 + L + \ldots + L^{n-2} + (L - k_1)(1 + \ldots + L^{n-3}) + \ldots + (L - k_t)(1 + \ldots + L^{n-3}) + t - t(\frac{L^{i_A+1}-L}{L-1}) = 1 + \ldots + L^{n-2} + (tL - k)(1 + \ldots + L^{n-3}) + t - t(\frac{L^{i_A+1}-L}{L-1}) = 1 + L + \ldots + L^{n-2} + t(1 + \ldots + L^{n-2} - \frac{L^{i_A+1}-L}{L-1}) - k(1 + \ldots + L^{n-3}) \leqslant \sum_{i=0}^{n-2} L^i + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1}) = \frac{L^{n-1}-1}{L-1} + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1})$

В случае $B \neq S$ в дереве вывода до $D^1, \ldots, D^t$ также не встречается S.

Таким образом, если $B \neq S$,

$$|\alpha| \leqslant \frac{L^{n-2} - 1}{L - 1} + k(L^{n-2} - \frac{L^{i_A+1} - L}{L - 1}).$$

Лемма 2 доказана.

Замечание. Оценка в лемме 2 не может быть улучшена.

Рассморим следующую грамматику $G_1 \in \widetilde{G}(n, L)$. Длина каждой команды $0, 1$ или L, $a, b \in \{0, 1\}$.

$S \to aI_{n-2-i_A} \ldots I_{n-2-i_A}, S \to bI_{n-2-i_A} \ldots I_{n-2-i_A}$,

$I_{i+1} \to aI_i \ldots I_i, I_{i+1} \to bI_i \ldots I_i, i = 1, \ldots, n - 3 - i_A$,

$I_1 \to aJ_1 \ldots J_1, I_1 \to bI_{n-2-i_A} \ldots I_{n-2-i_A}A$,

$A \to a$,

$J_i \to aJ_{i+1} \ldots J_{i+1}, J_i \to bJ_{i+1} \ldots J_{i+1}, i = 1, \ldots, i_A - 1$,

$J_{i_A} \to a, J_{i_A} \to b$.

Для данной грамматики и для каждого k, такого что $0 \leqslant k \leqslant L^{n-2}$, $\forall \alpha \in \{0, 1, A\}^*$, такого, что $S \Rightarrow^* \alpha$, выполнено $|\alpha| \geqslant \frac{L^{n-1}-1}{L-1} + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1})$.

Также для каждого k, такого что $0 \leqslant k \leqslant L^{n-2}$, $\forall \alpha \in \{0, 1, A\}^*$, такого, что $I_1 \Rightarrow^* \alpha$, выполнено $|\alpha| \geqslant \frac{L^{n-2}-1}{L-1} + k(L^{n-2} - \frac{L^{i_A+1}-L}{L-1})$.

Доказательство теоремы 1. *Достаточность.*

Пусть $A_1, \dots, A_p$ — НС только с одним правилом вывода.

Сначала сопоставим каждому НС $B \in N$, метку $\mu(B)$, равную длине кратчайшей выводимой из него терминальной цепочки, следующим образом.

Для каждого НС, такого что $B \to a$, положим $\mu(B) = 1$.

Остальным НС присвоим переменные метки $\mu'(C) = +\infty$.

Шаг алгоритма.

Рассмотрим символ C, такой, что $C \to aC_1 \dots C_t \in P$ и символам $C_1, \dots, C_t$ присвоены постоянные метки $\mu(C_1), \dots, \mu(C_t)$. Присвоим C переменную метку $\mu'(C) = \min\{\mu'(C), \mu(C_1) + \mu(C_2) + \dots + \mu(C_t) + 1\}$.

Покажем, почему найдется такой символ C, если еще не всем НС приписана постоянная метка.

Так как все НС достижимы, рассмотрим дерево вывода некоторого слова из данной грамматики, содержащее НС без постоянных меток. Если некоторая вершина D с постоянной меткой является корнем поддерева, где каждому НС приписана постоянная метка или терминальный символ, удалим это поддерево, оставив только его корень D. Тогда листьями дерева станут вершины с НС с постоянными метками и терминальные символы. Очевидно, найдется вершина, все ребра из которой ведут к листьям. На Рис.3 приведен пример.

Далее из переменных меток выбирается наименьшая, и соответствующему НС присваивается постоянная метка $\mu(C) = \mu'(C)$.

Шаг алгоритма повторяется до тех пор, пока каждому НС не будет сопоставлена постоянная метка. Максимальное число шагов — $|N| - 1$. Метка $\mu(B)$ равна наименьшей длине слова α, такого что $B \Rightarrow^* \alpha$.

Теперь тем НС, из которых достижимы какие-нибудь НС из $A_1, \dots, A_p$ (обозначим это множество $E \subseteq N$), присвоим сначала переменные метки $\nu'(B) = +\infty$, а НС $A_1, \dots, A_p$ — постоянные метки $\nu(A_i) = 0, i = 1, \dots, p$.

Рассмотрим правила, в правой части которых содержится хотя бы одна буква $A_1, \dots, A_p$(такие есть, так как они достижимы): $B \to aC_1 \dots A_i \dots C_t$. Тогда $\nu'(B) = \min\{\nu'(B), \mu(C_1) + \dots + \mu(C_t) + 1\}$.

Далее из переменных меток выбирается наименьшая, и этому символу присваивается постоянная метка: $\nu(B) = \nu'(B)$.

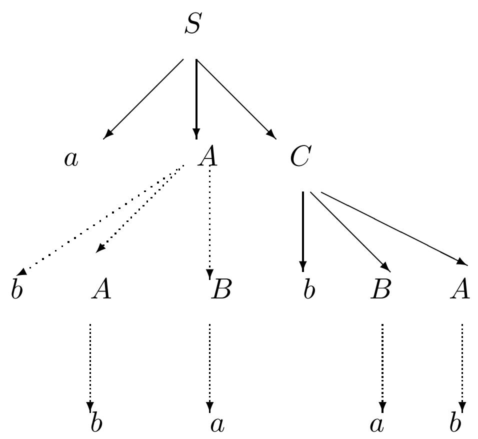

Рис. 3. $\mu(A) = \mu(B) = 1, \mu'(C) = 3$.

Шаг алгоритма.

Рассматриваются такие команды, что в их правой части есть НС с постоянной меткой ν. То есть $B \to aC_1 \dots C_t \in P$, и определены $\nu(C_{i_1}), \dots, \nu(C_{i_l}), \{i_1, \dots, i_l\} \subseteq \{1, \dots, t\}$, пусть $i_1 < i_2 < \dots < i_l$.

Тогда $\nu'(B) = \min\{\nu'(B), \nu(C_{i_1}) + \sum_{i \in \{1,\dots,t\} \setminus \{i_1,\dots,i_l\}} \mu(C_i) + 1\}$.

Далее выбирается один НС $C_0 = \arg\min_{C \in E}(\nu'(C))$, и для него $\nu(C_0) = \nu'(C_0)$.

Шаг алгоритма повторяется до тех пор, пока всем НС $B \in E$ не будут присвоены постоянные метки. Число шагов не превосходит $|E|$.

Величина $\nu(C)$ есть сумма длины кратчайшего слова $\beta \in \{0,1\}^*$, такого что $C \Rightarrow \beta A_i \xi, \xi \in \{0, 1, A_1, \dots, A_p\}^*$, и длин кратчайших слов, выводимых из появившихся НС из $N \setminus E$.

Был построен МП-автомат с одним состоянием $(\{q_0\}, \{0,1\}, \Gamma, Z, \Phi)$, допускающий $(L(G))^\infty$. Построим на его основе МП-автомат $\mathfrak{A}$ с n состояниями, также допускающий сверхитерацию данного языка и зададим для него выходную функцию $f : Q \times N \to \{0,1\}$ следующим образом.

$Q = \{q_A, A \in N\}, \Sigma = \{0,1\}, \Gamma = \{N \cup Z\}$.

Чтобы задать множество команд Φ', определим сначала функцию $\varphi : Q \times \Gamma \times \{0,1\} \to Q$ и f.

$\varphi(q_S, S, a) = q_C$, если $\nu(S) = 1 + \nu(C) + \sum_{i=1}^{t} \mu(C_i)$ и $S \to aCC_1 \dots C_t$, иначе $\varphi(q_S, S, a) = q_S$.

$f(q_C, C) = a$, если $C \to bC_1 \dots \widetilde{C} \dots C_t \in P$ и $\nu(C) = 1 + \nu(\widetilde{C}) + \sum_{i=1}^{t} \mu(C_i)$ или $C \in N \setminus E, C \to bC_1 \dots C_t$ и $\mu(C) = 1 + \sum_{i=1}^{t} \mu(C_i)$.

$f(q_A, C) = a$, если $C \to bC_1 \dots C_t$ и $\mu(C) = 1 + \sum_{i=1}^{t} \mu(C_i)$.

Пусть $C \to a\widetilde{C}\xi \in P$ и $A \in E$.

Тогда $\varphi(q_A, C, a) = q_{\widetilde{C}}$, если $\nu(A) = 1 + \nu(\widetilde{C}) + \sum_{i=1}^{t} \mu(C_i)$ или $f(q_A, C) = a$.

$\varphi(q_A, C, a) = q_A$, иначе.

Если $A \in N \setminus E$, то $\varphi(q_A, C, a) = q_{\widetilde{C}}$.

Множество команд $\Phi' = \{(q, B, a \to r, \xi), B \in N, \xi \in N^*, a \in \{0,1\}, q, r \in Q$, таких что $(q_0, B, a \to q_0, \xi) \in \Phi$ и $r = \varphi(q, B, a)\}$.

Таким образом, каждому НС сопоставляется буква, отличная от первой буквы кратчайшего слова, приводящего МП-автомат в конфигурацию с верхним символом из $A_1, \dots, A_p$ или слова, уменьшающего глубину стека.

Подсчитаем степень угадывания для МП-автомата с такой выходной функцией.

Находясь в конфигурации с верхним НС из $A_1, \dots, A_p$, автомат всегда угадывает следующий символ.

Пусть автомат находится в некотором состоянии q_D.

Выходы организованы так, что автомат выдает букву, отличную от буквы кратчайшего слова, приводящего в $A_1, \dots, A_p$. Состояние может измениться после угадывания или после достижения НС с меткой ν.

Во втором случае это слово β продолжается. Поэтому по лемме 2 длина такого неугаданного слова не больше $L^{i_A+1} + \dots + L^{n-2} + L^{n-2}$, и ранее, чем через это количество шагов, произойдет угадывание.

После угадывания автомат оказывается в состоянии q_C. Если C из E, для него повторяются те же рассуждения. Если состояние не из E, то неугаданные выводы также составляют кратчайшее слово, причем их длины были учтены в ν за исключением случая, если эти НС выведены из $A_i, i = 1, \dots, p$.

Наибольшая длина такого слова, выводимого из A_i, $\leqslant 1+\ldots+L^{i_A}$. Поэтому степень угадывания $\geqslant \frac{k}{1+\ldots+L^{n-2}+kL^{n-2}} \geqslant \frac{1}{1+\ldots+L^{n-2}+L^{n-2}}$.

Таким образом, $c^{\mathfrak{A}}(L(G)^\infty) \geqslant \frac{1}{1+L+\ldots+L^{n-3}+2L^{n-2}} > 0$.

Достаточность доказана.

Необходимость. Пусть есть некоторое устройство $\mathfrak{B}$, которое частично угадывает данное множество сверхслов. Его выходная функция сопоставляет текущему префиксу слова $\alpha]_t$ выходную букву $\psi(\alpha]_t) \in \{0,1\}$. Построим сверхслово α, степень угадывания данным устройством равна нулю.

Пусть $\alpha(1) = a$, и $\alpha(t) = \overline{\psi(\alpha]_{t-1})}$ до некоторого $t = h_1$. Рассмотрим работу МП-автомата, допускающего $(L(G))^\infty$, на этом отрезке. Обозначим за $\varphi(\beta, \xi)$ содержимое стека в конфигурации, в которой окажется МП-автомат после подачи слова β на конфигурацию (q_0, ξ).

Для того, чтобы сверхслово $\alpha \in (L(G))^\infty$ нужно, чтобы $\exists i_1, i_2, \ldots$, такие что $\varphi(\alpha|_{i_j}, Z) = Z, j = 1, 2, \ldots$

Какова бы ни была глубина стека МП-автомата в момент h_1, найдется конечное слово $\alpha(h_1+1)\ldots\alpha(h_1+g_1)$, такое, что $\varphi(\alpha]_{h_1+g_1}, Z) = Z$.

Далее допишем полностью неугаданное слово длины $h_2 = 2\cdot(h_1 + g_1)$, а потом слово, приводящее МП-автомат в Z, длины g_2.

Пусть сделано i шагов: построен префикс сверхслова α длины $h_1 + g_1 + \ldots + h_i + g_i$. Допишем полностью неугаданное слово длины $h_{i+1} = (i+1)\cdot(h_1 + g_1 + \ldots + h_i + g_i)$, а затем слово, приводящее МП-автомат в Z, некоторой длины g_{i+1}. И так далее.

В моменты времени $h_1 + g_1 + \ldots + h_t + g_t$ МП-автомат будет находиться в конфигурации с пустым магазином, поэтому $\alpha \in (L(G))^\infty$. А в моменты $h_1 + g_1 + \ldots + h_t + g_t + h_{t+1}$ доля угаданных символов будет $\leqslant \frac{1}{t+2}$.

Таким образом, степень $c^{\mathfrak{B}}(\alpha) \leqslant \underline{lim}_{t\to\infty} \frac{1}{t+2} = 0$.

Теорема 1 доказана.

Доказательство теоремы 2. Если $n = 1$, грамматика G находится в нормальной форме, а $(L(G))^\infty$ частично угадываемо, то в грамматике может быть только одно правило $S \to a$. Тогда это сверхсобытие может угадываться конечным автоматом, выдающим константу a, и степень угадывания равна 1.

Рассмотрим следующую грамматику G_0 для произвольных $n \geqslant 2$, $L \geqslant 1$. Длина каждого правила L или 1.

$S \to aI_{n-2} \dots I_{n-2}A, S \to b$,

$A \to aI_{n-2} \dots I_{n-2}$,

$I_j \to aI_{j-1} \dots I_{j-1}, I_j \to bI_{j-1} \dots I_{j-1}, j = 2, \dots, n-2$,

$I_1 \to a, I_1 \to b$.

Рассмотрим некоторое устройство $\mathfrak{B}$ с выходной функцией $\psi(\omega)$ от уже поступившей части слова ω. Построим сверхслово α из сверхитерации языка $L(G_0)$. Если в начальный момент $\psi(\Lambda) = a$, то сверхслово b^∞ не будет угадываться, пока выход не изменится.

Если $\psi(\Lambda) = b$ (или $\psi(bb \dots b) = b$), то пусть $\alpha(1) = a$. Далее можно строить $\alpha(i) = \overline{\psi(\alpha]_{i-1})}$ до тех пор, пока не будет достигнут символ A, что произойдет не ранее, чем через $1 + (L-1)(1 + L + \dots + L^{n-3}) = L^{n-2}$ тактов. После одного возможного угаданного символа снова можно подавать на вход $\alpha(i) = \overline{\psi(\alpha]_{i-1})}$ столько тактов j, чтобы $S \Rightarrow^* \alpha]_j$. Видно, что $j \geqslant 1 + L + \dots + L^{n-3} + 2L^{n-2}$. Дальше сверхслово строится аналогично.

Степень угадывания $c^{\mathfrak{B}}(L(G_0)^\infty) \leqslant c^{\mathfrak{B}}(\alpha) \leqslant \frac{1}{1+L+\dots+L^{n-3}+2L^{n-2}} = \frac{1}{\frac{L^{n-1}-1}{L-1}+L^{n-2}} = \frac{L-1}{2L^{n-1}-L^{n-2}-1}$ для произвольного $\mathfrak{B} \in \mathfrak{M}$.

Так как для предложенного в доказательстве теоремы 1 автомата $\mathfrak{A}$ для любого частично угадываемого сверхсобытия выполняется $c^{\mathfrak{A}}(L(G)^\infty) \geqslant \frac{L-1}{2L^{n-1}-L^{n-2}-1}$, теорема 2 доказана.

4. Угадывание в смысле верхнего предела

Доказательство теоремы 3. *Достаточность.*

Рассмотрим простую $LL(1)$-грамматику G в нормальной форме.

Зададим выходную функцию $f : N \to \{0, 1\}$, зависящую от верхних нетерминальных символов стека МП-автомата $(\{q_0\}, \{0, 1\}, \Gamma, Z, \Phi) = \mathfrak{A}$.

Для каждого $i = 0, \dots, k$ выполним следующее.

В грамматике $G^{(i)}$ (считаем $G^{(0)} = G$) рассмотрим элементы D_{i+1}. Для каждой сильно-связной регулярной компоненты $\widetilde{R}$ выберем один стирающий выход произвольным образом. Для такого НС

$A \to aB, B \notin \widetilde{R}$ назначим выход: $f(A) = b$. Для остальных НС из $\widetilde{R}$ расставим выходы следующим образом.

1-й шаг.

Выделим НС, из которых достижимо по одной букве. Тогда если $\varphi(q, x) = \hat{q}$, то $f(q) = \overline{x}$. Выделенные НС обозначим как класс Q'.

Пусть сделано $i - 1$ шагов. Опишем i-й шаг.

Если для некоторого НС $A \in \widetilde{R}$, на котором еще не определена выходная функция, и для некоторого x выполнено $A \to xB, B \in Q'$, то $f(A) = \overline{x}$. Из всех таких НС образуем класс Q' (рассмотренные на предыдущем шаге состояния удаляем из рассмотрения).

И так далее.

Для оставшихся элементов грамматики $G^{(i)}$ расставим выходы следующим образом.

Для тех НС, которые не вошли в $\cup_{i=1}^{k} D_i$, на выходе будет начальная буква стирающего правила: $f(A) = a$, если $A \to a$.

Для каждой регулярной компоненты $\widetilde{R}$ выбирается нестирающий выход $A \to aC_1 \ldots C_t, t \geqslant 1$, для него $f(A) = b$. Для остальных элементов $\widetilde{R}$ выходы расставляются так же, как выше для регулярной компоненты $\cup_{i=1}^{k} D_i$. Для остальных НС выбирается любая буква в качестве выхода.

Перенесем значения $f(A)$ на элементы A грамматики G.

Таким образом всем нетерминальным символам сопоставлены выходные символы.

Докажем, что при такой расстановке выходных символов множество будет частично угадано в смысле верхнего предела.

Рассмотрим произвольное слово из $L(G)$ $a_1 \ldots a_t$, поданное на начальную конфигурацию (q_0, S) или на конфигурацию (q_0, Z) МП-автомата. t символам слова соответствуют t конфигураций и t переходов от одной конфигурации к следующей.

Эти t переходов можно представить как сумму переходов по регулярным, стирающим и удлинняющим правилам $t = n_r + n_s + n_l$. Далее будем называть их соответственно регулярными, стирающими и удлинняющими переходами.

Рассмотрим отдельно переходы для символов из $\cup_{i=1}^{k} D_i$ и остальных. Рассмотрим стирающие переходы для НС не из $\cup_{i=1}^{k} D_i$. Они, очевидно, все угаданы. Какова доля неугаданных переходов? Число удлинняющих переходов меньше числа стирающих переходов в l раз (или еще меньше): $l \cdot n_l \leqslant n_s \Rightarrow \frac{n_s}{n_s+n_l} \geqslant \frac{l}{l+1}$.

Сколько возможно неугаданных регулярных переходов? Если регулярных переходов было $n'_r > R_c$ подряд, то доля угаданных регулярных переходов будет $\geqslant \frac{1}{R}$ так как для каждого регулярного цикла выходы расставлены так, что раньше, чем после $R-1$ неугаданных регулярных переходов МП-автомат или угадывет, или выходит из компоненты.

Если прошло $n'_r \leqslant R_c$ регулярных переходов подряд, и все они не были угаданы, то им будет соответствовать по крайней мере один угаданный стирающий переход, то есть $n'_r \leqslant R_c \cdot n'_s$. И тогда $\Rightarrow \frac{n'_s}{n'_r+n'_s} \geqslant \frac{1}{R_c+1}$.

Пользуясь неравенством

$$\frac{a+b}{A+B} \geqslant \min(\frac{a}{A}, \frac{b}{B}), a, b, A, B \in N,$$

получаем, что доля угаданных переходов не из $\cup_{i=1}^{k} D_i$ в данном слове не меньше $\min(\frac{1}{R_c+1}, \frac{l-1}{l})$.

Теперь рассмотрим НС из $\cup_{i=1}^{k} D_i$, переходы для них тоже можно разделить на удлинняющие, стирающие и регулярные: d_l, d_s, d_r. Для них выполнены те же соотношения, что и для переходов не из $\cup_{i=1}^{k} D_i$. Однако стирающие переходы не всегда будут угаданы.

Рассмотрим стирающий переход из D_1, например, из НС $D \to a$. Этот НС должен был появиться в стеке, то есть $A \to a\alpha D\beta \in P$. Если этот удлинняющий переход был угадан ($f(A) = a$), то он соотвествует неугаданному стирающему, хотя возможно, не одному. В случае $f(A) \neq a$ есть два варианта. Первый — $A \notin \cup_{i=1}^{k} D_i$, тогда в грамматике $G^{(k)}$ соответствующий переход не мог быть стирающим $A \to a$, потому что тогда он был бы угадан, значит, в правой части команды есть по крайней мере 1 НС $\notin \cup_{i=1}^{k} D_i$. Значит, НС $\in \cup_{i=1}^{k} D_i$ в данном слове не больше, чем в $L-1$ раз больше, чем НС $\notin \cup_{i=1}^{k} D_i$. В случае же $A \in \cup_{i=1}^{k} D_i$ нужно проследить, где A появилось в стеке.

Из всех t переходов можно выделить последовательность:
$A_{i^u} \to a_{i^u}\alpha_u A_{i^{u+1}}\beta_u$, $\alpha_u, \beta_u \in N^*, u = 1, \dots, j-1$,
$A_{i^j} \to a_{i^j}$,

где $1 \leqslant i^1 < \ldots < i^j \leqslant t-1$, $A_{i^j} = D \in D_1, A_{i^2}, \ldots, A_{i^{j-1}} \in \cup_{i=1}^k D_i$, а для A_{i^1} выполнено $f(A_{i^1}) = a_{i^1}$ или $A_{i^1} \to a_{i^1}\alpha_1' B \beta_1', B \notin \cup_{i=1}^k D_i$.

Уже упоминалось, почему в случае неугадывания символа $A \notin \cup_{i=1}^k D_i$ соответствующий переход содержит в правой части хотя бы один символ не из $\cup_{i=1}^k D_i$. Покажем теперь, почему данную последовательность можно выделить, то есть почему невозможен случай $A_{i^1} = Z, f(Z) \neq a_{i^1}$.

Пусть, действительно, все переходы — элементы $\cup_{i=1}^k D_i$, это значит, что в правой части каждой команды только элементы $\cup_{i=1}^k D_i$, причем это верно и для перехода по второй букве для каждого НС. Но так как $A_{i^1} = Z$, все НС, которые могут выведены из аксиомы, принадлежат $\cup_{i=1}^k D_i$. Так как изначально рассматривается грамматика без недостижимых символов, то это противоречие с предположением, что не все НС — элементы $\cup_{i=1}^k D_i$.

Подсчитаем, какая наибольшая длина j может быть у такой цепи переходов.

По условию $A_{i^2}, \ldots, A_{i^j} \in \cup_{i=1}^k D_i$. Известно, что в такой цепи элемент D_i может возникнуть только в правой части команды НС $\in D_{i'}, i' \leqslant i$, причем число неугаданных НС $\in D_i$ подряд не может превосходить $\max(R_c, R-1) = R_c$ из-за расстановки выходов в грамматике $G^{(i-1)}$, где элементы D_i образуют одну или несколько сильносвязных регулярных компонент. Таким образом $j \leqslant (R_c + 1) \cdot k$.

Если переход из A_{i^1} был угадан, то он соответствует неугаданному стирающему $A_{i^j} = D$. Но могли быть другие неугаданные стирающие переходы, для которых подобная цепь приводит к тому же самому A_{i^1}. Оценим их число.

Подсчитаем максимальное число НС в каждом классе D_i, их сумма вместе с единицей, соответствующей A_{i^1}, будет равна числу непосредственных выводов, то есть числу терминальных символов в данном выводе.

НС из D_k, если они выводимы из A_{i^1}, может появиться, самое большее, $L \cdot (R_c + 1)$. Причем из них только L могут быть выведены непосредственно из A_{i^1}, а остальные — из других НС из D_k.

Пусть НС из D_i выведено некоторое число T. Из тех элементов D_i, для которых соответствующее правило грамматики $G^{(i-1)}$ стирающе, может быть выведено, самое большее, по L НС из D_{i+1}, а из остальных (где переход регулярный) — по $L-1$. Так как стирающий переход может быть только один, а регулярных, самое большее, R_c, то наибольшее число НС из D_{i+1} — $\frac{T}{R_c+1}\cdot(L+R_c(L-1))(R_c+1)$. Таким образом, наибольшее число НС из D_i — $(R_c+1)\cdot L(R_c(L-1)+L)^{i-1}$.

Поэтому общее число неугаданных переходов, соответствующих одному угаданному:

$$\sum_{i=0}^{k-1} L(R_c+1)(R_c(L-1)+L)^i = \frac{L(R_c+1)((R_c(L-1)+L)^k-1)}{(R_c(L-1)+L)-1}.$$

В другом случае из A_{i^1} ведет переход в некоторый $B \notin \cup_{i=1}^k D_i$, а для таких НС угадывается по крайней мере один символ из R_c+1 или l из $l+1$. При этом длина остального вывода уменьшается на $\sum_{i=0}^{k-1}(R_c+1)(R_c(L-1)+L)^i$, что больше, чем R_c+1 и $l+1$. Следовательно, степень угадывания не меньше, чем в предыдущем случае.

Верхнюю степень угадывания можно оценить как

$$\overline{c}^{\mathfrak{A}}((L(G))^\infty) \geqslant \min\{\frac{l}{l+1}, \frac{1}{R_c+1}, \frac{L-1}{L(R_c(L-1)+L)^k-1}\},$$

так как $(1+\frac{L(R_c+1)((R_c(L-1)+L)^k-1)}{(R_c(L-1)+L)-1})^{-1} = \frac{R_c(L-1)+L-1}{(R_c+1)(L(R_c(L-1)+L)^k-1)} = \frac{L-1}{L(R_c(L-1)+L)^k-1}$.

Таким образом, множество частично угадываемо в смысле верхнего предела. Достаточность доказана.

Необходимость.

Теперь покажем, что в случае, когда все нетерминальные символы грамматики G вошли в $\cup_{i=1}^k D_i$, множество $(L(G))^\infty$ не является частично угадываемым.

Предположим, что есть некоторое устройство $\mathfrak{B}$, которое частично угадывает данное множество сверхслов. Его выходная функция сопоставляет текущему префиксу слова $\alpha]_t$ выходную букву $\psi(\alpha]_t) \in \{0,1\}$.

Рассмотрим работу МП-автомата, допускающего $(L(G))^\infty$, на этом отрезке.

Для того, чтобы сверхслово $\alpha \in (L(G))^\infty$ нужно, чтобы $\exists i_1, i_2, \ldots$, такие что $\varphi(\alpha|_{i_j}, Z) = Z, j = 1, 2, \ldots$

Для каждого НС (а, следовательно, и для каждой конфигурации) есть ровно два перехода, иначе все НС не могли бы быть в $\cup_{i=1}^k D_i$. Будем строить сверхслово α из данного множества, степень угадывания которого будет равна нулю.

Пусть первая буква сверхслова $a_1 = 0$, автомат оказывается в некоторой конфигурации $\gamma \in \Gamma^*$, где глубина стека не больше L. Рассмотрим верхний символ стека $B_1 : \gamma = B_1\gamma'$ и способы его стирания.

Шаг 1.

Пусть $B_1 \in D_{k'}, k' \leqslant k$. Подадим на эту конфигурацию полностью неугадываемое слово длины r_1: $a_{p+1} = \overline{\psi(\gamma, a_2 \ldots a_p)}, p = 1, \ldots, r_1$. Если для некоторого $p^* \leqslant r_1 + 1$ выполнено $\varphi(a_2 \ldots a_{p^*}, B_1\gamma') = C_1 \ldots C_t\gamma'$, где $C_i \in D_{k''}, k'' < k', i = 1, \ldots, t$, то переходим к следующему шагу.

Иначе подадим далее слово $a_{r_1+1} \ldots a_{r_1+t^*}$, такое, что $\varphi(a_1 \ldots a_{r_1+t^*}, Z) = C_1 \ldots C_t\gamma'$, где $C_i \in D_{k''}, k'' < k', i = 1, \ldots, t$, где $t^* \leqslant R$, так как в грамматике $G^{(k'-1)}$ это регулярная сильно-связная компонента, все выходы из которой стирающие, а R — наибольшее число НС в регулярной сильно-связной компоненте.

После этой процедуры в стек могло добавиться, самое большее, $(r_1 + t^* - 1) \cdot L$ НС из классов $D_{k''}, k'' < k'$.

Шаг 2.

Пусть глубина стека равна g_2. Для текущего верхнего символа стека B_1^2 из класса $D_{k''}$ также подадим полностью неугаданное слово длины r_1. Если за время подачи этого слова не произошло стирания B_1^2 или замены его на символы из классов $\cup_{i=1}^k D_i$ с меньшим номером, то подаем слово длины не большей R, которое не оставляет в стеке выше глубины g_2 символы из классов D_i с меньшим номером i.

И так далее. После каждого шага глубина стека может увеличиваться, но появляются только НС из $\cup_{i=1}^k D_i$ с номером, меньшим номера класса верхнего символа в начале шага.

Через конечное число шагов достигнется конфигурация γ'. Проделаем ту же операцию для остальных элементов стека. Число шагов возрастет не более, чем в L раз. В итоге автомат окажется в конфигурации (q_0, Z), то есть слово $a_1 \dots a_{n^1}$ будет принадлежать сверхитерации данного языка.

На каждом шаге построения этого слова доля угаданных символов, очевидно, не превосходит $\frac{t^*}{r_1+t^*} \leqslant \frac{R}{r_1+R}$. Причем эта оценка верна и для каждого префикса слова.

Будем строить сверхслово дальше таким же образом, только добавляя неугаданное слово длины r_i на каждом отрезке $a_{n^{i-1}+1} \dots a_{n^i}$.

Зададим $r_i = R \cdot (2^i - 1)$. Тогда доля угаданных символов будет не больше $\frac{R}{r_i+R} = \frac{1}{2^i}$. Длины отрезков $n^{i+1} - n^i$ будут возрастать, а степень угадывания уменьшаться.

Докажем неравенство

$$\frac{a+b}{A+B} \leqslant \frac{1}{2}(\frac{a}{A} + \frac{b}{B}),$$

верное при $A < B, \frac{a}{A} \geqslant \frac{b}{B}$. Или, что то же самое, $2aB^2 + 2bA^2 \geqslant ABa + ABb + aB^2 + bA^2$.

Дано $\frac{a}{A} \geqslant \frac{b}{B}$.

Домножим обе части на положительный множитель $B - A$ и приведем к общему значенателю: $aB(B-A) \geqslant Ab(B-A)$. Раскрыв скобки, получим требуемое неравенство.

Если u_i — число угаданных символов в отрезке слова $a_{n^{i-1}+1} \dots a_{n^i}$, то $\overline{c}^{\mathfrak{B}} = \overline{\lim}_{t\to\infty} \frac{u_1+\ldots+u_t}{n^t}$.

Применяя неравенство и то, что $\frac{u_i}{n^i - n^{i-1}} \leqslant \frac{1}{2^i}$, получим

$$\overline{c}^{\mathfrak{B}} \leqslant \lim_{t\to\infty} (\frac{u_1}{2^{t-1}n^1} + \frac{u_2}{2^{t-2}(n^2-n^1)} + \ldots + \frac{u_{t-1}}{2(n^{t-1}-n^t)} + \frac{u_t}{n^t}) \leqslant$$
$$\leqslant \lim_{t\to\infty} (2^{-t} + \ldots + 2^{-t}) = \lim_{t\to\infty} \frac{t}{2^t} = 0.$$

Теорема 3 доказана.

Доказательство теоремы 4.

Рассмотрим три случая.

1)
$$\frac{l}{l+1} = \min\{\frac{l}{l+1}, \frac{1}{R_c+1}, \frac{L-1}{L(R_c(L-1)+L)^k - 1}\}.$$

Рассмотрим следующую грамматику G'_0.

$Z \to a\underbrace{A\dots A}_{l}, Z \to b,$
$A \to b\underbrace{A\dots A}_{l}, A \to a.$

Для произвольного автомата с выходом $\mathfrak{A}$ построим полностью неугаданное слово α длины s. Тогда найдется такое слово β, что $\alpha\beta \in G_0'$. Причем $|\beta| \leqslant 1+s(l-1)$. Таким образом доля угаданных символов не превосходит

$$\frac{|\beta|}{s+|\beta|} \leqslant \frac{1+s(l-1)}{s+1+s(l-1)} = \frac{l-1+\frac{1}{s}}{l+\frac{1}{s}} \leqslant \frac{l}{l+1}.$$

Поэтому для слова $(\alpha\beta)^\infty \in (L(G_0'))^\infty$ верно

$$\overline{c}^{\mathfrak{A}}((\alpha\beta)^\infty) \leqslant \frac{l}{l+1}.$$

Так как предложенный в теореме 3 автомат угадывает это множество со степенью, не меньшей $\frac{l}{l+1}$, утверждение доказано.
2)
$$\frac{1}{R_c+1} = \min\{\frac{l}{l+1}, \frac{1}{R_c+1}, \frac{L-1}{L(R_c(L-1)+L)^k-1}\}.$$

Рассмотрим грамматику G_0''.

$$\begin{aligned}
&Z \to aA_1, bA_1,\\
&A_i \to aA_{i+1}, A_i \to bA_{i+1}, i = 1, \dots, R_c - 1,\\
&A_{R_c} \to a, A_{R_c} \to bB,\\
&B \to a, B \to bBB.
\end{aligned}$$

Для любого $\mathfrak{B}$ можно построить полностью неугаданное слово длины R_c, затем, если автомат выдаст b, то слово данной грамматики будет полностью неугаданным. Если будет выдана a, то сверхитерацией такого слова получится $\alpha \in (L(G_0''))^\infty$, такое что $\overline{c}^{\mathfrak{B}}(\alpha) \leqslant \frac{1}{R_c+1}$.

Так как для автомата из доказательства теоремы 3 выполнено $c^{\mathfrak{A}}(L(G_0'')^\infty) \geqslant \frac{1}{R_c+1}$, утверждение доказано.

3)

$$\frac{L-1}{L(R_c(L-1)+L)^k-1} =$$
$$= \min\{\frac{l}{l+1}, \frac{1}{R_c+1}, \frac{L-1}{L(R_c(L-1)+L)^k-1}\}.$$

Рассмотрим грамматику G_0'''.

$$Z \to a\underbrace{A_2^1 \dots A_2^1}_{L}, Z \to bC,$$
$$C \to a, \to bCC,$$
$$A_j^i \to aA_j^{i+1}\underbrace{A_{j+1}^1 \dots A_{j+1}^1}_{l-1}, A_j^i \to bA_j^{i+1}\underbrace{A_{j+1}^1 \dots A_{j+1}^1}_{l-1},$$
$$i = 1, \dots, R_c - 1, j = 2, \dots, k-1,$$
$$A_j^{R_c-1} \to aA_j^1\underbrace{A_{j+1}^1 \dots A_{j+1}^1}_{l-1}, A_j^{R_c-1} \to b\underbrace{A_{j+1}^1 \dots A_{j+1}^1}_{l},$$
$$j = 2, \dots, k-1,$$
$$A_k^i \to aA_k^{i+1}, A_k^i \to bA_k^{i+1}, i = 1, \dots, R_c - 1,$$
$$A_k^{R_c} \to a, A_k^{R_c} \to b.$$

Рассмотрим работу произвольного автомата $\mathfrak{B}$ на словах этой грамматики. Если выход в начальный момент времени b, то подадим a и полностью неугаданное слово α длины $\sum_{i=0}^{k-1} L(R_c+1)(R_c(L-1)+L)^i$, и тогда $a\alpha$ будет полностью неугаданным. Если же выход a, то для слова $a\alpha$ доля угаданных символов — $\frac{L-1}{L(R_c(L-1)+L)^k-1}$.

А так как для автомата $\mathfrak{A}$

$$\overline{c}^{\mathfrak{A}}((a\alpha)^\infty) \geqslant \frac{L-1}{L(R_c(L-1)+L)^k-1},$$

теорема 4 доказана.

Список литературы

[1] Трахтенброт Б. А., Бардзин Я. М. Конечные автоматы (поведение и синтез) // М.: Наука, 1970.

[2] Вереникин А. Г., Гасанов Э. Э. Об автоматной детерминизации множеств сверхслов // Дискретная математика. 2006. Т. 18, № 2. С. 84–97.

[3] Мастихина А. А. О частичном угадывании сверхслов // Интеллектуальные системы. 2007. Т. 11, вып. 1–4. С. 609–619.

[4] Мастихина А. А. Критерий частичного предвосхищения общерегулярных свехсобытий // Дискретная математика. 2011.

[5] Ахо А., Ульман Дж. Теория синтаксического анализа, перевода и компиляции / Пер. с англ. М.: Мир, 1978.

[6] Мастихина А. А. Частичное угадывание некоторых контекстно-свободных языков // Материалы XVIII Международной конференции студентов, аспирантов и молодых учёных «Ломоносов».

[7] Кудрявцев В. Б., Алешин С. В., Подколзин А. С. Введение в теорию автоматов // М.: Наука, 1985.