ПРОБЛЕМЫ ПЕРЕДАЧИ ИНФОРМАЦИИ

Том XII 1976 Вып. 1

УДК 62-507

МНОГОУРОВНЕВЫЕ МАГАЗИННЫЕ АВТОМАТЫ

А. Н. Маслов

Строится последовательность классов автоматов с памятью, являющейся расширением магазинной памяти, таким образом, что класс языков, допустимых автоматами из i-го класса, совпадает с обобщенными индексными языками уровня i.

С помощью порождающих грамматик в работе [1] введены индексные языки, а в работах [2, 3] введена иерархия обобщенных индексных языков, вторым членом которой являются индексные языки, а первым — бесконтекстные языки. Известно, что класс бесконтекстных языков совпадает с классом языков, допустимых магазинными автоматами. В настоящей статье строится последовательность классов автоматов с памятью, являющейся расширением магазинной памяти, таким образом, что класс языков, допустимых автоматами из i-го класса, совпадает с обобщенными индексными языками уровня i.

Магазинной памятью уровня 1 является обычный магазин (из символов), а магазинной памятью уровня i является магазинный список памятей уровня $i-1$. Действия с магазинной памятью уровня 1 описаны в [4, 5]. Над магазинной памятью уровня i можно совершить следующие действия: 1) все допустимые действия над ее вершинной памятью уровня $i-1$; 2) заменить вершинную память уровня $i-1$ на две с ней совпадающие. При опустошении вершинной памяти уровня $i-1$ автомат переходит к следующей памяти уровня $i-1$.

Память уровня i удобно рассматривать расположенной в i-мерном пространстве так, чтобы элементы магазинного списка (памяти уровня $i-1$) занимали параллельные гиперплоскости; при действии 2 происходит дублирование вершинной гиперплоскости.

Предварительное сообщение о результатах настоящей статьи имеется в [2], сходные конструкции автоматов рассматривались в [6].

В первом разделе вводятся вспомогательные классы автоматов и устанавливается их эквивалентность классам индексных грамматик, с помощью этого во втором разделе излагаются основные результаты.

1. Для определения класса индексных грамматик нам потребуется операция возведения в степень на множестве языков. Обозначим через $V^V=\{A^B|A,\ B\in V\}$ множество всех выражений вида A^B. Доопределим операцию возведения в степень до операции над языками с помощью равенств: $A^\varepsilon=A$, $\varepsilon^A=\varepsilon$, $(L_1\cup L_2)^L=L_1{}^L\cup L_2{}^L$, $(L_1L_2)^L=L_1{}^LL_2{}^L$, $A^{L_1\cup L_2}=A^{L_1}\cup A^{L_2}$ и $A^{BL}=(A^B)^L$, где A и B — символы из алфавита V, а L_1, L_1 и L_2 — языки в алфавите V. Указанные равенства позволяют по интерпретации для выражений A^B (т. е. по отображению $V^V\to V$) определить язык $L_1{}^{L_2}$ для языков L, и L_2 в алфавите V. Тем самым, в частности, определена операция возведения в степень для слов. В выражении A^B букву B будем называть индексом буквы A. Теперь можно определить класс индексных языков, используя несколько более удобные обозначения, чем в [1].

Определение [2]. Индексной грамматикой называется пятерка $G=\langle\Sigma, V, P, I, S_0\rangle$, где $\Sigma=\{\sigma_1, \dots, \sigma_d\}$ — терминальный алфавит, $V=\{S_0, S_1, \dots, S_m\}$ — нетерминальный алфавит, S_0 — начальный символ, P — конечное множество продукций вида $S_i\to\omega$, $\omega\in(V^V\cup V\cup\Sigma)^*$ и $I: V\xrightarrow{V}V$ — интерпретация операции возведения в степень.

Слово y выводимо из слова x применением продукции $p\in P$ (обозначается $x\overset{p}{\vdash}y$), если $x=aS_i{}^zb$, $y=a\omega^zb$, $p: S_i\to\omega$. Слова a, b и z в произвольном алфавите. Слово y выводимо из слова x, обозначается $x\vdash y$, если существует последовательность слов $x=y_0, y_1, \dots, y_n=y$, такая, что либо $y_i\overset{p}{\vdash}y_{i+1}$ для $p\in P$, либо y_{i+1} получается из y_i заменой некоторого вхождения A^B на $I(A^B)$. Каждая грамматика G определяет язык $L(G)=\{x\in\Sigma^*|S_0\vdash x\}$ выводимых в ней слов. Отметим, что выражения σ^A и A^σ, $\sigma\in\Sigma$, являются тупиками.

Определение выводимости в индексной грамматике допускает естественное расширение. Пусть $V(1)=V^*$, $V(2)=(V^V\cup V)^*$, $V(k)=\{A^x|x\in V(k-1)\}^*$ и $V(\infty)=\bigcup_k V(k)$. Например, $A^{BCD}B^F\in V(3)$ и $A_1^{A_2\dots A_k}\in V(k)$.

Слово y выводимо из слова x применением правила $p\in P$ на уровне $k>1$ (обозначается $x\overset{p,k}{\vdash}y$), если существуют слова a_0 и b_0 из $(V(\infty)\cup\Sigma)^*$, a_{i+1}, b_{i+1}, z_i и t_i из $V(\infty)$, такие, что $x=a_0S_{i_0}{}^{z_0}b_0$, $y=a_0S_{i_0}{}^{t_0}b_0$, $z_j=a_{j+1}S_{i_{j+1}}^{z_{j+1}}b_{j+1}$, $t_j=a_{j+1}S_{i_{j+1}}^{t_{j+1}}b_{j+1}$ при $0\leqslant j<k-1$, $z_{k-1}=a_kS_{i_k}^{z_k}b_k$, $t_{k-1}=a_k\omega^{z_k}b_k$, $p: S_{i_k}\to\omega$. Будем отождествлять выводимость $\overset{p}{\vdash}$ и выводимость $\overset{p,1}{\vdash}$. Неформально, выводимость на уровне $k>1$ означает применение продукции к нетерминалу, являющемуся индексом k-го уровня. Слово y выводится из слова x на уровне k, обозначается $x\overset{k}{\vdash}y$, если существует последовательность слов $x=y_0$, $y_1, \dots, y_n=y$, $y_i\in(V(k)\cup\Sigma)^*$, такая, что либо $y_i\overset{p',k}{\vdash\!-}y_{i+1}$ для $p\in P$ и $k'\leqslant k$, либо y_{i+1} получается из y_i заменой обозначения A^B на $I(A^B)$ на максимальном уровне, т. е. B не имеет индекса. Ясно, что $\vdash$ и $\overset{2}{\vdash}$ эквивалентны, а выводимость $\overset{1}{\vdash}$ эквивалентна выводимости в бесконтекстной грамматике. Пару (G, k) будем называть индексной грамматикой уровня k. По грамматике (G, k) легко построить грамматику (G', k), порождающую тот же язык, что и (G, k), такую, что в выводах грамматики G' продукции (вида $A\to BC$) на уровне k не применяются.

Язык $L(G, k)=\{x\in\Sigma^*|S_0\overset{k}{\vdash}x\}$, образованный из слов, выводимых на уровне k, назовем индексным языком уровня k, а язык $L(G, \infty)=\{x\in\Sigma^*|\exists k(S_0\overset{k}{\vdash}x)\}$ индексным языком уровня ∞. Через L_k будем обозначать класс индексных языков уровня k, а также класс индексных грамматик уровня k, если это не приводит к двусмысленности.

В ряде случаев элементы множества $V(\infty)$ будем называть словами. При этом элементы множества $V(1)$ будем называть одноэтажными словами, а элементы множества $V(k)$ k-этажными словами. Пустое слово будем обозначать ε, длину (одноэтажного) слова x будем обозначать $l(x)$, а пустое множество символом $\varnothing$. Через $\langle x\rangle$ обозначим множество символов, входящих в одноэтажное слово x.

Определение. Левыми нетерминалами в слове xA^yz, где $x\in\Sigma^*$, а $A\in V$ назовем A и все левые нетерминалы из y. Таким образом, в k-этажном слове может быть k левых нетерминалов. Назовем вывод в индексной грамматике левым, если правила вывода применяются только к левым нетерминалам.

Ясно, что из каждого вывода перестановкой применения правил можно получить левый вывод, приводящий к тому же результату.

В промежуточных словах выводов индексных грамматик уровня k один и тот же нетерминал может встречаться на разных уровнях. При $k<\infty$ можно, однако, по данной индексной грамматике уровня k построить эквивалентную ей такую, чтобы каждый нетерминал мог встречаться только на каком-то одном своем уровне. Для этого каждый нетерминал A заменим на k нетерминалов $A(1), \ldots, A(k)$, а каждую продукцию $A \to \omega$ на k продукций $A(i) \to \omega(i)$, $1 \leqslant i \leqslant k$, где $\omega(i)$ получено из ω заменой каждого вхождения нетерминала B на уровне j (относительно ω) на $B(i+j-1)$; если для некоторого i и некоторого вхождения B выполнено $i+j-1>k$, то продукция $A(i) \to \omega(i)$ ликвидируется. Каждое интерпретационное правило $I(A^B)=C$ заменим на $k-1$ правило $I(A(i)^{B(i+1)})=C(i)$, где $1 \leqslant i \leqslant k-1$. После указанного преобразования нетерминальный алфавит будем называть разделенным.

Класс бесконтекстных языков совпадает с классом языков, допускаемых недетерминированными магазинными автоматами [4, 5]. Аналогичную роль для индексных языков уровня $i \geqslant 2$ играют магазинные автоматы уровня $i \geqslant 2$.

Действия магазинного автомата (с одним состоянием) совпадают с левым выводом в бесконтекстной грамматике, продукции которой приведены к виду: $A \to BC$ или $A \to \sigma$. В этом случае нетерминальную часть промежуточного слова при левом выводе можно рассматривать как содержимое магазина (точнее магазинного списка). Продукции индексной грамматики можно привести (как будет доказано в теореме 1) к виду $A \to B^C$, $A \to BC$ или $A \to \sigma$. Будем считать, что нетерминальный алфавит разделен по уровням. Промежуточное слово при левом выводе имеет вид $\sigma_{j_1} \ldots \sigma_{j_n} A_{i_1}^{z_1} \ldots$ $\ldots A_{i_m}^{z_m}$, $z_i \in V(\infty)$. Нетерминальная часть должна естественным образом располагаться в памяти допускающего автомата.

Память уровня 1 — это магазин. Автомат может совершать с магазином следующие действия: 1) прочитать очередной символ на входной ленте и символ в вершине магазина, а затем стереть последний; 2) прочитать и заменить на два символа верхний символ в магазине. Память уровня 2 — это магазинный список пар вида: «активный символ — магазинный список символов». Автомат работает с вершинной парой магазина. При опустошении вершинной пары автомат переходит к следующей паре. Автомат может совершать следующие действия, возможность выполнения которых зависит от читаемых символов: 1) если магазин в вершинной паре пуст, то прочитать очередной символ на входной ленте и активный символ, а затем стереть вершинную пару; 2) прочитать активный символ и заменить вершинную пару на две отличающихся от первоначальной лишь активным символом; 3) прочитать активный символ и заменить его, добавив при этом символ в вершину магазина символов; 4) прочитать символ и символ в вершине магазина, а затем стереть символ в вершине магазина и изменить активный символ. В левом выводе активные символы соответствуют первому уровню, а остальные символы памяти — второму уровню.

Память уровня k — это магазинный список пар вида: «активный символ — память уровня $k-1$». Автомат работает с вершинной парой, а при ее опустошении переходит к следующей паре. Автомат может совершать (недетерминированно) следующие действия, возможность выполнения которых зависит от читаемых символов (символ на входной ленте не читается, если не указано обратное);

1) если память уровня $k-1$ в вершинной паре пуста, то прочитать оче-

редной символ на входной ленте и активный символ, а затем стереть вершинную пару;

2) прочитать активный символ и заменить вершинную пару на две, отличающихся от первоначальной лишь активным символом;

3) прочитать активный символ и изменить его, добавив при этом вершинный символ в память уровня $k-1$ (т. е. в вершину магазина уровня $k-1$ добавить пару «некоторый символ — пустая память уровня $k-2$»);

4) прочитать активный символ (вершинной пары) и активный символ ее памяти уровня $k-1$, если при этом память уровня $k-2$ (в вершине памяти уровня $k-1$) пуста, то стереть вершинную пару в памяти уровня $k-1$ и изменить прочитанный активный символ;

5) совершить одно из действий 2—5 с памятью уровня $k-1$, находящейся в вершинной паре.

В начальный момент в магазине хранится единственная пара «начальный символ — пустая память уровня $k-1$». Если, прочитав слово, автомат опустошил память, слово считается допустимым. Ясно, что левый вывод в грамматике уровня $k \geqslant 1$ и допустимость соответствующим автоматом (с одним состоянием), использующим память уровня $k \geqslant 1$, тождественны.

Назовем грамматики эквивалентными, если они порождают один и тот же язык.

Будем говорить, что грамматика в приведенной форме, если ее продукции имеют вид $A \to B^C$, $A \to BC$, $A \to \sigma$ и, возможно, $S_0 \to \varepsilon$, где S_0 — начальный символ, не встречающийся в правых частях продукций и интерпретационных правил.

Теорема 1. *По индексной грамматике (G, k) можно построить эквивалентную индексную грамматику (G', k) в приведенной форме.*

Доказательство. Не уменьшая общности, можно считать, что G приведена к виду, указанному в лемме 3 из [3], т. е. ее продукции имеют вид $A \to B^C$, $A \to BC$, $A \to \sigma$ или $A \to \varepsilon$.

Преобразование грамматики (G, k) к (G', k_1) состоит из следующих этапов:

1) Произведем разделение нетерминального алфавита (см. выше). После этого множества V_i символов, которые могут появляться на i-уровне, станут пересекающимися. Полученный нетерминальный алфавит будем обозначать через V.

2) На всех уровнях, выше первого, заменим применения продукций $A \to \varepsilon$ на $A \to E$, где E — новый нетерминал, обладающий свойством $I(B^E) = B$ и $I(E^B) = E$ для каждого нетерминала B.

3) Нетерминалу A на первом уровне может быть придано в качестве параметра (обозначается $[A, N]$) некоторое подмножество N первоэтажных нетерминалов, каждый из которых под влиянием находящихся над ним индексов имеет возможность перейти в пустое слово (в противном случае вывод заходит в тупик).

4) Если при некотором применении продукции $A \to BC$ на первом этаже в выводе грамматики G предполагается, что B или C в дальнейшем перейдет под влиянием находящихся над ним индексов в пустое слово, то добавим это B или C к параметру другого, т. е. применим продукцию $[A, N] \to [C, B \cup N]$ или $[A, N] \to [B, C \cup N]$. Введем также продукции $[A, N] \to [B, N][C, N]$, $[A, \varnothing] \to A$ и $A \to [A, \varnothing]$, где A, B и C нетерминалы из V_1, а $N \in 2^{V_1}$.

5) Добавим второэтажные нетерминалы: (N_1, N_2), $[N_1, B, N_2]$, $\{B, N_3\}$, $\bar{B}$ и $\bar{\bar{B}}$, где N_i — подмножества первоэтажных нетерминалов из V, а B — имевшийся второэтажный нетерминал. Добавим продукции $[A, N] \to B^{\{C, N\}}$ и $\{C, N\} \to C(\varnothing, N)$, если в G имелась продукция $A \to B^C$, позволяющие на

период времени от порождения индекса C до исчезновения всех его наследников избавиться от параметров.

6) Введем продукции $B \to \overline{\overline{B}}\overline{B}$ и $\overline{\overline{B}} \to (N_1, N_2)[N_1, B, N_2]$, где B — второэтажный нетерминал, а N_i — подмножества первоэтажных нетерминалов. Введем интерпретационные правила

$$I([A, N_1]^{(N_1, N_2)}) = [A, N_2] \text{ и } I((N_1, N_2)^z) = (N_1, N_2)$$

(позволяющие, кроме прочего, ликвидировать последствия продукций из 5). В подслове $[A, N_1]^{B^\omega}$ применение введенных продукций и интерпретационных правил обозначает, что B, используя ω, переводит * N_1 в N_2, т. е.

$$\exists x \exists y \left(\langle x\rangle = N_1 \& \langle y\rangle = N_2 \& x^{B^\omega} \vdash_G^k y\right).$$

7) Правила (продукции и интерпретационные) для обращения с нетерминалами $[N_1, B, N_2]$ совпадают с введенными в теореме 2 из [3], но только продукции $X \to \varepsilon$ заменяются на $X \to E$, а знак $\subset$ заменяется на $=$. Нетерминал $\overline{B}$ преобразуется, используя те же правила, которые были применимы к B до п. 6, но черта сохраняется над всеми его наследниками. Соответствующие интерпретационные правила имеют вид $I([A, N]^{\overline{B}}) = [I(A^B), N]$.

8) Добавим продукции $[A, N] \to [A, N_1]$, если

$$\exists x \exists y \left(\langle x\rangle = N \& \langle y\rangle = N_1 \& x \vdash_G^k y\right).$$

9) Теперь необходимо избавиться от продукций вида $A \to B$. Для этого заменим каждый нетерминал A на $[A] = \left\{B \mid A \vdash_G^k B\right\}$. Нетерминалами новой грамматики будут подмножества нетерминалов старой грамматики; $[S_0]$ — начальный. Продукции соответствуют продукциям старой грамматики: 1) $N \to [B][C]$, если $A \in N$ и была продукция $A \to BC$; 2) $N \to [B]^{[C]}$, если $A \in N$ и была продукция $A \to B^C$; 3) $N \to \sigma$, если $A \in N$ и была продукция $A \to \sigma$. Интерпретационные правила задаются равенствами: $I(N_1^{N_2}) = \left\{D \mid \exists A, B, C\left(A \in N_1 \& B \in N_2 \& I(A^B) = C \& C \vdash_G^k D\right)\right\}$.

10) Полученная грамматика эквивалентна грамматике (G, k) с точностью до пустого слова. В случае необходимости можно добавить новый начальный символ $\overline{S}_0$ и продукции $\overline{S}_0 \to \varepsilon$ и $\overline{S}_0 \to [S_0]$.

Теорема доказана.

2. Теперь мы покажем, что конечно-автоматное управление (левым) выводом в индексных грамматиках любого уровня не расширяет класс выводимых языков. И это позволит определить классы автоматов, допускающих индексные языки заданного уровня.

Пусть продукциям и интерпретационным правилам индексной грамматики G с разделенными нетерминалами приписаны номера $r_1, \ldots, r_n$ и имеется конечный недетерминированный автомат U с входным алфавитом $\{r_i\}$. Каждому левому выводу в индексной грамматике G соответствует строка $r_{i_1} \ldots r_{i_m}$ номеров применявшихся правил (продукций или интерпретационных правил). Будем называть вывод допустимым автоматом U, если

---

* При применении продукций п. 6 преобразуемый нетерминал B «предполагает», что под ним стоит нетерминал $[A, N_1]$, и что, используя свои индексы, преобразуемый нетерминал B сможет перевести N_1 в N_2. Если эти предположения не оправдаются, то вывод зайдет в тупик.

ему соответствует строка $r_{i_1}\dots r_{i_m}$, допустимая автоматом U. Множество слов, полученных допустимыми выводами, обозначим через $L(U, G, k)$.

Теорема 2. *Пусть G — индексная грамматика уровня k с разделенными нетерминалами, $\{r_i\}$ — множество номеров правил грамматики G, а U — конечный недетерминированный автомат с входным алфавитом $\{r_i\}$. Тогда $L(U, G, k)$ — индексный язык уровня k.*

Доказательство. Нетерминалу A на первом уровне придадим два состояния q_i и q_j и будем выводить из A, используя его индексы, такие терминальные слова, что использованная последовательность правил (точнее, их номеров) переводит q_i в q_j. При левом выводе информация о применении правила на любом уровне может беспрепятственно достичь первого уровня.

Формальное доказательство проводится следующим образом. Пусть индексная грамматика $G=\langle\Sigma, V, P, I, S_0\rangle$ в приведенной форме и автомат $U=\langle\{r_i\}, Q, \lambda, q_0, q_f\rangle$, где $\lambda : Q\times\{r_i\}\to 2^Q$, q_0 — начальное, а q_f — заключительное состояние. Введем новые нетерминалы: 1) $[q_i, A, q_j]$, где $A\in V$, $q_i\in Q$, $q_j\in Q$; 2) (q_i, q_j), где $q_i\in Q$, $q_j\in Q$; 3) D — тупиковый символ. Нетерминал $[q_0, S_0, q_f]$ начальный. Нетерминал $[q_i, A, q_j]$ обозначает, что из A вместе с его индексами в дальнейшем будет выведено терминальное слово таким образом, что строка номеров правил этого подвывода может перевести состояние q_i автомата U в q_j. Нетерминал (q_i, q_j) обозначает, что на некотором уровне (но не на первом) применено правило с номером r, таким, что $q_j\in\lambda(q_i, r)$.

Определим продукции индексной грамматики $G(U)$ уровня k, порождающей язык $L(U, G, k)$:

1) $[q_0, S_0, q_f]\to\varepsilon$, если $(S_0\to\varepsilon)\in P$ и $q_f\in\lambda(q_0, r)$, где r — номер продукции $S_0\to\varepsilon$;

2) $[q_i, A, q_j]\to[q_t, B, q_k][q_k, C, q_j]$, если $(A\to BC)\in P$, $q_k\in Q$ и $q_t\in\lambda(q_i, r)$, где r — номер продукции $A\to BC$;

3) $[q_i, A, q_j]\to[q_t, B, q_j]^C$, если $(A\to B^C)\in P$ и $q_t\in\lambda(q_i, r)$, где r — номер продукции $A\to B^C$;

4) $A\to(q_i, q_j)\omega$, если $(A\to\omega)\in P$, $q_j\in\lambda(q_i, r)$, где r — номер продукции $A\to\omega$, а $q_i\in Q$;

5) $[q_i, A, q_j]\to\sigma$, если $(A\to\sigma)\in P$ и $q_j\in\lambda(q_i, r)$, где r — номер продукции $A\to\sigma$.

Интерпретация грамматики $G(U)$ задается равенствами: 1) $I_1(A^B)=$ $=(q_i, q_j)C$, если $I(A^B)=C$ и $q_j\in\lambda(q_i, r)$, где r — номер интерпретационного правила $I(A^B)=C$; 2) $I_1([q_i, A, q_j]^B)=[q_t, C, q_j]$, если $I(A^B)=C$ и $q_t\in\lambda(q_i, r)$, где r — номер интерпретационного правила $I(A^B)=C$; 3) $I_1(A^{(q_i, q_j)})=$ $=(q_i, q_j)A$, где $A\in V$, $q_i\in Q$, $q_j\in Q$; 4) $I_1([q_i, A, q_j]^{(q_i, q_t)})=[q_t, A, q_j]$, $A\in V$, q_i, q_j и $q_t\in Q$; 5) $I_1(X^V)=D$ в тех случаях, когда I_1 еще не определена.

Левые выводы грамматики $G(U)$ моделируют левые выводы грамматики G и действия автомата U. Теорема доказана.

Рассмотрим следующее устройство. Конечный недетерминированный автомат U с входным алфавитом Σ, к которому присоединена память уровня k, причем автомат U имеет доступ к активным символам на всех уровнях. Автомат U читает входной символ (или ε) и активные символы (если вершинная память уровня i состоит из активного символа и пустой памяти уровня $i-1$, то i нетерминальных символов и $k-i$ пустых слов), затем изменяет состояние и совершает с памятью одно из действий 1—5. Заметим, что чтение активных символов можно устранить (не изменяя при этом допустимый язык), заставив автомат U угадывать набор активных символов и совершая соответствующие действия с памятью; если какой-то активный символ угадан неправильно, то первое же действие с ним приведет в тупик.

Итак, предположим, что у автомата U устранены чтения активных символов, и что устройство допускает язык L. Заменим автомат U на недетерминированный автомат с одним состоянием, который может совершить под влиянием входного символа любое действие с присоединенной к нему памятью, которое может совершить U, находясь в некотором состоянии. Последовательность действий такого автомата, приводящая к допустимости слова x, в точности соответствует левому выводу в некоторой индексной грамматике G уровня k с разделенным нетерминальным алфавитом.

Пусть U' — недетерминированный конечный автомат, состояниями которого являются состояния автомата U, входной алфавит состоит из номеров правил грамматики G, его начальное состояние — начальное состояние автомата U. Под воздействием входа r автомат U' может перейти из состояния q_i в q_j, если автомат U в состоянии q_i мог совершить действие, соответствующее правилу с номером r, и перейти при этом в состояние q_j. Ясно, что $L(U', G, k)=L$. Таким образом доказана

Теорема 3. *Класс языков, допустимых автоматами (недетерминированными с конечным числом состояний) с дополнительной памятью уровня i, совпадает с классом индексных языков уровня i.*

После того как мы добавим управляющий автомат U, некоторые действия над памятью могут быть представлены в виде последовательности более простых (при увеличении числа состояний автомата U). Пусть некоторая память уровня i с активным символом A заменяется на две памяти уровня i, пусть B — активный символ вершинной из этих двух памятей, а C — активный символ второй (внутренней) из этих двух памятей. Тогда можно прочитать A и запомнить номер r продукции $A\to BC$. Затем заменить активный символ A на C. Затем произвести дублирование вершинной памяти уровня i. Затем изменить активный символ C на B и перейти в состояние, соответствующее номеру r и исходному состоянию. Таким образом дублирование можно производить вместо действия 2 (не изменяя при дублировании активные символы).

Далее, занумеруем все символы, которые могут записываться в память какого-либо автомата, и закодируем i-й символ в этой нумерации посредством 01^i0. После этого память, использующую любое конечное число символов, можно промоделировать памятью того же уровня, использующей два символа 0 и 1, заменяя запись (или считывание) символа A_i на запись (или считывание) кода 01^i0.

Память любого уровня с двумя используемыми символами, как легко проверить, удовлетворяет аксиомам **A0** и **A1** из [7], поэтому верна

Теорема 4. *Каждое из семейств языков L_k образует полное главное абстрактное семейство языков [8], и каждый язык L из L_k представим в виде $L=g(h^{-1}(Q_k)\cap R)$, где g и h — гомоморфизмы, R — регулярный язык, а Q_k — язык вход-выходных последовательностей памяти уровня k.*

В частности, Q_2 порождается грамматикой уровня 2

$$G=\langle \{a, \bar{a}, b, \bar{b}, c, \bar{c}\}, \{S_0, S, A, B\}, P, I, S_0\rangle,$$

где P содержит продукции $S_0\to aS_0^A$, $S_0\to bS_0^B$, $S_0\to cSS_0$, $S_0\to cS_0$, $S\to aS^A$, $S\to bS^B$, $S\to cSS$, $S\to\bar{c}$, $S_0\to\bar{c}$, $S_0\to\varepsilon$, а I содержит интерпретационные правила $I(S^A)=\bar{a}S$, $I(S^B)=\bar{b}S$, $I(S_0^A)=\bar{a}S_0$ и $I(S_0^B)=\bar{b}S_0$.

Отметим, что гнездный стэк [9] получается из автомата с памятью уровня 2 отождествлением совпадающих начал у составляющих магазинов.

Автор признателен А. А. Мучнику и Э. Д. Стоцкому за полезные замечания.

ЛИТЕРАТУРА

1. *Aho A. V.* Indexed Grammars – an Extension of Context-Free Grammars. J. Assoc. Comput. Machinery, 1968, 15, *4*, 647–671. (Русск. перев.: *Ахо А.* Индексные грамматики – расширение контекстно-свободных грамматик. Сб. «Языки и автоматы». М., «Мир», 1975, 130–165).
2. *Маслов А. Н.* Иерархия индексных языков произвольного уровня. Докл. АН СССР, 1974, 217, *5*, 1013–1016.
3. *Маслов А. Н.* Индексные грамматики и грамматики Вейнгаардена. Проблемы передачи информации, 1975, 11, *3*, 81–89.
4. *Гинзбург С.* Математическая теория контекстно-свободных языков. М., «Мир», 1970.
5. *Гладкий А. В.* Формальные грамматики и языки. М., «Наука», 1973.
6. *Greibach S. A.* Full AFL's and Iterated Substitution. Inform. and Control., 1970, 16, *1*, 7–35.
7. *Маслов А. Н.* Аксиоматический подход к описанию систем с управлением (в печати).
8. *Ginsburg S., Greibach S. A.,* Abstract Families of Languages. Mem. Amer. Math. Soc., 1969, *87*, 1–32. (Русск. перев.: *Гинзбург С., Грейбах Ш. А.* Абстрактные семейства языков. Сб. «Языки и автоматы», М., «Мир», 1975).
9. *Aho A. V. Nested Stack* Automata. J. Assoc. Compt. Machinery, 1969, 16, *3*, 383–406.

Поступила в редакцию
23 сентября 1974 г.