# ПОСЛЕДОВАТЕЛЬНАЯ КЛАССИФИЦИРУЮЩАЯ СИСТЕМА КАК МОДЕЛЬ НЕКОТОРЫХ ПРОЦЕССОВ ВЕРБАЛЬНОЙ СИСТЕМЫ ПАМЯТИ. II

*В. А. Ловицкий*

В работе [1] была рассмотрена последовательная классифицирующая система (ПКС), память которой представляет собой сложную древовидную структуру. Память ПКС составлена из распознающих элементов, заданных 10 свойствами. Предполагая, что возможности системы определяются структурой ее памяти и свойствами составляющих эту структуру элементов, рассмотрим вопрос о том, какие задачи может выполнять ПКС, если структура ее памяти имеет вид дерева, а распознающие элементы обладают указанными в работе [1] свойствами.

Будем говорить, что ПКС функционирует, если она каждому буквосочетанию в алфавите Г, поданному на первый вход модели, и заданию, поданному на второй вход модели, сопоставляет определенное выходное буквосочетание в том же алфавите Г.

Если распознающие элементы (РЭ), из которых составлена память ПКС, обладают свойствами 1, 4 и 5, то ПКС может функционировать только в режиме «чтения».

Будем говорить, что буква прочитана, если она возбудила соответствующий элемент 1-го слоя памяти ПКС, независимо от того, на каком месте входного буквосочетания эта буква находилась.

Будем говорить, что буквосочетание длиной $l$ прочитано, если каждая буква этого буквосочетания возбуждает соответствующий элемент соответствующего слоя памяти ПКС. Так как за последней буквой каждого буквосочетания следует пустое слово $e$, то будем считать, что все $РЭ_i$ $(i = 1, 2, \ldots, s)$, из которых сформирована память ПКС, обладают свойством 8. Иными словами, при подаче на вход $РЭ_i$ слова $e$ функционирование всей системы прекращается.

Пусть на первый вход модели подается буквосочетание длиной $l$, не содержащее буквы $a_0$, а на второй вход модели — параметр, заданный в виде предложения: «прочитать входное буквосочетание». Функционирование ПКС в этом режиме выражается следующей формулой:

$$(\exists! a_{i_1}(x)\exists! РЭ_{j_1}(x)\, B(a_{i_1}(x),\ РЭ_{j_1}(x)) \Rightarrow \exists! РЭ_{j_1}(x)\, C(РЭ_{j_1}(x))) \Rightarrow (\exists x\ \overline{\ }\ T(x, l)) \Rightarrow$$

$$\Rightarrow ((\exists! a_{i_1}(x)\exists! a_{i_2}(x+1)\, F(a_{i_1}(x),\ a_{i_2}(x+1)) \wedge \exists! РЭ_{j_1}(x) \vee РЭ_{j_2}(x+1) \times$$

$$\times D(РЭ_{j_1}(x),\ РЭ_{j_2}(x+1)) \wedge \exists! a_{i_2}(x+1)\exists! РЭ_{j_2}(x+1)\, B(a_{i_2}(x+1),$$

$$РЭ_{j_2}(x+1)) \Rightarrow \exists! a_{i_2}(x+1)\exists! РЭ_{j_2}(x+1)\, B(a_{i_2}(x+1),\ РЭ_{j_2}(x+1))) \Rightarrow$$

$$\Rightarrow \exists! РЭ_{j_1}(x)\, C(РЭ_{j_1}(x))) \vee ((\exists! a_{i_1}(x)\exists!\, a_{i_2}(x+1)\, F(a_{i_1}(x),\ a_{i_2}(x+1)) \wedge$$

$$\wedge \exists! РЭ_{j_1}(x) \vee РЭ_{j_2}(x+1)\, D(РЭ_{j_1}(x),\ РЭ_{j_2}(x+1)) \wedge \exists! a_{i_2}(x+1) \vee РЭ_{j_2}(x+$$

$$+ 1) \urcorner B(a_{i_2}(x+1),\ РЭ_{j_2}(x+1)) \Rightarrow \exists! a_{i_2}(x+1)\exists! РЭ_{j_3}(1)\, B(a_{i_2}(x+1),$$

$$РЭ_{j_3}(1)) \Rightarrow \exists! РЭ_{j_3}(1)\, C(РЭ_{j_3}(1))) \vee ((\exists! a_{i_1}(x)\exists! a_{i_2}(x+1)\, F(a_{i_1}(x),$$

$$a_{i_2}(x+1)) \wedge \exists \text{РЭ}_{i_1}(x) \forall \text{РЭ}_{i_2}(x+1) \neg D(\text{РЭ}_{i_1}(x),\ \text{РЭ}_{i_2}(x+1)) \Rightarrow$$
$$\Rightarrow \exists! a_{i_2}(x+1)\, \exists! \text{РЭ}_{i_3}(1)\, B(a_{i_2}(x+1),\ \text{РЭ}_{i_3}(1))) \Rightarrow \exists! \text{РЭ}_{i_3}(1)\, C(\text{РЭ}_{i_3}(1))))) \nabla$$
$$\nabla (\exists! x T(x,\ l) \Rightarrow ((\exists! a_{i_1}(x)\, F(a_{i_1}(x),\ e) \wedge \exists! \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1)\, D(\text{РЭ}_{j_1} \times$$
$$\times (x),\ \text{РЭ}_{j_2}(x+1)) \Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{j_2}(x+1))) \nabla (\exists! a_{i_1}(x)\, F \times$$
$$\times (a_{i_1}(x),\ e) \wedge \exists \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1) \nabla D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \Rightarrow$$
$$\Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{i_2}(x+1))))) \quad (x = 1,\ 2,\ \ldots,\ l). \qquad (1)$$

Символ разделительного «или» $\nabla$ введен для сокращенного обозначения в виде $a_i \nabla a_j$ выражения $\neg a_i \wedge a_j \vee a_i \wedge \neg a_j$.

Заметим, что как в данном режиме, так и в последующих, если выполняются условия, что $\exists! a_{i_1}(x)\, \exists! \text{РЭ}_{j_1}(x)\, B(a_{i_1}(x),\ \text{РЭ}_{j_1}(x))$ или $\exists! \text{РЭ}_{j_1}(x) \times R(\text{РЭ}_{j_1}(x))$, происходит увеличение веса возбужденного распознающего элемента. Как будет видно из последующих режимов функционирования, указанное дополнение придает ПКС черты самоизменяющейся системы.

Пусть распознающие элементы обладают свойствами 1, 4, 5, 6 и 8, а для входных сигналов соблюдаются требования 1 и 2. Согласно теореме 1 в процессе обучения формируется древовидная структура памяти ПКС, а введение свойства 6 и требования 1 позволит ПКС в результате функционирования ответить на вопрос: «является ли входное буквосочетание словом?».

Если на первый вход модели подано буквосочетание $V_k$ ($V_k \in \Theta$) длиной $l$, не содержащее буквы $a_0$, а на второй вход модели — задание «определить, является ли буквосочетание $V_k$ словом», то функционирование ПКС в указанном режиме выражается следующей формулой:

$$(\exists! a_{i_1}(x)\, \exists! \text{РЭ}_{j_1}(x)\, B(a_{i_1}(x),\ \text{РЭ}_{j_1}(x)) \Rightarrow \exists! \text{РЭ}_{j_1}(x)\, C(\text{РЭ}_{j_1}(x))) \Rightarrow$$
$$\Rightarrow (\exists x \neg T(x,\ l) \Rightarrow ((\exists! a_{i_1}(x)\, \exists! a_{i_2}(x+1)\, F(a_{i_1}(x),\ a_{i_2}(x+1)) \wedge$$
$$\wedge \exists! \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1)\, D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \wedge \exists! a_{i_2}(x+1)\, \exists! \text{РЭ}_{j_2} \times$$
$$\times (x+1)\, B(a_{i_2}(x+1),\ \text{РЭ}_{j_2}(x+1)) \Rightarrow \exists! a_{i_2}(x+1)\, \exists! \text{РЭ}_{j_2}(x+1)\, B(a_{i_2}(x+$$
$$+1),\ \text{РЭ}_{j_2}(x+1))) \Rightarrow \exists! \text{РЭ}_{j_2}(x+1)\, C(\text{РЭ}_{j_2}(x+1))) \nabla ((\exists! a_{i_1}(x)\, \exists! a_{i_2}(x+$$
$$+1)\, F(a_{i_1}(x),\ a_{i_2}(x+1)) \wedge \exists! \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1)\, D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \wedge$$
$$\wedge \exists! a_{i_2}(x+1) \forall \text{РЭ}_{j_2}(x+1) \neg B(a_{i_2}(x+1),\ \text{РЭ}_{j_2}(x+1)) \Rightarrow \neg L(V_k)) \Rightarrow$$
$$\Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{j_2}(x+1))) \nabla ((\exists! a_{i_1}(x)\, \exists! a_{i_2}(x+1)\, F(a_{i_1}(x),$$
$$a_{i_2}(x+1)) \wedge \exists \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1) \neg D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \Rightarrow$$
$$\neg L(V_k)) \Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{j_2}(x+1)))) \nabla (\exists! x T(x,\ l) \Rightarrow$$
$$\Rightarrow ((\exists a_{i_1}(x)\, F(a_{i_1}(x),\ e) \wedge \text{E}!\text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1)\, D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \wedge$$
$$\wedge \exists \text{РЭ}_{j_1}(x)\, G(\text{РЭ}_{j_1}(x)) \Rightarrow L(V_k)) \Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{j_2}(x+1))) \nabla$$
$$\nabla ((\exists! a_{i_1}(x)\, F(a_{i_1}(x),\ e) \wedge \exists \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1) \neg D(\text{РЭ}_{j_1}(x),$$
$$\text{РЭ}_{j_2}(x+1)) \wedge \exists \text{РЭ}_{j_1}(x)\, G(\text{РЭ}_{j_1}(x)) \Rightarrow L(V_k)) \Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,$$
$$\text{РЭ}_{j_2}(x+1))) \nabla ((\exists! a_{i_1}(x)\, F(a_{i_1}(x),\ e) \wedge \exists! \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1)\, D(\text{РЭ}_{j_1}(x),$$
$$\text{РЭ}_{j_2}(x+1)) \wedge \exists \text{РЭ}_{j_1}(x) \neg G(\text{РЭ}_{j_1}(x) \Rightarrow \neg L(V_k)) \Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg$$
$$\neg B(e,\ \text{РЭ}_{j_2}(x+1))) \nabla ((\exists! a_{i_1}(x)\, F(a_{i_1}(x),\ e) \wedge \exists \text{РЭ}_{j_1}(x) \forall \text{РЭ}_{j_2}(x+1) \neg$$
$$\neg D(\text{РЭ}_{j_1}(x),\ \text{РЭ}_{j_2}(x+1)) \wedge \exists \text{РЭ}_{j_1}(x) \neg G(\text{РЭ}_{j_1}(x)) \Rightarrow \neg L(V_k)) \Rightarrow$$
$$\Rightarrow \forall \text{РЭ}_{i_2}(x+1) \neg B(e,\ \text{РЭ}_{j_2}(x+1))) \quad (x = 1,\ 2,\ \ldots,\ l), \qquad (2)$$

где $L(V_k)$ — одноместный предикат, истинный тогда и только тогда, когда $V_k \in \Xi$.

Введем новые четыре свойства 2, 3, 7 и 9, т. е. будем считать, что распознающие элементы, из которых сформирована структура памяти ПКС, обладают 9 свойствами. По сути говоря, в предыдущих двух режимах уже использовалось свойство 3, т. е. в процессе функционирования происходило увеличение веса возбужденных $РЭ_j(x)$, но полученные значения $P(РЭ_j(x))$ раньше не находили применения при функционировании ПКС. Введение свойств 2, 7 и 9 позволяет использовать в процессе функционирования приобретенные значения весов $РЭ_j$. С этой же целью вводится и блок БУРЭ.

Пусть на первый вход модели подано буквосочетание $V_k (V_k \in \Theta)$ вида: $e \to a_0(1) \to a_0(2) \to \ldots$, а на второй вход модели — параметр, заданный в виде предложения: «заменять последовательно буквы $a_0(x)$ буквами $a_i(x)$ $(a_i(x) \in \Gamma)$ до тех пор, пока не получится слово» или в «результате функционирования воспроизвести слово произвольной длины». Функционирование ПКС в этом режиме можно представить в следующем виде:

$$\forall\, РЭ_{j_1}(x)\, Q(РЭ_{i_1}(x)) \Rightarrow (((\exists!РЭ_{j_1}(x)\, R(РЭ_{i_1}(x)) \Rightarrow \exists!РЭ_{i_1}(x)\, C(РЭ_{i_1}(x))) \wedge$$
$$\wedge\, \exists РЭ_{i_1}(x)\, G(РЭ_{i_1}(x)) \Rightarrow L(V_k)) \Rightarrow \forall\, РЭ_{i_2}(x+1) \neg B(e,\ РЭ_{i_2}(x+1))) \vee$$
$$\vee ((\exists!РЭ_{j_1}(x)\, R(РЭ_{i_1}(x)) \Rightarrow \exists!РЭ_{j_1}(x)\, C(РЭ_{i_1}(x)) \wedge \exists РЭ_{i_1}(x) \neg G(РЭ_{i_1}(x)) \wedge$$
$$\wedge\, \exists!РЭ_{j_1}(x) \forall\, РЭ_{j_2}(x+1)\, D(РЭ_{j_1}(x),\ РЭ_{i_2}(x+1)) \Rightarrow$$
$$\Rightarrow \forall\, РЭ_{j_2}(x+1)\, Q(РЭ_{j_2}(x+1)))) \quad (x = 1,\ 2,\ \ldots). \tag{3}$$

Введение последнего, 10-го свойства РЭ позволяет ПКС функционировать в режиме, аналогичном предыдущему, с той лишь разницей, что входное буквосочетание $V_k$ представлено в виде $e \to a_0(1) \to \ldots \to a_0(l) \to e$, а сигнал, подаваемый на второй вход модели, записывается следующим образом: «так последовательно заменять буквы $a_0(x)$ буквами $a_i(x)$ $(a_i(x) \in \Gamma)$, чтобы букву $a_0(l)$ можно было заменить конечной буквой $a_i(l)$». Иными словами, задание можно сформулировать так: «в результате функционирования воспроизвести слово длиной $l$». По этому сигналу из БУРЭ на вторые входы всех элементов, начиная с 1-го слоя и кончая $(l-1)$-м слоем, подается буква $\beta$. Функционирование ПКС в указанном режиме выражается формулой

$$\forall\, РЭ_{j_1}(x)\, Q(РЭ_{i_1}(x)) \Rightarrow (\exists x \neg T(x,\ l) \Rightarrow (\forall\, РЭ_{j_1}(x)\, S(РЭ_{i_1}(x)) \Rightarrow$$
$$\Rightarrow ((\exists!\, РЭ_{j_1}(x) R(РЭ_{i_1}(x)) \Rightarrow \exists!РЭ_{j_1}(x)\, C(РЭ_{i_1}(x))) \wedge \exists!РЭ_{j_1}(x) \forall\, РЭ_{j_2}(x+1) \times$$
$$\times\, D(РЭ_{j_1}(x),\ РЭ_{j_2}(x+1)) \Rightarrow \forall\, РЭ_{j_2}(x+1)\, Q(РЭ_{j_2}(x+1))) \vee$$
$$\vee ((\exists!РЭ_{j_1}(x)\, R(РЭ_{i_1}(x)) \Rightarrow \exists!РЭ_{j_1}(x)\, C(РЭ_{i_1}(x))) \wedge \exists РЭ_{j_1}(x) \forall\, РЭ_{j_2}(x+$$
$$+1) \neg D(РЭ_{j_1}(x),\ РЭ_{j_2}(x+1)) \Rightarrow \forall\, РЭ_{j_2}(x+1) \neg B(e,\ РЭ_{j_2}(x+1))))) \vee$$
$$\vee (\exists!xT(x,\ l) \Rightarrow (\forall\, РЭ_{j_1}(x) \neg S(РЭ_{j_1}(x)) \Rightarrow (((\exists!РЭ_{j_1}(x)\, R(РЭ_{i_1}(x)) \Rightarrow$$
$$\Rightarrow \exists!РЭ_{j_1}(x)\, C(РЭ_{i_1}(x))) \wedge \exists РЭ_{j_1}(x)\, G(РЭ_{j_1}(x)) \Rightarrow L(V_k)) \Rightarrow \forall\, РЭ_{j_2}(x+1) \neg$$
$$\neg B(e,\ РЭ_{j_2}(x+1))) \vee (((\exists!РЭ_{j_1}(x)\, R(РЭ_{j_1}(x)) \Rightarrow \exists!РЭ_{j_1}(x)\, C(РЭ_{j_1}(x))) \wedge$$
$$\wedge\, \exists РЭ_{j_1}(x) \neg G(РЭ_{j_1}(x)) \Rightarrow \neg L(V_k)) \Rightarrow \forall\, РЭ_{j_2}(x+1) \neg$$
$$\neg B(e,\ РЭ_{j_2}(x+1))))) \quad (x = 1,\ 2,\ \ldots,\ l). \tag{4}$$

С целью упрощения записи данного режима последний дизъюнктивный член формулы (4) отражает только тот факт, что если ПКС не удается с первого раза воспроизвести слово заданной длины, то на этом ее функционирование прекращается. В действительности ПКС функционирует до тех пор, пока не решит поставленную перед ней задачу. При этом в БУРЭ уже не всегда определяется для возбужде-

ния распознающий элемент, имеющий максимальный вес. Так, например, при первой неудачной попытке воспроизвести слово заданной длины в первом слое был возбужден $РЭ_{j_1}$ (1), имеющий максимальный вес по сравнению с весами $h_1 - 1$ элементов 1-го слоя. ($h_1$ — это число элементов 1-го слоя). При повторной попытке ПКС воспроизвести слово заданной длины БУРЭ возбудит в 1-м слое уже не $РЭ_{j_1}(1)$, а $РЭ_{j_2}(1)$, причем $P(РЭ_{j_2}(1)) < P(РЭ_{j_1}(1))$, но больше веса остальных $h_1 - 2$ элементов и т. д., т. е. БУРЭ будет производить последовательное возбуждение элементов по убывающим весам сначала только в первом слое, затем в 1-м, 2-м слоях и т. д. до получения слова заданной длины.

Следует отметить, что в предыдущих двух режимах при последовательной замене буквы $a_0(x)$ на букву $a_i(x)$ в БУРЭ, согласно свойству 7, формируется величина $P(РЭ_j(x))$. Эти величины, полученные для определенных элементов каждого слоя памяти ПКС, перемножаются между собой, а результат подается на выход модели. Пусть в результате замены букв $a_0(x)$ получено $l$-буквенное слово $W_k (W_k \in \Xi)$. Тогда вероятность воспроизведения этого слова $p(W_k)$ определяется в БУРЭ по следующей формуле:

$$p(W_k) = \frac{P(РЭ_{j_1}(1))}{\sum\limits_{j_1=1}^{h_1} P(РЭ_{j_1}(1))} \cdot \frac{P(РЭ_{j_2}(2))}{\sum\limits_{j_2=1}^{h_2} P(РЭ_{j_2}(2))} \cdots \frac{P(РЭ_{j_l}(l))}{\sum\limits_{j_l=1}^{h_l} P(РЭ_{j_l}(l))} = \frac{P(РЭ_{j_l}(l))}{\sum\limits_{j_1=1}^{h_1} P(РЭ_{j_1}(1))}, \quad (5)$$

где $h_1$, $h_2$, ..., $h_l$ обозначают соответственно число элементов первого слоя, число распознающих элементов, подчиненных одному из элементов 1-го слоя и т. д., и, наконец, число элементов, подчиненных одному из элементов $(l-1)$-го слоя.

Заметим, что для древовидных структур, если $\exists! РЭ_{j_1}(x) \; \forall \; РЭ_{j_2}(x+1) \; D(РЭ_{j_1}(x), РЭ_{j_2}(x+1))$, то справедливо соотношение

$$P(РЭ_{j_1}(x)) = \sum_{j_2=1}^{h} P(РЭ_{j_2}(x+1)), \quad (6)$$

где $h$ — число элементов, подчиненных $РЭ_{j_1}(x)$.

Назовем режимы функционирования ПКС, заданные формулами (1), (2), (3) и (4), соответственно первым, вторым, третьим и четвертым. Отметим еще один режим функционирования ПКС, отличающийся от предыдущих четырех тем, что при функционировании ПКС в новом режиме не используется ее структура памяти.

Пусть на первый вход модели подано буквосочетание, а на второй — задание: «сравнить между собой две буквы $a_{i_1}(x)$ и $a_{i_k}(x+1)$ входного буквосочетания». Очевидно, что эта задача не может быть решена ПКС, обладающей только рассмотренной структурой памяти. Видимо, для решения поставленной задачи ПКС должна иметь такую память, которая, во-первых, могла бы запомнить букву $a_{i_1}(x)$, во-вторых, сравнить букву $a_{i_k}(x+1)$ с запомненной буквой $a_{i_1}(x)$.

Введем конечное множество Ф распознающих элементов, каждый из которых обладает, по крайней мере, свойствами 1, 4 и 5. Заметим, что множество Ф не пересекается с конечным множеством Z. Все элементы множества Ф находятся в безразличном состоянии. При поступлении на вход модели буквы $a_i(x)$ случайным образом выбирается $РЭ_j$ из множества Ф и возбуждается этой буквой. При подаче на вход буквы $a_{i_k}(x+1)$ она либо возбуждает $РЭ_j$ (это свидетельствует о том, что $a_{i_1}(x) = a_{i_k}(x+1)$ при $k=1$), либо не возбуждает его. Если буква $a_{i_k}(x+1)$ возбуждает $РЭ_j$, то на его выходе 3 появляется буква

$a_{i_1}(x)$. После выполнения задания из БУРЭ на второй вход РЭ$_j$ подается буква $\gamma$ ($\gamma \notin \Gamma$), которая переводит его в безразличное состояние. Будем считать, что способность РЭ реагировать на букву $\gamma$ составляет 11-е свойство распознающих элементов.

Память, при формировании и функционировании которой использовались 1—10 свойства распознающих элементов, будем называть долговременной памятью (ДП) ПКС. А память, для функционирования которой достаточно, чтобы элементы обладали только свойствами 1, 4, 5 и 11, назовем кратковременной памятью (КП) ПКС. В кратковременной памяти осуществляется непродолжительное хранение информации. Длительность функционирования КП задается БУРЭ и определяется сигналом, поданным на второй вход модели. Следует заметить, что введение КП было вызвано необходимостью отразить умение человека сравнивать «незнакомые» ему символы.

Как отмечалось в работе [2], структура памяти модели состоит из нескольких уровней. Предполагается, что, используя определенный уровень памяти, модель может решать соответствующий этому уровню класс задач. Так, на первом уровне модель способна решать задачи, относящиеся соответственно к 1-му классу. К указанному классу отнесем все задачи, решение которых сводится к операциям над буквосочетаниями и не требует при этом использования связей между последними. Напомним, что буквосочетание может быть представлено в виде одной буквы или слова.

В процессе решения любой задачи 1-го класса происходит обращение к ДП или КП ПКС. Обращение к ДП осуществляется только при функционировании ПКС в одном из четырех рассмотренных режимов. Предполагается, что любая задача 1-го класса, которая решается человеком, может быть решена и ПКС, использующей для этого обращение к ДП или КП. Очевидно, что достоверность высказанного предположения доказывается только путем накопления большого количества задач, относящихся к 1-му классу, и проверки возможности их решения как человеком, так и моделью. Ниже в качестве примера приводится ряд задач, относящихся к 1-му классу.

**Задача 1.** Пусть задано буквосочетание $V_f = a_2 a_5 a_0^n a_7 a_1$, полученное из слова путем замены его $n$ букв пустыми буквами $a_0$. Необходимо восстановить исходное слово иными словами, заменить пустые буквы таким буквосочетанием $a_{i_1}^{m_1} a_{i_2}^{m_2} \ldots a_{i_k}^{m_k}$, где $i_j \in \Delta (j = 1, 2, \ldots k$ $a_{i_j} \neq a_{i_{j+1}}$, $\sum_{t=1}^{k} m_t = n$, чтобы $V_f' \in \Xi$. $V_f'$ — это буквосочетание, полученное из $V_f$ путем замены в нем пустых букв. Сформулируем условие еще одной задачи аналогичного типа.

**Задача 2.** Пусть задано буквосочетание $V_f = a_2 a_5 a_0^{n_1} a_8 a_0^{n_2} a_4 a_3$, полученное из слова путем замены его $n_1$ и $n_2$ букв соответственно пустыми буквами $a_0$. Необходимо восстановить первоначальный вид слова. Легко видеть, что варьируя количество восстанавливаемых групп и число заменяемых в них букв, можно продолжить список задач указанного типа. В процессе их решения осуществляется неоднократное обращение к ДП ПКС в режимах 2-м и 4-м. Например, очередность обращения к ДП при решении первой из рассмотренных задач можно представить в виде схемы $2 \to 4 \to 2$, где цифры 2 и 4 обозначают соответственно 2-й и 4-й режимы функционирования ПКС. В процессе решения этой задачи может происходить неоднократное обращение к ДП, т. е. $2 \to 4 \rightleftarrows 2$.

**Задача 3.** Пусть задано буквосочетание $V_f = a_i \ldots a_o^n$. Необходимо пустые буквы заменить таким буквосочетанием $a_{i_1}^{m_1} a_{i_2}^{m_2} \ldots a_{i_k}^{m_k}$,

$i_j \in \Delta (j = 1, 2, ..., k)$, $a_{i_j} \neq a_{i_{j+1}}$, $\sum_{i=1}^{k} m_i = n$, чтобы $V_f \in \Xi$. Очередность обращения к ДП в процессе решения этой задачи может быть представлена в виде схемы $2 \rightleftarrows 4$.

**Задача 4.** Пусть заданы два буквосочетания: $V_{f_1} = a_5 a_8 a_1 a_4$ и $V_{f_2} = a_5 a_8 a_2 a_4$. Необходимо сравнить их между собой и определить, что $V_{f_1} = V_{f_2}$ или $V_{f_1} \neq V_{f_2}$. При решении этой задачи используется КП ПКС.

В работе [3] приведен более полный перечень задач, которые могут быть отнесены к 1-му классу задач. Анализ решений указанного набора задач позволил предположить, что когда задача относится к 1-му классу, то для ее решения достаточно обращения к ДП или КП ПКС, причем если при решении задачи используется ДП, то очередность обращения к ней может быть задана в виде схемы, элементами которой являются режимы функционирования ПКС, а стрелки указывают на последовательность перехода от одного режима функционирования к другому.

Алгоритмы решения некоторых задач 1-го класса описаны на языке ассоциативного программирования и приведены в работе [3]. Испытание алгоритмов проводилось на ЭЦВМ «Урал-2» в вычислительном центре Харьковского института радиоэлектроники. Результаты испытания алгоритмов можно считать удовлетворительными.

## ЛИТЕРАТУРА

1. В. А. Ловицкий. Последовательная классифицирующая система как модель некоторых процессов вербальной системы памяти. I (статья в настоящем сборнике).

2. Ю. П. Шабанов-Кушнаренко, В. А. Ловицкий. Структурная модель вербальной системы памяти. Тезисы докладов на третьем съезде общества психологов СССР, т. I, М., 1968.

3. В. А. Ловицкий. Эвристическое моделирование поведения человека, определяемое уровнем организации хранимой информации. Сб. «Вопросы теории электронных цифровых математических машин», вып. 2, изд. ИК АН УССР, К., 1969.