

О. В. Бисикало, к. т. н., доц.

## ПОСТРОЕНИЕ ЦЕПИ ОБРАЗОВ В РАМКАХ МОДЕЛИ АССОЦИАТИВНОГО ОБРАЗНОГО МЫШЛЕНИЯ

*В работе представлена формализация психолингвистического понятия ассоциативно-вербальной сети на основе модели образного мышления человека. С учетом особенностей модели рассматривается метод определения пути на графе, позволяющий существенно уменьшить пространство поиска. Предлагается алгебраическая модель рекурсивного построения цепи образов по критериям минимума длины и максимума силы ассоциативной связи.*

**Ключевые слова:** образное мышление, ассоциативно-вербальная сеть, поиск в графе, рекурсивный алгоритм, алгебраическая модель.

Парадоксы ассоциативного образного мышления издавна привлекают к себе внимание исследователей, однако феноменологический характер соответствующих психофизиологических процессов существенно усложняет их формальное описание. Актуальные в настоящее время, в связи с ростом сети Интернет, разработки логических моделей лингвистического процессора уже подошли к пределу своих синтаксических и семантических возможностей. Поскольку человек лучше понимает словесные выражения за счет образного мировосприятия. **Научно важной проблемой** будем считать отсутствие методологии моделирования процессов ассоциативного образного мышления для построения лингвистического процессора [1].

**Анализ последних исследований и публикаций**, в которых предлагаются подходы к решению данной проблемы и на которые опирается автор, позволяет констатировать, в основном, концептуальный характер существующих моделей образного мышления. Наиболее близким научным инструментарием, имеющим отношение к рассмотренной проблеме, можно считать свободный ассоциативный эксперимент. Данный метод исследования связан с психолингвистическим изучением речевой деятельности человека и формированием его языковой способности. Определение ассоциативного значения слова ввел в современную научную парадигму Дж. Диз, ставший одним из наиболее авторитетных его исследователей в психолингвистике XX века. В интерпретации Дж. Диза ассоциативным считалось наиболее адекватное психологической структуре значение, иначе – “потенциальное распределение ответов (слов-ассоциатов) на определенное слово-стимул”. В процессе исследования ассоциаций Дж. Дизом подробно изучены вероятностные характеристики ассоциативных связей слова с другими словами на основе кластерного анализа [2].

Многолетние и широкомасштабные эксперименты по исследованию особенностей текстотворческого потенциала ассоциативного поля слова А. А. Залевской и группы ее сотрудников развили подход к анализу значения слова, предложенный Дж. Дизом. По мнению А. А. Залевской, этот подход себя еще далеко не исчерпал, а перспективы его последующего применения определяются возможностью интеграции в более общую теорию специфики индивидуального знания и принципов его функционирования [3]. Результаты последних исследований в данном направлении позволяют говорить и о возникновении отдельного направления в психолингвистике – ассоциативной лингвистики (концепция Ю. Г. Караулова), представленной в современном состоянии ассоциативными лексикографией, лексикологией и грамматикой [4]. Согласно этой концепции, язык может быть представлен не только в виде системы отношений, не только как большая совокупность текстов, но и в виде ассоциативно-вербальной сети (АВС), которая соотносится с речевой способностью носителя языка. АВС состоит из иерархически подчиненных уровней, которые

последовательно усложняются: слово или словосочетание – ассоциативное поле – совокупность ассоциативных полей. На основе такого подхода в пределах ассоциативной лингвистики строится ассоциативное пространство текстов, изоморфное языку. [5].

Формальные аспекты ассоциативной теории исследованы в литературе частично, однако можно считать известными базовые методы ассоциативного поиска, в частности, когда для одного элемента множества определяются наиболее близкие по силе ассоциативной связи элементы в виде хэш-таблицы. Очевидно, что пересечение списков слов-ассоциатов для двух разных элементов (узлов ABC) позволяет определить наиболее близкий между ними общий элемент [6]. Нерешенной остается задача построения цепи из трех и больше узлов ABC, которая возникает в лингвистическом процессоре при ответах на вопросы и в других задачах образного мышления [7, 8].

Введем лингвистическое понятие образа как совокупность вербальных обозначений однокорневого объекта-качества-метода [9]. Тогда ассоциативную сеть образов получаем путем такого «уплотнения» ABC, когда несколько родственных узлов объединяют в один, причем соответствующие ассоциативные связи переходят «в наследство» новому узлу. Обозначим двоичным кодом  $Bi-I_i$   $i$ -й элемент множества образов  $X$ , а ассоциативную сеть образов представим в виде пространства пар  $Assoc-Twice \subset X \times X$ . Будем считать, что в пределах модели образного мышления построен алгоритм  $Hash-Table(Bi-I_i)$ , который определяет отсортированный список образов  $List_i$ , наиболее близких к  $i$ -му образу по силе ассоциативной связи  $S_i$ . Известным можно также считать алгоритм  $Find-I(Bi-I_1, Bi-I_2)$  – поиск промежуточного или инсайтного образа-звена между образами  $Bi-I_1$  и  $Bi-I_2$ . В результате выполнения операции  $Hash-Table(Bi-I_i)$  для каждого из образов пары  $(Bi-I_1, Bi-I_2)$  определяются два отсортированных по уменьшению  $S_i$  списка ассоциируемых образов  $List_1$  и  $List_2$ , тогда оператор  $Find-I(Bi-I_1, Bi-I_2)$  находит те образы, которые присутствуют в обоих списках (пересечение двух множеств). На рис. 1 приведено изображение списков  $List_1$  и  $List_2$  в виде пузырьков, площадь которых пропорциональна силе ассоциативной связи  $S_i$ , причем общие элементы двух списков (образы-инсайты) выделены цветом.

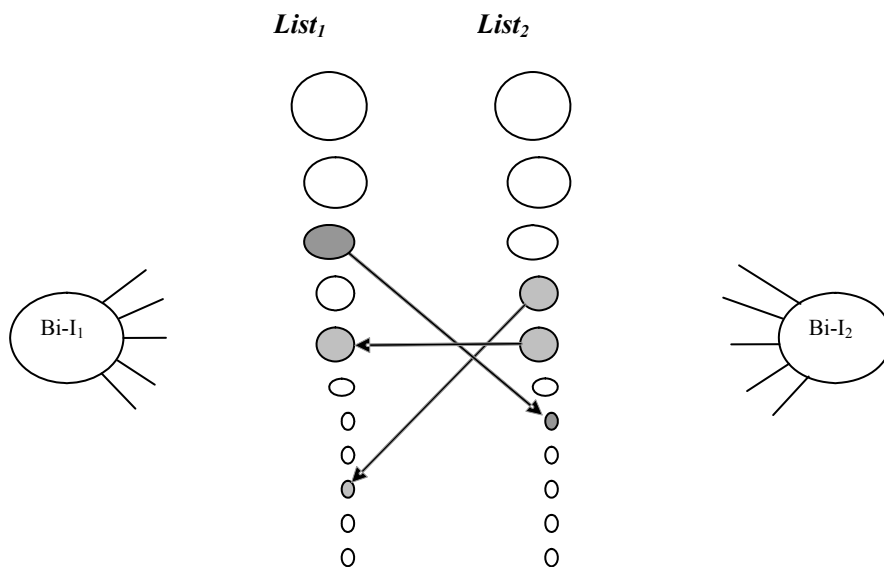


Рис. 1. Пересечение списков  $List_1$  та  $List_2$  в результате действия оператора  $Find-I(Bi-I_1, Bi-I_2)$

В этих условиях рассмотрим случай, когда необходимо найти  $n$  неизвестных звеньев в цепи образов. Задача стает актуальной тогда, когда алгоритм  $Find-I(Bi-I_1, Bi-I_2)$  не находит общих элементов в списках  $List_1$  и  $List_2$ . Формулируем **задачу исследования** следующим образом: с помощью уже известных алгоритмов  $Hash-Table(Bi-I)$ ,  $Find-I(Bi-I_1, Bi-I_2)$  найти минимальную по длине  $n$  цепь промежуточных звеньев между начальным  $Bi-I_1$  и конечным

образом  $Bi-I_2$  с максимальным суммарным весом ассоциативной связи

$$S = \sum_{i=1}^{n-1} S_i .$$

**Способ решения** поставленной задачи проиллюстрируем изображением фрагмента ассоциативной сети образов на рис. 2. Данный фрагмент сети показывает отрицательный результат начального действия оператора  $Find-1(Bi-I_1, Bi-I_2)$ , поскольку образы  $Bi-I_1$  та  $Bi-I_2$  невозможно соединить одним звеном. Если применить известные методы поиска критического пути в сетевых моделях, то, в общем случае, решение задачи требует прямого перебора размерностью  $n!$  возможных маршрутов для  $n$  узлов графа.

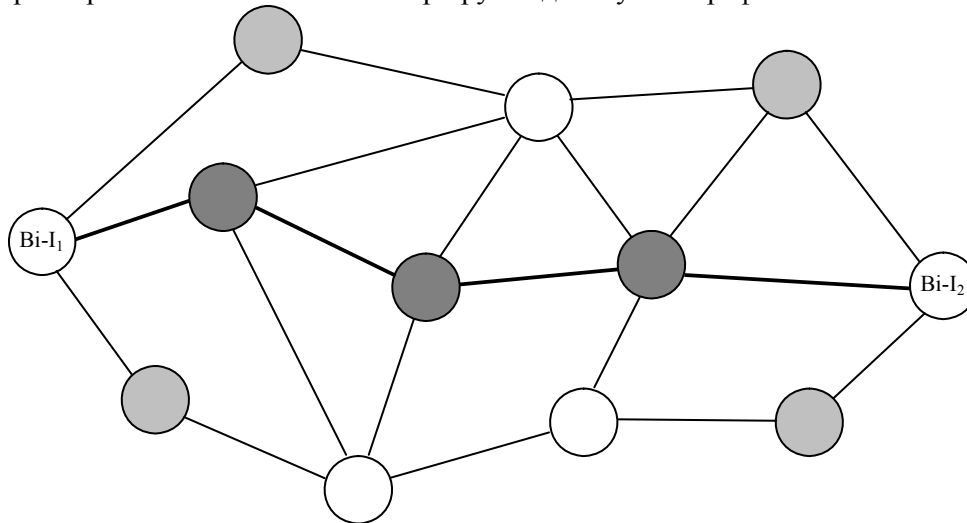


Рис. 2. Фрагмент ассоциативной сети образов с результатом действия оператора  $Find-n(Bi-I_1, Bi-I_2)$

Однако можно учесть следующие особенности графа, определяемые моделью образного мышления [10], и, тем самым, отойти от NP-полной задачи:

1. Количество значимых связей от каждого узла графа (образа) к другим узлам (образам) не больше  $7 \pm 2$ , но не меньше, чем 2.
2. Не существует двух различных узлов (образов), имеющих одинаковый набор смежных узлов (ассоциируемых образов).
3. Между любыми двумя узлами графа размерностью  $n$  обязательно существует конечный путь, длина которого не больше, чем  $(n+1)/2$ .

Предлагается следующий подход к решению поставленной задачи: строится опорный маршрут на основе рекурсивной сортировки по силе связи  $S_i$  полученного на каждом  $i$ -м шаге списка образов. Для следующего шага выбирается наилучший образ из списка за исключением тех, которые были задействованы в предыдущих итерациях. Хранение звеньев маршрута обеспечивает стек  $Route_i$ , причем внесение нового звена маршрута (образа) сопровождается проверкой на заикливание, то есть в стеке не должно быть двух одинаковых образов. Полученный опорный маршрут оптимизируется с целью уменьшения количества ассоциативных звеньев, поскольку второй целевой функцией служит лаконичность языковой конструкции (высказывания).

**Алгоритм** решения поставленной задачи построим с помощью следующих формальных операций и предикатов:

1. Операция  $Hash-Table(Bi-I_i)$  создаёт отсортированный по силе ассоциативной связи  $S_i$  список образов  $List_i$  для образа-стимула  $Bi-I_i$ :

$$(Bi - I_i, S_i) \xrightarrow{Hash-Table} List_i . \tag{1}$$

2. Предикат  $Stop$ , сигнализирующий о полном построении ассоциативного маршрута:

$$Bi - I_i = Bi - I_2 \rightarrow Stop . \tag{2}$$

3. Операция  $Find-1(Bi-I_1, Bi-I_2)$  находит промежуточные звенья  $Bi-I_x$  (образа-инсайта) для образов  $Bi-I_1$  и  $Bi-I_2$ :

$$(Bi - I_1, Bi - I_2) \xrightarrow{Find-1} Bi - I_x . \quad (3)$$

4. Предикат  $Insight$ , показывающий нахождение образа-инсайта в результате применения операции  $Find-1(Bi-I_1, Bi-I_2)$ :

$$\exists Bi - I_x \rightarrow Insight . \quad (4)$$

5. Предикат  $Cycle$ , определяющий заикливание в работе алгоритма построения цепи образов вследствие повторения в стеке  $Route_i$  одинакового образа:

$$Bi - I_i \in Route_j, j = \overline{1, i-1} \rightarrow Cycle . \quad (5)$$

Для решения поставленной задачи с учетом формальных операций и предикатов (1)-(5) представим двухосновную алгебраическую систему в виде:

$$Algebra = \langle B; \Omega \rangle , \quad (6)$$

где

$$B = \{Bi - I, S\} \quad (7)$$

основы, а

$$\Omega = \{OP, IF\} \quad (8)$$

– сигнатура системы, состоящая из операций  $OP$  та предикатов  $IF$ . В качестве основ используются  $Image$  – множество вербальных обозначений образов и  $Bi-I$  – бинарные коды элементов множества  $Image$ . В состав операций и предикатов системы включены рассмотренные ранее

$$OP = \{Hash - Table, Find - 1\}, \quad (9)$$

$$IF = \{Stop, Insight, Cycle\}. \quad (10)$$

Покажем, что задача построения цепи образов в рамках алгебраической системы  $Algebra$  фактически сводится к рекурсивному алгоритму на основе операций  $OP$  и предикатов  $IF$ . В граф-схемах алгебраических конструкторов, рассмотренных далее с этой целью, используются такие обозначения для операторов структурного программирования [11]:

- Do – цикл по параметру или по условию;
- If (+ -) – альтернатива;
- \* – композиция.

Достаточным для решения поставленной задачи является последовательное построение следующих операторов из сигнатуры (8) с учетом введенных ранее обозначений.

1. Оператор  $R_I$ , увеличивающий на 1 переменную  $i$ –номер текущего звена цепи образов и заносящий в  $i$ -ту ячейку стека  $Route_i$  значение кода конечного образа маршрута  $Bi-I_2$ :

$$R_I ::= i + 1 \rightarrow i * Bi - I_2 \rightarrow Route_i . \quad (11)$$

Граф-схема оператора  $R_I$  представлена на рис. 3.

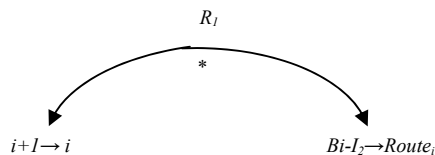


Рис. 3. Граф-схема оператора  $R_I$

2. Оператор  $R_2$ , увеличивающий на 1 переменную  $i$  и заносящий в  $i$ -ту ячейку стека  $Route_i$  значение кода образа-инсайта  $Bi-I_x$ :

$$R_2 ::= i + 1 \rightarrow i * Bi - I_x \rightarrow Route_i . \quad (12)$$

Граф-схема оператора  $R_2$  представлена на рис. 4.

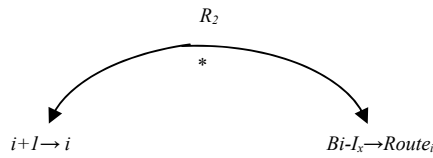


Рис. 4. Граф-схема оператора  $R_2$

3. Конечный оператор  $R_A$ , выводящий полученный маршрут или случай заикливания в виде заполненных ячеек стека  $Route_i$  от 1-й по  $i$ -ю при помощи процедуры  $Output()$ :

$$R_A ::= \{ [j = \overline{1, i}] Output(Route_j) \} . \quad (13)$$

Граф-схема оператора  $R_A$  представлена на рис. 5.

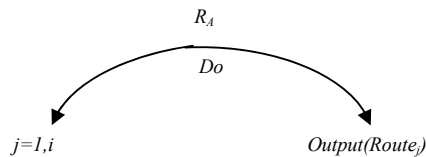


Рис. 5. Граф-схема оператора  $R_A$

4. Оператор  $Begin$ , начинающий алгоритм построения маршрута обнулением переменной  $i$ , далее заносит значение кода начального образа маршрута  $Bi-I_1$  в переменную текущего образа  $Bi-I_i$  и передает управление оператору  $Next$ :

$$Begin ::= 0 \rightarrow i * Bi - I_1 \rightarrow Bi - I_i * Next . \quad (14)$$

Граф-схема оператора  $Begin$  представлена на рис. 6.

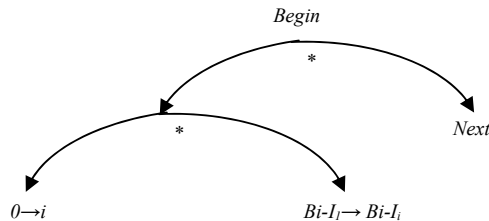


Рис. 6. Граф-схема оператора  $Begin$

5. Оператор  $Next$ , увеличивающий на 1 переменную  $i$ , далее заносит в  $i$ -ту ячейку стека  $Route_i$  значение кода текущего образа  $Bi-I_i$  и передает управление оператору  $One$ :

$$Next ::= i + 1 \rightarrow i * Bi - I_i \rightarrow Route_i * One . \quad (15)$$

Граф-схема оператора  $Next$  представлена на рис. 7.

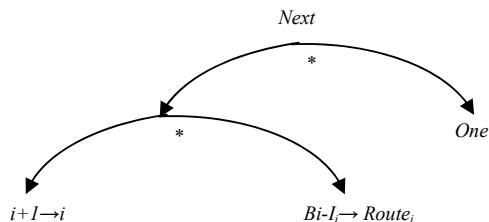


Рис. 7. Граф-схема оператора  $Next$

6. Оператор *One*, который находит список образов-ассоциатов к образу  $Bi-I_i$  при помощи операции  $Hash-Table(Bi-I_i)$  и проверяет предикат *Stop*. При условии обнаружения конечного образа маршрута выполняются операторы  $R_i$  та  $R_A$ , иначе управление передается оператору *Step*:

$$One ::= Hash - Table(Bi - I_i) * ([Stop] (R_i * R_A), Step) . \quad (16)$$

Граф-схема оператора *One* представлена на рис. 8.

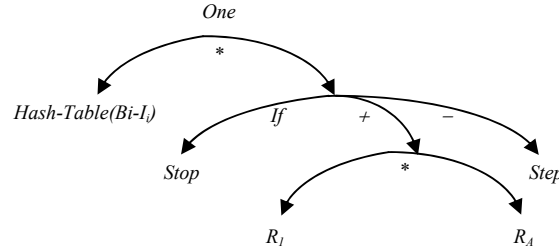


Рис. 8. Граф-схема оператора *One*

7. Оператор *Step*, который присваивает наибольшее значение из списка для  $i$ -го образа на место  $i+1$ -го звена маршрута  $Bi-I_{i+1}$  и проверяет предикат *Cycle*. При обнаружении заикливания маршрута выполняется оператор  $R_A$ , иначе управление передается оператору *Two*:

$$Step ::= Max(List_i) \rightarrow Bi - I_{i+1} * ([Cycle] R_A, Two) . \quad (17)$$

Граф-схема оператора *Step* представлена на рис. 9.

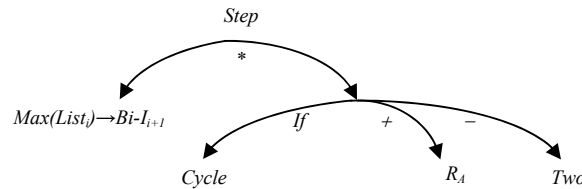


Рис. 9. Граф-схема оператора *Step*

8. Оператор *Two*, который находит образ-инсайт между текущим образом  $Bi-I_i$  и конечным образом маршрута  $Bi-I_2$ , а потом проверяет предикат *Insight*. При обнаружении образа  $Bi-I_x$  последовательно выполняются операторы  $R_2$ ,  $R_1$  и  $R_A$ , иначе управление рекурсивно переключается на операторе *Next*:

$$Two ::= Find - I(Bi - I_i, Bi - I_2) * ([Insight] (R_2 * R_1 * R_A), Next) . \quad (18)$$

Граф-схема оператора *Two* представлена на рис. 10.

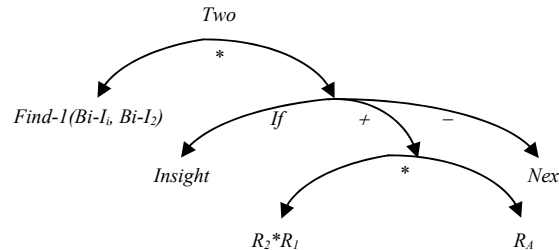


Рис. 10. Граф-схема оператора *Two*

Таким образом, предложенный рекурсивный алгоритм в качестве точки останковки имеет только оператор  $R_A$  для демонстрации как найденного маршрута, так и ситуации заикливания. Для разграничения этих двух различных случаев в дальнейшем можно усложнить алгоритм процедурой преодоления заикливания путем назначения следующего звена после проблемного  $i$ -го образа следующим по весу образом в списке  $List_i$ . В силу

особенностей графа, которые исходят из модели образного мышления, новый конечный маршрут также существует.

Полученный с помощью алгоритма (11)÷(18) опорный маршрут построен по критерию максимума силы ассоциативной связи между его образами-звеньями. Однако нельзя утверждать, что эта образная цепь является кратчайшей из всех возможных, поскольку на каждом шаге алгоритма проверялось существование инсайтного звена только между текущим образом и конечным. Следовательно, обратное применение операции *Find*- и предиката *Insight* к парам образов цепи с номерами  $(i-1,1)$ ,  $(i-1,2)$ , ...,  $(i-1,i-4)$ ,  $(i-2,1)$ ,  $(i-2,2)$ , ...,  $(i-2,i-5)$ , ...,  $(4,1)$  потенциально может сократить маршрут. На рис. 11 последовательность анализа таких пар показана пронумерованными дугами для цепи из 7-ми образов.

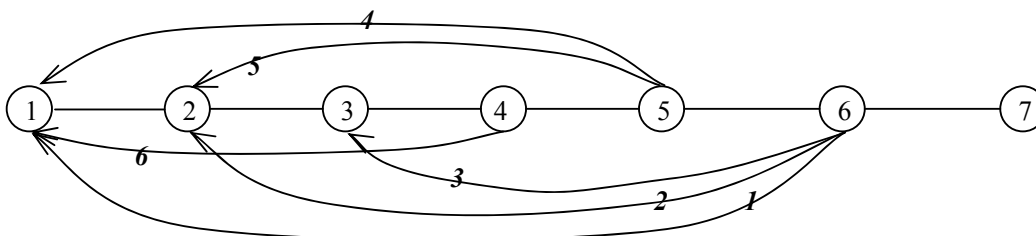


Рис. 11. Последовательность вариантов оптимизации по критерию минимума длины цепи

Характерной особенностью предложенного подхода является существенное уменьшение пространства поиска в графе, поскольку методом математической индукции легко доказать, что наибольшее количество вариантов  $V$  сокращения маршрута из  $n$  звеньев не больше арифметической прогрессии

$$V = 1 + 2 + \dots + n - 5 + n - 4 = \sum_{i=1}^{n-4} i. \quad (19)$$

**Выводы.** В работе показано, что психолингвистическое понятие ассоциативно-вербальной сети в рамках модели образного мышления формализуется в виде графа с характерными особенностями построения. С учетом этих особенностей рассмотрен метод определения в графе цепи ассоциативно связанных между собой образов по критериям минимума длины маршрута и максимума силы ассоциативной связи между образами. В отличие от известных методов решения аналогичных задач поиска кратчайших путей в графах, предложенный алгоритм моделирует процессы образного мышления человека и уменьшает пространство поиска от геометрической к арифметической прогрессии. Формализация алгоритма в виде двухосновной алгебраической системы позволяет рекурсивно строить очередные звенья цепи образов и определять главную проблему поиска – заикливание маршрута в графе. Перспективными задачами предлагаемого подхода можно считать дальнейшую алгебраическую формализацию:

- реакции системы на случаи заикливания;
- рекуррентное применение операции *Find-1c* целью минимизации длины опорного маршрута.

**Внедрение и апробация** рассмотренных моделей осуществлена в учебном процессе кафедры экономической кибернетики и информатики ВГАУ при подготовке электронных учебников по профессиональным дисциплинам. Действующий прототип системы разработан на основе технологии Python-SQLite [8], в которой соединены парадигмы объектно-ориентированного и функционального программирования с возможностями языка запросов SQL к реляционной базе данных.

#### СПИСОК ЛИТЕРАТУРЫ

1. Бисикало О.В. Принципы построения лингвистической системы на основе модели образного мышления / О.В. Бисикало // Сборник «Труды восьмой международной научно-практической конференции «Современные Наукові праці ВНТУ, 2009, № 2

информационные и электронные технологии», г. Одесса, 21-24 мая 2007 года». – Одесса: СИЭТ–2007, 2007. С.63.

2. Белянин В.П. Психолингвистика: Учебник. 5-е изд. / В.П. Белянин. – М.: Флинта, Московский психолого-социальный институт, 2008. – 232 с.

3. Залевская А.А. Значение слова и возможности его описания / А.А. Залевская // Языковое сознание: формирование и функционирование. Сборник статей / Отв. ред. Н.В. Уфимцева. – М., 1998. С. 35 – 54.

4. Караулов Ю.Н. Семантический гештальт ассоциативного поля и образы сознания / Ю.Н. Караулов // Языковое сознание: содержание и функционирование. XIII Международный симпозиум по психолингвистике и теории коммуникации: тезисы докладов, Москва, 1-3 июня 2000 г. / Редактор Е.Ф. Тарасов. – М., 2000, С. 107-108.

5. Международный симпозиум по психолингвистике и теории коммуникации: тезисы докладов, (Москва, 1-3 июня 2000 г.) / Редактор Е.Ф. Тарасов. – М.: ИРЯ РАН, 1999. – 180 с.

6. Бісікало О.В. Алгебраїчна модель лінгвістичного процесора / О.В. Бісікало // Інформаційні технології в управленні складними системами: Сборник докладов и тезисов Международной научно-практической конференции (Днепропетровск, 22-23 мая 2008). – Днепропетровск: ИТМ НАНУ и НКАУ, 2008. – С. 23-24.

7. Бісікало О.В. Конструювання образу-рішення для моделі інтелектуального управління / О.В. Бісікало // Системні технології. Регіональний міжвузівський збірник наукових праць. – Випуск 3 (56). – Том 2. – Дніпропетровськ, 2008. – С. 123-128.

8. Bisikalo O. Knowledge base of teaching system construction supported by creative thinking model / O. Bisikalo // Third International Conference “New Information Technologies in Education for All: e-education”, Proceedings (1-3 October 2008). – Kiev: Akadempriodika, 2008. – p. 413-421 (in Russian).

9. Бісікало О.В. Структура блоку пам'яті на основі моделі образного мислення людини / О.В. Бісікало // Искусственный интеллект. – 2007. – № 3. – С. 461-468.

10. Bisikalo O. Approach to the modeling of imaging mechanism of operative memory / O. Bisikalo // Second International Conference “New Information Technologies in Education for All”, Conference Proceedings (21-23 November 2007). – Kiev: Akadempriodika, 2007. – p. 336-344 (in Russian).

11. Цейтлин Г.Е. Введение в алгоритмику / Г.Е. Цейтлин. – Киев: издательство “Сфера”, 1998. – 310 с.

**Бісікало Олег Владимирович** – к. т. н., доцент кафедри економічної кібернетики и информатики, e-mail: obisikalo@gmail.com, тел.: (0432)-439370.

Винницький державний аграрний університет.