

УДК 004+004.8+528.9+681.3

## ГРАФОДИНАМИЧЕСКИЕ АССОЦИАТИВНЫЕ МОДЕЛИ И СРЕДСТВА ПАРАЛЛЕЛЬНОЙ ОБРАБОТКИ ИНФОРМАЦИИ В СИСТЕМАХ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

В.В. ГОЛЕНКОВ, Н.А. ГУЛЯКИНА

*Белорусский государственный университет информатики и радиоэлектроники  
П. Бровки, 6, Минск, 220013, Беларусь*

*Поступила в редакцию 6 января 2003*

Рассмотрены графодинамические ассоциативные модели обработки информации и соответствующие им абстрактные графодинамические ассоциативные машины, осуществляющие параллельную асинхронную обработку семантических сетей в графодинамической ассоциативной памяти. Одним из перспективных направлений применения предложенных в данной работе моделей являются интеллектуальные обучающие системы и распределенные интеллектуальные системы

*Ключевые слова:* технологии искусственного интеллекта, графодинамические ассоциативные модели, интеллектуальные системы.

### Введение

Расширение областей применения технологий искусственного интеллекта приводит к необходимости создания интеллектуальных систем принципиально нового типа, поддерживающих обработку сложноструктурированных знаний, а также легко расширяемые и модифицируемые стратегии решения неформализованных задач. Поскольку без организации параллельной обработки знаний в рассматриваемых интеллектуальных системах нового поколения невозможно рассчитывать на практически приемлемую их производительность, в основе создаваемых инструментальных средств должны лежать параллельные модели обработки сложноструктурированных знаний, параллельные языки программирования, обеспечивающие интерпретацию указанных моделей, и соответствующие им параллельные компьютерные архитектуры.

Суть развиваемого нами подхода к созданию средств построения интеллектуальных систем нового поколения заключается в ориентации на графодинамическую парадигму обработки информации, в основе которой лежит понятие графодинамической (нелинейной структурно перестраиваемой) памяти, где поддерживается развитая форма ассоциативного доступа, а обработка информации сводится не только к изменению состояния элементов памяти, но и к изменению конфигурации связей между ними [1]. Выбор графодинамической парадигмы обусловлен тем, что графодинамические модели обработки информации по сравнению с другими моделями имеют ряд преимуществ:

- поддерживают развитые формы ассоциативного доступа;
- поддерживают параллельную реализацию операций;
- поддерживают асинхронную реализацию операций;
- легко интегрируются;
- легко интерпретируются друг другом;

легко поддерживают работу со сложноструктурированной информацией любой степени сложности;

поддерживают достаточно простой переход от информации к метаинформации и согласованную обработку информации и метаинформации;

поддерживают гибкие (открытые, легко расширяемые, модифицируемые) и мощные стратегии и механизмы решения трудноформализуемых задач.

Перспективными направлениями применения предлагаемого комплекса средств проектирования интеллектуальных систем являются все те приложения, где приходится оперировать сложноструктурированными знаниями, где требуется использование сложных моделей решения задач, где необходим высокий уровень гибкости систем, где имеют место серьезные ограничения во времени. Примерами таких приложений являются:

подсистемы естественно-языкового интерфейса, для создания которых необходимы иерархические сложноструктурированные лингвистические базы знаний, переработка которых должна осуществляться в реальном масштабе времени;

интеллектуальные обучающие и тренажерные системы, семантические электронные учебники, семантические web-сайты;

распределенные интеллектуальные системы, многоагентные интеллектуальные системы, интеллектуальные системы комплексной автоматизации деятельности организаций, виртуальные организации, интеллектуальные системы управления корпоративными знаниями;

интеллектуальные системы защиты информационных ресурсов открытых корпоративных систем;

интеллектуальные геоинформационные системы;

интеллектуальные системы поддержки принятия решений в условиях дефицита времени;

интеллектуальные системы управления сложными объектами;

интеллектуальные системы поддержки проектирования сложных объектов.

### **Графовые языки**

В рамках графодинамической парадигмы предполагается создание целого семейства графовых языков различного вида, т.е. языков, текстами которых являются графовые структуры, а также использование принципа языка-ядра и ориентация на открытые языки [2].

Важным классом графовых языков являются семантические графовые языки, тексты которых называют семантическими сетями. К достоинствам семантических графовых языков представления знаний, по сравнению с другими способами представления знаний, относятся:

компактность, обусловленная тем, что в отличие от традиционного символьного текста в семантической сети знак каждого объекта или понятия описываемой предметной области присутствует только в одном экземпляре и состоит из одного символа;

ассоциативность, заключающаяся в существовании достаточно простых процедур поиска элементов семантической сети с заданными характеристиками;

наличие простой возможности введения метаинформации в семантическую сеть путем простого наращивания исходной семантической сети метасетью без какого-либо изменения исходной семантической сети;

возможность рассмотрения описываемых предметных областей одновременно на неограниченном числе уровней детализации;

приспособленность к поддержке структур любого вида, и в частности к поддержке сложноструктурированных знаний;

приспособленность к интеграции самых различных специализированных языков и самых различных моделей представления знаний;

приспособленность к представлению различного рода лингвистических знаний, что делает эффективным использование семантических графовых языков для создания естественно-языковых интерфейсных подсистем в интеллектуальных системах;

приспособленность к параллельной асинхронной обработке знаний, что делает эффективным использование семантических графовых языков для создания интеллектуальных систем, поддерживающих сложноструктурированные знания и сложные логические операции.

В основе предлагаемого комплекса графодинамических ассоциативных моделей представления и обработки знаний лежит семейство совместимых графовых языков, тексты которых представляют собой однородные семантические сети с базовой теоретико-множественной интерпретацией. В указанное семейство языков входят:

фактографический язык SCB (Semantic Code Basic);

язык SC (Semantic Code), являющийся ядром всевозможных языков представления знаний;

язык SCP (Semantic Code Programming) – процедурный язык параллельного программирования;

язык SCL (Semantic Code Logic) – логический язык графового типа;

семейство специализированных языков представления знаний;

язык SCK (Semantic Code Knowledge) – интегрированный язык представления знаний.

**Язык SCB** [2] — это фактографический язык, алфавит символов которого включает в себя:

$P$  — предметные scb-узлы;  $V$  — вторичные scb-узлы;  $B$  — scb-дуги.

Каждый scb-текст  $T$  задается тройкой  $\rightarrow PT, VT, BT \leftarrow$ ,

где  $PT$  — множество предметных scb-узлов, входящих в текст  $T$ ;  $VT$  — множество вторичных scb-узлов, входящих в текст  $T$ ;  $BT$  — множество scb-дуг, входящих в текст  $T$ .

При этом  $BT \subset VT \setminus PT \square VT \square BT$ .

Все символы, входящие в состав scb- текста  $T$ , трактуются как знаки множеств. Кроме того:

каждый символ из множества  $VT$  трактуется как знак множества scb-символов, некоторые из которых могут не входить в состав текста  $T$ ;

каждый символ  $b$  из множества  $BT$  ( $b \in BT$ ) трактуется как знак пары принадлежности, связывающей знак  $v$ , обозначающий некоторое множество scb- символов  $v \in VT$ , с scb-символом  $e$ , являющимся одним из элементов указанного множества  $b = \rightarrow v, e \leftarrow$ ;

каждый символ из множества  $PT$  трактуется как знак 1-элементного множества, при этом элементом указанного множества может быть все, что угодно, кроме множества и знака множеств.

Очевидно, что scb-текст можно трактовать как орграф, отличающийся от классического орграфа тем, что множество его узлов разбито на два класса ( $P_T$  и  $V_T$ ), а его дуги могут входить не только в узлы, но и в дуги. Кроме того, scb-текст в отличие от классического орграфа имеет четкую теоретико-множественную интерпретацию.

Особенностью scb-текстов, отличающей эти тексты от текстов традиционных языков, является их нелинейный характер. Нелинейный характер scb-текстов определяется тем, что в отличие от традиционных линейных текстов, каждый входящий в scb-текст символ может иметь более двух соседей (инцидентных ему scb-символов) – из каждого вторичного scb-узла может выходить любое количество scb-дуг и в каждый символ scb-текста может входить любое количество scb-дуг. С помощью scb-текстов легко представляются всевозможные математические структуры. С помощью специальных ключевых узлов явно задаются различные классы scb-символов, что позволяет проводить их более тонкую семантическую классификацию [2].

Важнейшим достоинством предлагаемого языка SCB является единство языка и метаязыка. Это проявляется в том, что описание связей между объектами описываемой предметной области в scb-тексте осуществляется так же, как и описание связей между scb-текстами и множествами символов scb-текстов.

Языку SCB ставится в соответствие несколько модификаций, определяемых вариантом изображения scb-символов и изображения их инцидентности. Так, например, графической модификации языка SCB соответствует алфавит графических примитивов, в котором scb-дугам соответствуют линии со стрелкой, а scb-узлам – кружки, соединяемые указанными линиями.

**Язык SC** [3] является расширением языка SCB и рассматривается как язык-ядро при разработке языков представления знаний различного вида. Алфавит символов языка SC определяется классификацией sc-символов по следующим признакам:

структурный тип sc-символа: предметный sc-узел, вторичный sc-узел, sc-дуга, sc-символ неопределенного типа;

количество значений sc-символа: sc-константа (sc-символ, имеющий единственное значение, каковым является он сам), sc-переменная (sc-символ, принимающий различные дополнительно уточняемые значения);

оценка существования множества, обозначаемого sc-символом: позитивный sc-символ, негативный sc-символ, нечеткий sc-символ.

В рамках языка SC символы языка SCB трактуются следующим образом: предметные scb-узлы – как предметные константные позитивные sc-узлы, вторичные scb-узлы – как вторичные константные позитивные sc-узлы, scb-дуги – как константные позитивные sc-дуги.

Всевозможные комбинации значений перечисленных признаков и определяют алфавит символов языка SC.

Более тонкая семантическая классификация sc-символов осуществляется с помощью ключевых sc-узлов, которые относятся к классу вторичных константных позитивных sc-узлов. К числу таких ключевых sc-узлов относятся sc-узел, обозначающий множество знаков всевозможных стационарных множеств, sc-узел, обозначающий множество знаков всевозможных нестационарных множеств, sc-узел, обозначающий множество всех точных sc-символов, sc-узел, обозначающий множество всех неточных sc-символов.

Так как алфавит символов языка SC по сравнению с языком SCB значительным образом расширен, то его графическая модификация требует введения дополнительных графических примитивов для изображения различных типов sc-символов.

Главным достоинством языка SC является то, что он представляет собой удобную основу для создания целого семейства языков, имеющих различное назначение и легко интегрируемых друг с другом. При этом построение каждого конкретного графового языка, являющегося подязыком базового языка SC, в конечном счете сводится к формированию набора ключевых узлов, обозначающих основные понятия (в том числе и метапонятия), используемые в создаваемом языке. Список этих ключевых узлов может быть легко расширен.

**Язык SCP** [4, 5, 6] является языком процедурного программирования, ориентированным на обработку однородных семантических сетей и предназначенным для эффективной интерпретации графодинамических моделей различного вида. Язык SCP относится к классу графовых языков программирования. Особенностью этого языка является то, что не только данные, но и сами программы, написанные на языке SCP, представляются в виде sc-текстов. Язык SCP является языком параллельного асинхронного программирования. Это необходимо для того, чтобы обеспечить адекватную интерпретацию не только последовательных, но и параллельных, не только синхронных, но и асинхронных формальных моделей.

Особенностями языка SCP по сравнению с другими языками указанного класса являются:

ориентация на структурно перестраиваемую (графодинамическую) память – как тексты языка SCP (scp-программы), так и перерабатываемые данные являются графовыми структурами, хранимыми в графодинамической памяти;

ориентация на ассоциативный метод доступа к хранимой в памяти информации по заданному структурному образцу;

приспособленность к обработке нечисловых структур, имеющих мощные выразительные возможности, т.е. обеспечивающих описание любых сложноструктурированных предметных областей;

ориентация на обработку непосредственно семантических сетей, а не структур, с помощью которых семантические сети кодируются в памяти традиционных компьютеров;

возможность описания (с помощью scp-программ) механизмов решения задач, имеющих различный уровень сложности и поддерживающих различные стратегии решения задач;

высокий потенциал распараллеливания процессов переработки информации в графодинамической ассоциативной памяти;

возможность интегрировать scr-программы с программами, написанными на традиционных языках и описывающими обработку традиционных информационных конструкций.

Текст языка SCP (scr-программа) с формальной точки зрения есть определенным образом устроенный sc-текст, хранимый в графодинамической памяти и описывающий некоторую систему параллельных взаимодействующих процессов переработки sc-текста, хранимого в той же памяти.

Язык представления данных для scr-программ полностью совпадает с языком SC. Данными scr-программ являются sc-конструкции произвольного вида и, в частности, конструкции любого графового языка, построенного на базе языка SC. Программа, написанная на языке SCP, может входить в состав данных по отношению к какой-либо другой scr-программе либо по отношению к самой себе.

В ядре языка SCP выделено семь классов scr-операторов. Перечислим идентификаторы ключевых узлов, соответствующих этим классам: *gen* (scr-операторы генерации sc-текстов), *erase* (scr-операторы удаления sc-текстов), *search* (scr-операторы ассоциативного поиска sc-текстов), *if* (scr-операторы проверки условий), *change* (scr-операторы изменения свойств sc-символов), *manage* (scr-операторы управления scr-процессами).

**Язык SCL** [2] является аналогом классического логического языка, но в отличие от последнего тексты языка SCL представляют собой однородные семантические сети, являющиеся текстами языка-ядра SC. Атомарная формула языка SCL трактуется как множество всех символов некоторого sc-текста. Каждая неатомарная формула языка SCL трактуется как связка, принадлежащая отношению, соответствующему типу неатомарной формулы (конъюнкция, дизъюнкция, импликация, эквиваленция, существование, всеобщность) и связывающая знаки формул, входящих в состав указанной неатомарной формулы.

**Семейство специализированных графовых языков представления знаний** [2] включает в себя язык описания динамических предметных областей, язык описания задач, в состав которого входят не только языковые средства формулировки задач, но и средства описания связей между задачами, семейство языков описания методов решения задач [7], язык спецификации методов решения задач и другие языки.

**Язык SCK** [2] является языком представления знаний, полученным в результате интеграции всех рассмотренных языков, построенных на основе языка-ядра SC.

**Принципы построения предложенного семейства графовых языков** сводятся к следующему:

- в качестве языка-ядра выделяется язык SC;
- все остальные языки (кроме SCB) имеют алфавит символов и базовую семантическую интерпретацию текстов, совпадающую с языком SC;
- уточнение синтаксиса и семантики всех языков (кроме SC и SCB) осуществляется с помощью специально вводимых ключевых узлов, имеющих соответствующую семантику;
- язык SCB есть результат сужения алфавита языка SC.

### **Абстрактные графодинамические ассоциативные машины**

Формальным уточнением различных моделей обработки информации в графодинамической ассоциативной памяти являются абстрактные графодинамические ассоциативные машины [2]. К числу указанных моделей обработки информации, в частности, относятся:

модели параллельной переработки знаний, соответствующие различным логикам, различным стратегиям решения задач;

модели, обеспечивающие реализацию хранимых в памяти программ и неформальных процедур, устроенных различными способами;

базовые модели, обеспечивающие интерпретацию всех вышеперечисленных моделей, ориентированные на реализацию в специально предназначенных для этого параллельных компьютерах и определяющие архитектуру этих компьютеров.

Указанные модели, соответствующие различным уровням параллельной интеллектуальной системы, нецелесообразно разрабатывать независимо друг от друга, так как хорошая

согласованность этих моделей, основанная на простых принципах их интеграции и интерпретации, является важнейшим фактором эффективности интеллектуальной системы в целом.

Преимущество использования графодинамических ассоциативных машин в качестве инструмента для создания интеллектуальных систем нового поколения обусловлено следующими обстоятельствами:

принципиально проще реализуется ассоциативный метод доступа к перерабатываемой информации;

существенно проще поддерживается открытый характер как самих машин, так и реализуемых на них формальных моделей;

являются удобной основой для интеграции различных моделей обработки информации.

Остальные достоинства графодинамических ассоциативных машин обусловлены достоинствами графовых текстов и графовых языков.

*Абстрактная sc-машина* [2] является формальным уточнением понятия абстрактной графодинамической ассоциативной машины, ориентированной на обработку однородных семантических сетей с базовой теоретико-множественной интерпретацией.

В абстрактной sc-машине фиксируется внутренний язык представления информации в памяти машины – язык SC. Абстрактная sc-машина  $SCm$  задается тройкой:

$$SCm = \rightarrow L, S, W \leftarrow,$$

где  $L$  — язык представления информации (в данном случае — язык SC) в памяти машины (внутренний язык);  $S$  — исходная информационная конструкция;  $W$  – множество операций над информационными конструкциями, хранимыми в памяти машины.

Каждой sc-операции (операции sc-машины) из множества  $W$  соответствует своя микропрограмма, а каждой микропрограмме — множество элементарных процессов ее выполнения. Элементарные процессы (процессы реализации операций) абстрактной sc-машины выполняются параллельно и асинхронно над общей для них памятью. Если области памяти, которые обрабатывают элементарные процессы, не пересекаются, то эти процессы никак не влияют друг на друга. Взаимодействие элементарных процессов, обрабатывающих пересекающиеся области памяти, осуществляется следующим образом. При параллельном выполнении элементарных процессов над общими фрагментами памяти sc-машины действия, выполняемые над каждым хранимым в памяти sc-символом, выполняются строго последовательно. Каждый элементарный процесс блокирует символы обрабатываемого им sc-текста с помощью различных типов блокировок. Этот принцип является основным механизмом разрешения конфликтов при интеграции различных sc-машин.

Каждой sc-операции однозначно соответствует некоторое множество sc-узлов, хранимых в памяти абстрактной sc-машины и являющихся константами всех микропрограмм, используемых для реализации указанной sc-операции. Множество всех sc-узлов, являющихся константами всех микропрограмм, используемых для реализации некоторой sc-операции, будем называть ключевыми узлами этой sc-операции. Узел, хранимый в памяти абстрактной sc-машины и являющийся ключевым по крайней мере для одной из ее операций, будем называть ключевым узлом этой sc-машины.

Очевидно, что все ключевые узлы абстрактной sc-машины должны входить в число ключевых узлов внутреннего языка этой машины. Заметим при этом, что ключевые узлы внутреннего языка абстрактной sc-машины кроме ключевых узлов самой машины могут включать в себя и другие узлы.

Поскольку разные абстрактные sc-машины отличаются разным набором операций, то и наборы ключевых узлов у них в общем случае будут разными. Следовательно, строго говоря, разные абстрактные sc-машины работают на разных внутренних языках, имеющих разные наборы ключевых узлов, но являющихся при этом подязыками одного и того же базового языка – языка SC.

Итак, для всего семейства абстрактных sc-машин фиксируются:  
ядро внутренних языков этих машин — язык SC;

принципы организации памяти, обеспечивающей хранение и обработку sc-текстов: память всех sc-машин является графодинамической и ассоциативной, поддерживающей ассоциативный доступ по заданному образцу и произвольной конфигурации;

параллельный, асинхронный характер реализации элементарных процессов над памятью;

принципы разрешения конфликтов между элементарными процессами, параллельно выполняемыми над общей графодинамической памятью;

принципы построения микропрограмм, в частности, запрещающие использование микропрограммами каких-либо вспомогательных данных, не хранимых в графодинамической памяти в виде соответствующих sc-текстов.

Таким образом, в рамках всего семейства абстрактных sc-машин не фиксируются ни конкретный подязык языка SC (разные sc-машины в качестве своего внутреннего языка могут использовать разные подязыки языка SC), ни набор операций и соответствующих им микропрограмм (разные sc-машины могут иметь разный набор операций). Общим для всех абстрактных sc-машин является только небольшой набор специальных операций, обеспечивающих просмотр (навигацию) и редактирование sc-текстов, хранимых в графодинамической памяти.

Графодинамическая абстрактная sc-машина является перспективной как в теоретическом, так и в практическом плане уточнением понятия абстрактной машины. Отметим, что абстрактные sc-машины являются мощным средством описания операционной семантики различных языков (языков программирования, языков представления знаний). Для того чтобы описать операционную семантику какого-либо языка с помощью абстрактной sc-машины, необходимо, во-первых, разработать способ отображения произвольных конструкций описываемого языка в семантически эквивалентные sc-тексты, и, во-вторых, для каждой операции абстрактной машины, определяющей операционную семантику описываемого языка, построить эквивалентную ей операцию абстрактной sc-машины, описывающую определенный класс преобразований sc-текстов, эквивалентных соответствующим конструкциям описываемого языка.

**Абстрактная scp-машина** [4, 8] — это абстрактная машина, определяющая операционную семантику языка SCP, являющаяся формальным описанием интерпретатора языка SCP и представляющая собой частный вид абстрактной sc-машины. В соответствии с назначением языка SCP абстрактная scp-машина предназначена для эффективной интерпретации всевозможных графодинамических моделей.

Перечислим общие принципы организации переработки информации в абстрактной scp-машине:

абстрактная scp-машина принадлежит к семейству абстрактных sc-машин;

абстрактная scp-машина относится к классу параллельных асинхронных sc-машин, в которых выполняемые параллельные процессы работают над общедоступной графодинамической памятью;

абстрактная scp-машина относится к классу абстрактных sc-машин, обеспечивающих реализацию (интерпретацию) хранимых в ее памяти процедурных программ – программ, представленных на графовом языке SCP.

Основными видами информации, хранящейся в памяти абстрактной scp-машины, являются реализуемые (интерпретируемые) scp-программы; sc-тексты, являющиеся данными реализуемых scp-программ; sc-тексты, являющиеся вспомогательными (специальными) данными, определяемыми и формируемыми самой scp-машиной.

Граница между scp-программами и их данными относительна, так как данными для scp-программы могут быть другие scp-программы, или даже она сама. Доступ scp-программ к вспомогательным данным абстрактной scp-машины блокируется. К этим данным имеют доступ только микропрограммы абстрактной scp-машины.

Процесс переработки информации в абстрактной scp-машине представляет собой совокупность одновременно протекающих последовательных процессов, которые будем называть **scp-процессами**.

**Абстрактная scl-машина** [9, 10, 11] является машиной логического вывода и относится к классу абстрактных sc-машин. Внутренним языком scl-машины является указанный выше

графовый логический язык SCL, ее операции соответствуют правилам логического вывода [10, 11].

Семейство специализированных абстрактных графодинамических машин обработки знаний является формальным уточнением операционной семантики указанных выше специализированных графовых языков представления знаний, каждому из которых соответствует одна или несколько абстрактных машин. Эти абстрактные машины соответствуют различным моделям решения задач, различным логикам, различным моделям правдоподобных рассуждений [12].

**Абстрактная sck-машина** — это результат интеграции всех вышеуказанных абстрактных графодинамических машин, построенных на основе абстрактной sc-машины.

Принципы реализации предложенного семейства графодинамических ассоциативных машин сводятся к следующему:

в качестве базовой графодинамической ассоциативной машины выделяется scr-машина; реализация scr-машины осуществляется либо на традиционных компьютерах, либо путем создания соответствующего графодинамического ассоциативного компьютера с нетрадиционной архитектурой;

все остальные графодинамические ассоциативные машины реализуются на базе scr-машины путем описания операций этих машин на языке SCP.

### **Реализация базовой графодинамической ассоциативной машины традиционными средствами**

Первым вариантом реализации базовой графодинамической ассоциативной машины (scr-машины) является ее моделирование на традиционных компьютерах. Основные решаемые здесь задачи следующие:

разработка модели графодинамической ассоциативной памяти;

реализация операций sc-машины над графодинамической ассоциативной памятью.

Перспективной основой реализации базовой графодинамической ассоциативной машины являются также суперкомпьютеры кластерной архитектуры.

### **Графодинамические ассоциативные компьютеры**

Очевидно, что наиболее эффективным вариантом реализации базовой графодинамической ассоциативной машины является создание специализированных графодинамических ассоциативных компьютеров с тем или иным вариантом аппаратной поддержки графодинамической ассоциативной памяти и параллельных асинхронных процессов над этой памятью [13–15]. Особый интерес представляет реализация графодинамического ассоциативного компьютера как однородной процессоро-памяти, особенностью которой является стирание границы между памятью и процессором и организация обработки информации непосредственно в памяти.

### **Базовое программное обеспечение графодинамических ассоциативных машин**

Любая машина обработки информации независимо от варианта ее реализации (на базе традиционного компьютера или путем создания компьютера с нетрадиционной архитектурой) нуждается в базовом программном обеспечении, которое превращает машину обработки информации в практически пригодный инструмент проектирования и воплощения различных систем на ее основе.

В состав базового программного обеспечения графодинамической ассоциативной машины входят: операционная среда [16, 17]; пользовательский интерфейс [18]; базовая среда программирования (на языке SCP) [4]; среда проектирования баз знаний (на языке SCK).

Особо следует отметить, что графовые языки предоставляют хорошие перспективы развития пользовательского интерфейса. Это обусловлено прежде всего возможностью наглядной визуализации семантических сетей.

## **Интеллектуальные обучающие системы и инструментальные средства их разработки, семантические электронные учебники, семантические web-сайты**

Одним из наиболее перспективных применений графодинамических моделей являются интеллектуальные обучающие системы (ИОС). Интеллектуальные обучающие системы [19–21] представляет собой сложную иерархическую систему, состоящую из совокупности взаимодействующих между собой подсистем, каждая из которых решает определенный класс задач. В качестве базового компонента интеллектуальных обучающих систем используется семантический электронный учебник, в котором база знаний учебного и учебно-методического материала представляется в виде гипермедийной семантической сети и обеспечивается возможность ассоциативного доступа к любому фрагменту этого учебного материала.

Основными функциями интеллектуальных обучающих систем являются: мониторинг деятельности обучаемых и постоянное уточнение базы знаний об обучаемых; выбор рекомендуемой последовательности изучения учебного материала; выбор рекомендуемой последовательности задаваемых обучаемому вопросов, задач и лабораторных заданий; тестирование знаний обучаемых; отслеживание прерываний в процессе обучения каждого обучаемого и обеспечение возможности возврата в прерванное состояние; управление переходами между режимами обучения.

## **Распределенные интеллектуальные системы, многоагентные системы и виртуальные организации**

Поскольку графодинамические ассоциативные модели ориентированы на параллельную асинхронную обработку баз знаний, одним из перспективных направлений их применения являются распределенные интеллектуальные системы. Активно развиваемым в настоящее время подходом к построению распределенных интеллектуальных систем являются многоагентные интеллектуальные системы, каждая из которых представляет собой коллектив достаточно самостоятельных, активных и целенаправленных интеллектуальных систем, объединяющих свои усилия в соответствии с некоторыми общими корпоративными целями [22].

Многоагентные интеллектуальные системы являются перспективной основой создания систем комплексной автоматизации деятельности современных организаций и, в частности, виртуальных организаций, особенностью которых является распределенный характер и применение современных коммуникационных технологий. Задача создания виртуальной организации рассматривается как инженерная задача проектирования социотехнической сетевой системы, в которой органически переплетается деятельность сотрудников организации и функционирование ее корпоративной системы. Проектирование виртуальной организации рассматривается нами как построение и реализация трех согласованных моделей: процессной модели организации, уточняющей структуру ее деятельности, агентно-ориентированной модели и модели управления корпоративными знаниями.

В связи с развитием технологий дистанционного обучения особое место среди виртуальных организаций учебного назначения занимают виртуальные кафедры [20, 21, 23, 24, 25]. Преобразование традиционной кафедры в виртуальную предполагает создание интеллектуальной кафедральной корпоративной системы, обеспечивающей комплексную автоматизацию всех видов деятельности кафедры, координацию взаимодействия студентов, штатных и внештатных сотрудников кафедры.

# GRAPHO-DYNAMIC ASSOCIATION MODELS AND FACILITIES OF PARALLEL INFORMATION HANDLING IN ARTIFICIAL INTELLIGENCE SYSTEMS

V.V. GOLENKOV, N.A. GULYAKINA

## Abstract

Grapho-dynamic association models of information handling and corresponding abstract grapho-dynamic association machines producing parallel asynchronous processing of semantic networks in grapho-dynamic association memory are viewed. One of the most prospective directions of offered in the article models application are intelligence coursewares and allocated intelligence systems.

## Литература

1. *Голенков В.В.* Графодинамические методы и средства параллельной асинхронной переработки информации в интеллектуальных системах. Мн.: БГУИР, 1996. 295 с.
2. Представление и обработка знаний в графодинамических ассоциативных машинах / В.В.Голенков, О.Е.Елисеева, В.П.Ивашенко и др.; Под ред. В.В.Голенкова. Мн.: БГУИР, 2001. 412 с.
3. *Голенков В.В.* Описание графового языка SC. Мн.: ИТК АН Беларуси, 1994. 86 с.
4. Программирование в ассоциативных машинах / В.В.Голенков, Г.С.Осипов, Н.А.Гулякина и др. Мн.: БГУИР, 2001. 276 с.
5. *Голенков В.В., Гулякина Н.А.* Язык параллельного программирования, ориентированный на переработку сложноструктурированных знаний в структурно перестраиваемой ассоциативной памяти. Мн.: БГУИР, 1998. 83 с.
6. *Голенков В.В., Гулякина Н.А., Елисеева О.Е.* Описание языка SCP. Мн.: ИТК АН Беларуси, 1995. 152 с.
7. *Ивашенко В.П.* Представление нейронных сетей и систем продукции в однородных семантических сетях // Изв. Белорус. инж. акад. 2003. № 1(15)/1. С.184-188.
8. *Гапонов П.А.* Методы организации параллельной переработки информации в графодинамической ассоциативной памяти // Интеллектуальные системы: Сб. науч. тр. Мн., 1998. С.167-178.
9. *Голенков В.В., Королев В.Г.* Базовые преобразования текстов языка SCL для реализации механизмов дедуктивного логического вывода. Мн.: ИТК АН Беларуси, 1996. 116 с.
10. *Голенков В.В., Королев В.Г., Елисеева О.Е.* Операции языка SCL для обработки простых запросов. Мн.: ИТК АНБ, 1995. 140 с.
11. *Голенков В.В., Королев В.Г., Елисеева О.Е.* Решение на языке SCL задач из области геометрии. Мн.: ИТК АНБ, 1995. 135 с.
12. *Хованский С.В., Ивашенко В.П.* // Изв. Белорус. инж. акад. 2003. № 1(15)/1. С.201 - 202.
13. *Голенков В.В.* // Новости искусственного интеллекта. 1993. № 4. С. 128 -148.
14. *Кузьмицкий В.М.* Принципы построения графодинамического параллельного ассоциативного компьютера, ориентированного на переработку сложноструктурированных знаний // Интеллектуальные системы: Сб. науч. тр. Мн.: ИТК АН Беларуси, 1998. С. 156-166.
15. *Кузьмицкий В.М.* Принципы построения графодинамической ассоциативной памяти // Интеллектуальные системы: Сб. науч. тр. Мн.: ИТК АН Беларуси, 1999. С. 125-133.
16. *Сердюков Р.Е.* Сегментная организация базы знаний, представленных в виде графодинамических структур // Изв. Белорус. инж. акад. 2003. № 1(15)/2. С.218-221.
17. *Кондратенко А.Л.* Протокол синхронизации фрагментов распределенной памяти для графодинамической ассоциативной машины // Изв. Белорус. инж. акад. 2003. № 1(15)/2. С.206-209.
18. *Колб Д.Г.* // Изв. Белорус. инж. акад. 2003. № 1(15)/1. С.188-191.
19. *Елисеева О.Е.* Инструментальные средства проектирования интеллектуальных обучающих систем. Мн., Вышэйш. шк., 1998. № 1. С.63-69.
20. Интеллектуальные обучающие системы и виртуальные учебные организации / В.В.Голенков, В.Б.Тарасов, О.Е. Елисеева и др.; Под ред. В.В. Голенкова и В.Б. Тарасова. Мн.: БГУИР, 2001. 488 с.
21. *Голенков В.В., Емельянов В.В., Тарасов В.Б.* Виртуальные кафедры и интеллектуальные обучающие системы // Новости искусственного интеллекта. 2001. № 4. С.3-13.
22. *Хатько А.В.* // Изв. Белорус. инж. акад. 2003. № 1(15)/2. С.120-122.
23. *Гулякина Н.А.* // Докл. БГУИР. 2003. Т.1, № 2/2. С.15-25.
24. *Лемешева Т.Л.* // Изв. Белорус. инж. акад. 2003. № 1(15)/1. С.191-194.
25. *Агашков В.В., Лемешева Т.Л.* // Изв. Белорус. инж. акад. 2003. № 1(15)/1. С.174-176.