



OSTIS-2012

(Open Semantic Technologies for Intelligent Systems)

УДК 004.8

МИВАРНЫЕ ТЕХНОЛОГИИ СОЗДАНИЯ ЛОГИЧЕСКОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА: ЛОГИЧЕСКАЯ ОБРАБОТКА С ЛИНЕЙНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТЬЮ БОЛЕЕ 3 МИЛЛИОНОВ ПРОДУКЦИЙ И ВОЗМОЖНОСТЬ ПОНИМАНИЯ СМЫСЛА ЧЕРЕЗ МОДЕЛИРОВАНИЕ БОЛЬШОГО КОНТЕКСТА

Варламов О.О.

*Московский автомобильно-дорожный государственный технический университет (МАДИ),
Московский физико-технический институт (государственный университет), г. Москва, Россия*

Ovar@narod.ru, OVarlamov@gmail.ru

Выделены 3 уровня научных исследований в области искусственного интеллекта (ИИ). Приведены результаты практических расчетов и решений задач, которые экспериментально подтвердили линейную вычислительную сложность логического вывода и/или автоматического конструирования алгоритмов в формализме миварных сетей (MIVAR net's). Программа УДАВ обрабатывает более 1,17 млн объектов и более 3,5 млн правил. Обоснован путь создания ИИ: Мивары => БД + Логика => Контекст => Смысл => Искусственный интеллект! Мивары позволят компьютерам обучаться и понимать смысл, что создаст логический искусственный интеллект. **Ключевые слова:** искусственный интеллект, логический вывод, мивар, миварные сети, экспертные системы.

ВВЕДЕНИЕ

Проблема создания интеллектуальных систем и логического искусственного интеллекта (ИИ) является актуальной и важной. Этой тематике посвящено большое количество научных работ [Поспелов, 1989], [Люгер, 2005], [Джарратано и др., 2007], [Варламов, 2002], [Когаловский, 2005], [Кузнецов, 2009], [Санду и др., 2010]. Миварный подход позволяет предложить новые модели и методы обработки информации и управления [Варламов и др., 2010], [Варламов, 2002], [Варламов, 2011], [Мивар, 2011], [Санду и др., 2010]. Под системами искусственного интеллекта будем понимать активные самообучающиеся логически рассуждающие системы. Ранее были разработаны технологии создания экспертных систем по узконаправленным предметным областям. Это обусловлено сложностями формализованного описания предметных областей и тем, что из-за факториальной вычислительной сложности системы логического вывода не могли обрабатывать много объектов/правил [Кузнецов, 2009], [Поспелов, 1989], [Люгер, 2005]. Для реализации экспертных систем, основанных на знаниях, используются продукции, логика предикатов и др. [Джарратано и др., 2007]. В то же время, получили развитие "интеллектуальные пакеты прикладных программ" (ИППП), которые позволяли решать в автоматизированном режиме задачи в разных областях, где требовались

вычисления и конструирование алгоритмов решения задач. Технологии ИППП развиваются в миварах, сервисно-ориентированных архитектурах.

Российский фундаментальный инновационный миварный подход [Варламов, 2002], [Варламов, 2003], [Варламов, 2011], [Мивар, 2011], [Санду и др., 2010] позволяет использовать эволюционные базы данных и правил для формирования единого глобального пространства накопления и обработки информации. Миварный подход позволит создать мультипредметные экспертные системы и перейти к реальной обработке больших контекстов, представляющих собой информационные модели сложных предметных областей. Уже сейчас мивары обрабатывают более 3,5 миллионов продукции. На основе миварных сетей можно создавать сложные и сверхбольшие информационные модели. Это позволит работать с адекватными контекстами и, в ближайшей перспективе позволит компьютерам "понимать смысл информации". Следовательно, тема данной работы актуальна и перспективна.

1. Миварный подход

В миварном подходе объединяются базы данных, вычисления, сервисы и логика. Мивары изначально были рассчитаны на многомерное и эволюционное представление информации для создания различных интеллектуальных систем, включая глобальные системы обучения, обработки информации и тренажеры [Варламов, 2002], [Варламов, 2003],

[Варламов, 2011], [Мивар, 2011], [Санду и др., 2010]. Информатика занимается 5 основными процессами: сбор, передача, накопление, обработка и представление информации. В настоящее время мивары из этих 5 процессов реализует два: накопление и обработку информации, а миварный подход включает две основные технологии.

1) *Миварная технология накопления информации* - это способ создания глобальных эволюционных баз данных и правил (знаний) с изменяемой структурой на основе адаптивного дискретного миварного информационного пространства унифицированного представления данных и правил, базирующегося на трех основных понятиях "вещь, свойство, отношение" (полученных на основе гносеологии и работ Райбекаса А.Я.).

2) *Миварная технология обработки информации* - это способ решения логико-вычислительных задач путем создания системы логического вывода или "конструирования алгоритмов из модулей, сервисов или процедур" на основе активной обучаемой миварной логико-вычислительной сети правил ("миварная сеть" - "MIVAR net") с линейной вычислительной сложностью.

Миварная технология накопления информации предназначена для хранения любой информации с возможным эволюционным изменением структуры и без ограничений по объему и формам представления. Миварная технология обработки информации предназначена для обработки информации, включая логический вывод, вычислительные процедуры и сервисы. Фактически, миварные сети позволяют развить продукционный подход и создать автоматическую обучаемую логически рассуждающую систему реального времени. Миварное представление логики позволило перейти от однодольных графов продукций к многодольным сетям и выполнять все операции с линейной сложностью.

Мивары быстрее, чем продукции, потому что:

1) в продукциях за основу поиска были взяты правила, которые перебирались для поиска решения, что порождало полный перебор, факториальную сложность и циклы (рисунок 1);

2) в миварных сетях явно выделены две доли: "правила" и "объекты" ("переменные"), а за основу поиска алгоритма логического вывода взяты именно "объекты", которые могут иметь только одно значение и их можно найти только один раз, что исключает циклы и полный перебор (рисунок 1);

3) Миварная сеть может быть задана двухмерной матрицей (рисунок 2), в которой каждое правило знает все свои входные и выходные объекты, а каждый объект, соответственно, знает все свои правила и свои роли в них ("вход" или "выход"), что позволяет избежать перебора и постепенно выявляя новые известные объекты через соответствующие правила, постоянно сокращать размерность исходной миварной матрицы обеспечивая линейную

вычислительную сложность логического вывода относительно общего количества правил в матрице.

При поиске логического вывода в миварной сети, как показано на рисунке 3, заданные объекты как бы "вытягивают" на следующем уровне свои соответствующие правила, которые затем "вытягивают" соответствующие полученные объекты.

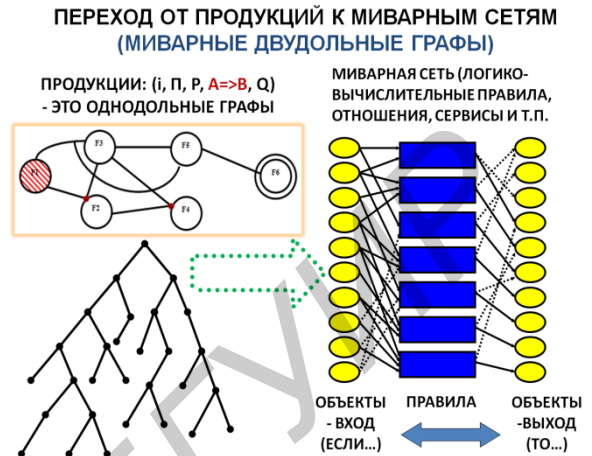


Рисунок 1 – Переход от однодольных графов продукций ("Правила") к двудольным графам миварных сетей ("Объекты; Правила")

ФОРМИРОВАНИЕ ДВУХМЕРНОЙ МАТРИЦЫ МИВАРНОЙ СЕТИ

ОБОЗНАЧЕНИЯ:
M – КОЛИЧЕСТВО ПРАВИЛ В ОПИСАНИИ ЗАДАЧИ;
N – КОЛИЧЕСТВО ВСЕХ ОБЪЕКТОВ - ПЕРЕМЕННЫХ В ПРАВИЛАХ;
 ПРЯМОУГОЛЬНАЯ МАТРИЦА РАЗМЕРОМ (M+1; N+1) – СОДЕРЖИТ ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ, В КОТОРОЙ ПО СТРОКАМ РАСПОЛОЖЕНЫ ВСЕ ПРАВИЛА, А ПО СТОЛБЦАМ – ВСЕ ПЕРЕМЕННЫЕ;
X – ВХОДНЫЕ ПЕРЕМЕННЫЕ В КОНКРЕТНОМ ПРАВИЛЕ;
Y – ВЫХОДНЫЕ ПЕРЕМЕННЫЕ В КОНКРЕТНОМ ПРАВИЛЕ;
Z – ПРИЗНАК ИЗВЕСТНОСТИ ПЕРЕМЕННОЙ, КОТОРЫЙ ПРОСТАВЛЯЕТСЯ В (M+1)-ОЙ СТРОКЕ. ЭТО ЛИБО ЗАДАННЫЕ ВХОДНЫЕ ПЕРЕМЕННЫЕ («ДАНО»), ЛИБО ПОЛУЧЕННЫЕ В ПРОЦЕССЕ РЕШЕНИЯ ЗАДАЧИ;
W – ИСКАОМАЯ ПЕРЕМЕННАЯ, ЗНАЧЕНИЕ КОТОРОЙ НАДО НАЙТИ («НАЙТИ»).

	1	2	3	4	5	...	N-2	N-1	N	N+1
1	X	X	X							
2			X	Y	Y				X	X
...						...				
M		X		X	X			Y		
M+1		Z	Z				W	W		

Рисунок 2 – Формирование матрицы миварной сети

ФОРМИРОВАНИЕ ДВУДОЛЬНОЙ МИВАРНОЙ СЕТИ

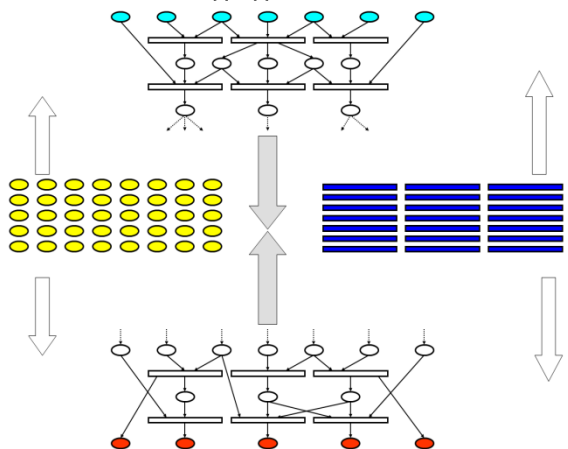


Рисунок 3 – Формирование двудольной миварной сети на основе потока входных данных объектов, типа "Дано" (сверху) и "Найти" (снизу)

Такое "вытягивание" миварами может быть реализовано и "сверху-вниз", и "снизу-вверх", и одновременно от "дано" и от "найти". При этом нет никакого перебора и каждое правило, и каждый объект используется и ищется только один раз. В худшем случае, если есть решение, то будет задействована вся логическая сеть путем такого "вытягивания" с линейной вычислительной сложностью. Если в процессе логического вывода входных данных не будет хватать, то по мере исчерпания всех "запускаемых" правил (т.е. правил у которых известны все входные объекты) процесс остановится и сообщит о нехватке входных данных.

В отличие от традиционных подходов, разделяющих хранение в базах данных, логический вывод и вычислительную обработку, миварный подход позволяет создавать многомерные и эволюционные системы, обрабатывающие информацию в реальном масштабе времени с совмещением логических выводов и вычислительной обработки. Миварный подход - это новый подход для разработки интеллектуальных систем и, в ближайшей перспективе, для создания систем логического искусственного интеллекта [Варламов, 2002], [Варламов, 2003], [Варламов, 2011], [Мивар, 2011], [Санду и др., 2010].

2. Выделение трех уровней исследований в области ИИ

На основе системного анализа выделены 3 основных уровня исследований в области создания искусственного интеллекта.

1. *Рефлексивный ДО-интеллектуальный уровень* - "инстинкты, рефлексы". Нейросети и им подобные методы подобны инстинктивным реакциям, рефлексам и т.п., когда надо мгновенно выполнить некое действие. Образно говоря, если представить систему принятия решения в виде "черного ящика", то на рефлексивном уровне нас интересуют только "входы и выходы" такого ящика - главное, чтобы все эти реакции были правильными и быстрыми. Получается, что мы не пытаемся понять устройство "черного ящика", а только моделируем его работу с целью правильного определения внешних воздействий и поведения системы принятия решений. В основе подобных исследований лежат различные методы статистики, которые имеют ограничения по их применению. Отметим, что этот уровень встречается у животных.

2. *Логический интеллектуальный уровень*. Под этим уровнем будем понимать традиционные логические системы, реализованные в экспертных системах и т.д. Важным отличием от рефлексивного уровня является то, что изучается не поведение, а само устройство "черного ящика". Мы пытаемся "залезть внутрь черного ящика" и смоделировать его, т.е. раскрыть секрет черного ящика и понять его логическую структуру. Термин "интеллект" ближе к логическому осознанному мышлению, когда человек понимает что и как он делает. Это гораздо

медленнее, чем рефлексы и реакции, но зато больше универсальности. Здесь - познание, а нейросети - это диагностика и непосредственное управление. Отметим, что миварный подход работает именно на логическом уровне исследований.

3. *Неформализуемый НАД-интеллектуальный уровень*. В последнее время уделяется внимание вопросам сознания, совести и т.п. свойствам интеллектуальных систем. Эти проблемы явно выходят за пределы "логического осознанного мышления", т.к. даже человек эти процессы не осознает и не всегда может формализовать. Сюда можно отнести и разнообразные эмоции, чувства, мораль и подобные вопросы. Есть позиция: "давайте сначала сделаем хотя бы "логический интеллект", а уж потом займемся вопросами его совести и эмоций". Но, если есть ученые, которым важны и интересны проблемы, которые возникнут сразу после создания ИИ, то этим тоже нужно заниматься. На данном уровне исследований в настоящее время исследуется много неформализуемых задач. Причем, по некоторым предпосылкам, вполне может оказаться, что часть задач принципиально нельзя формализовать или придется работать с бесконечными множествами описания различных предметных областей. В настоящий момент этот уровень изучен меньше других, он является наиболее абстрактным. Учитывая успехи на других уровнях исследований ИИ, вопросы взаимодействия, защиты человечества от роботов и ИИ надо решать заранее, ибо делать это потом будет или уже поздно, или некому. Актуальность и важность этому уровню добавляет и то, что миварный подход уже позволил создать работающие программные комплексы, которые автоматически создают алгоритмы [Варламов и др., 2010], [Варламов, 2002], [Варламов, 2011], [Мивар, 2011], [Санду и др., 2010] и являются прототипом обучаемого логического искусственного интеллекта.

3. Практические реализации миваров

В настоящее время разработано уже несколько программных комплексов на основе миварного подхода [Варламов, 2011], [Мивар, 2011], [Санду и др., 2010]. Прежде всего, развивается проект создания мультипредметных экспертных систем на основе программного комплекса УДАВ с переходом к глобальной миварной активной Интернет-энциклопедии. УДАВ (Универсальный делатель алгоритмов Варламова) предназначен для решения сложных логико-вычислительных задач в реальном времени путем "конструирования алгоритмов" или "поиска маршрута логического вывода" с линейной вычислительной сложностью. Это не простой поиск информации или выполнение жестких алгоритмов, а именно интеллектуальное решение задач с построением алгоритмов, активными запросами недостающей информации и поиском решений.

Результаты практических исследований и экспериментов. В настоящее время, наибольшую трудность представляет описание предметной

области в виде миварной сети. Поэтому для проведения вычислительных экспериментов был создан специальный программный комплекс, основанный на программе УДАВ. В тестовом комплексе реализован генератор миварных матриц описания предметных областей. Например, в процессе экспериментов были созданы миварные матрицы размерности 1 170 007 объектов на 3 510 015 правил. Программе задается количество объектов - переменных. Далее генератор матриц сам генерирует простые арифметические правила для введенного количества объектов. Например, следующие: $a+b=c$ и/или $c-b=a$ и/или $c-a=b$.

Генератор матриц выполняет формирование матрицы и сохраняет ее в специальный файл. Затем тестовая программа считывает тестовую матрицу и начинает обработку. Для однозначности и адекватности проводимых сравнений все генерируемые матрицы в определенном смысле являются подобными. Пользователь может сам сгенерировать матрицы или изменить созданные. Такие эксперименты тоже были проведены. В конце своей работы тестовая программа строит граф решения: наверху рисуются исходные данные, а в внизу объект, который искали. Генератор настроен таким образом, чтобы решение всегда существовало, а его нахождение требовало пропорционального увеличения числа шагов искомого алгоритма решения задачи. Время работы программы определяется по внутренним часам компьютера и может сильно зависеть от общей нагрузки и параллельно выполняемых задач. Время решения задач выводится в миллисекундах (ms). Было рекомендовано запускать тестовую программу с определенным шагом увеличения объектов и в относительно равнозначных условиях внешней загрузки компьютера. Выявлено, что основным ограничением является ресурс оперативной памяти компьютера и корректность работы с ней операционной системы.

Для проведения тестов были задействованы различные компьютеры, начиная от простейших нетбуков с оперативной памятью 512 Мб и до небольших серверов с процессорами Intel 3,8 ГГц и оперативной памятью 4 Гб. Количество ядер не влияло на скорость работы, т.к. основные тесты проводились с использованием однопоточной программы. Использовались два семейства операционных систем: Windows и Mac OS. На семействе ОС Windows получены следующие результаты: до 150 000 объектов и 450 000 правил. Время решения составляло от нескольких миллисекунд до 47 минут.

Наибольший интерес представляют эксперименты, проведенные на ноутбуке MacBook, операционная система Mac OS X версии 10.6.7, процессор Intel Core 2DUO с частотой 2 ГГц и с оперативной памятью DDR3 объемом 4 Гб. На этом ноутбуке в различные периоды времени были проведены тестовые испытания двух версий тестовой программы: B1 и B2. Результаты

испытаний приведены на рисунке 4. Итак, на обычном серийном ноутбуке MacBook неоптимизированная тестовая программа УДАВ, написанная на языке ЯВА (не самом быстром языке), обрабатывает миварные матрицы размерности 1 миллион объектов на 3,5 миллиона правил примерно за 3 часа.

Испытания на максимальное значение получили следующий результат: 1 170 007 объектов - переменных, 3 510 015 правил и время решения 12 239 183 миллисекунды, что примерно равно 200 минутам. Для сравнения приведем следующие оценки: система управления атомными станциями оперирует всего 20 000 правил; описание всей программы средней школы и технического ВУЗа потребует около 300 правил на каждую научную область и в сумме не превысит 100 000 правил. Напомним, что продукции и предикатные системы не могли работать и с сотней правил.

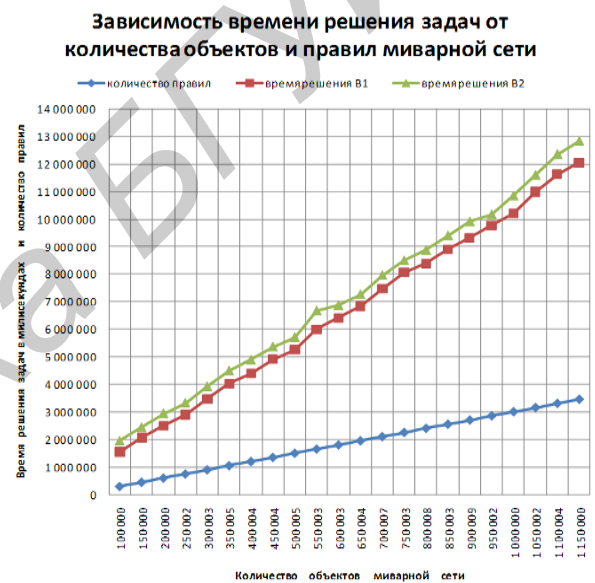


Рисунок 4 – Результаты экспериментов на MacBook

Коботаевым Н.С. была создана многопоточная версия УДАВ на языке C++ и с MPI для систем с распределенной памятью. На рисунке 5 приведены результаты при суммарном количестве процессов $N=8$, $NS=3$, $NC=4$, $C=3$. Время выполнения задач прямо пропорционально количеству правил.

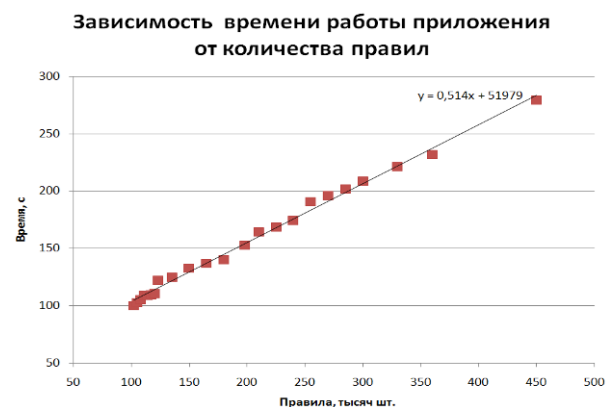


Рисунок 5 – Результаты экспериментов на MPI

Эксперименты проведены на 15 компьютерах. Для каждого значения количества объектов проводилось более 10 однотипных испытания в примерно одинаковых условиях. Расчеты показали, что ошибка измерения времени работы программы не превышала 3 %. Все проведенные эксперименты подтвердили линейную вычислительную сложность логического вывода и/или автоматического конструирования алгоритмов на основе миваров.

Миварный подход не отвергает существующие научные достижения и различные модели, а позволяет объединить их в единую систему для достижения принципиально новых возможностей по обработке и накоплению информации.

Современные "технические средства" в виде ГРИД и облачных вычислений позволяют проводить параллельную обработку и решать требуемые задачи с одновременным использованием различных моделей и программных комплексов. Миварный подход позволяет параллельно решать эти задачи, объединять полученные результаты и обеспечивает реальное время логической обработки и автоматического конструирования алгоритмов в сложных случаях.

4. Этапы обработки информации

Выделяют три основных этапа миварной обработки информации:

- 1) формирование миварной матрицы описания предметной области;
- 2) работа с матрицей и конструирование алгоритма решения заданной задачи;
- 3) по полученному алгоритму выполнение всех вычислений и нахождение ответа.

Первый этап - формирование матрицы является по существу этапом синтеза концептуальной модели предметной области и ее формализации в виде продукционных правил с переходом к миварным сетям формата:

"входные объекты => правила+процедуры => выходных объекты".

В настоящее время именно этот этап является наиболее сложным и требует участия человека-специалиста для создания миварной модели предметной области. С точки зрения обучения важно, что на основе второго этапа УДАВ выполняет "объяснения" для обучаемого, показывает и обосновывает ход решения. Преимущества миварного подхода:

- 1) скорость: линейная вычислительная сложность и реальное время;
- 2) универсальность: решение логических и вычислительных задач;
- 3) эволюция: адаптивное описание и непрерывное решение задач;

4) управление потоком входных данных и оперативная диагностика;

5) активная работа с запросами или уточнениями входных данных на эволюционной сети правил и объектов (обучение).

Рассмотрим этапы миварной обработки на практическом примере. Решение задач треугольников в области геометрии является хорошим тестовым примером для экспертных систем с логическим выводом. Считается, что логический вывод это полно-переборная задача с факториальной вычислительной сложностью, которая зависит от количества правил.

Область "Геометрия. Треугольники" хорошо изучена и описана в учебниках. В нашем случае мы взяли 39 объектов-переменных (углы, стороны, высоты и т.д.), которые описываются 177 правилами - зависимостями одних объектов от других. Решение задач состоит в том, что по заданному набору входных параметров ("дано") надо найти выходные параметры ("найти"). Ученику известны все уравнения и зависимости в этой области. Тогда под каждую "задачу" надо составить алгоритм ее решения, затем вычислить и получить из "дано" требуемые "найти". Известно 2 метода решения:

- 1) логическим выводом строить алгоритмы и
- 2) сформировать матрицу всех алгоритмов для любых наборов "дано" и "найти".

В первом случае вычислительная сложность определяется количеством правил, т.е. $177!$ (факториал). Для второго случая под каждый набор входных и выходных данных требуется до $39!$ строк матрицы. Оба варианта не осуществимы на практике.

Есть третий вариант: миварные сети, которые управляются потоком входных данных и могут автоматически конструировать алгоритмы с линейной вычислительной сложностью логического вывода. Носов А.В. реализовал миварный метод в программе "УДАВ", которая решает все задачи для 39 объектов и 177 правил в реальном времени.

На основе опроса преподавателей средней школы был выявлен наиболее удобный интерфейс проекта, показанный на рисунках 6-10 сначала задают "дано" и "найти", затем программа формирует (автоматически конструирует) алгоритм решения, вычисляет переменные и рисует алгоритм решения задачи. На рисунке 6 показано задание исходных данных цифрами в столбце "значение" и выделение "галочками" требуемых значений в столбце "найти".

В нашем примере заданы 3 высоты - строки P15, P16 и P17, а найти надо параметры в строках: P1 (угол A), P4 (сторона A), P24 (площадь треугольника), P25 (расстояние от точки пересечения медианой стороны а до точки пересечения медиан Ma0) и P39 (расстояние от точки пересечения медиан до стороны C).

Таблица исходных данных:			
id	описание переменных	значение	найти
P1	Угол A, противолежащий стороне a (град)	0	<input checked="" type="checkbox"/>
P2	Угол B, противолежащий стороне b (град)	0	<input type="checkbox"/>
P3	Угол C, противолежащий стороне c (град)	0	<input type="checkbox"/>
P4	Сторона a	0	<input checked="" type="checkbox"/>
P5	Сторона b	0	<input type="checkbox"/>
P6	Сторона c	0	<input type="checkbox"/>
P7	Радиус описанной окружности R	0	<input type="checkbox"/>
P8	Радиус вписанной окружности r	0	<input type="checkbox"/>
P9	Расстояние между центрами вписанной и описанной окружностями d	0	<input type="checkbox"/>
P10	Периметр треугольника P	0	<input type="checkbox"/>
P11	Полупериметр треугольника p	0	<input type="checkbox"/>
P12	Средняя линия Sa, параллельная стороне a	0	<input type="checkbox"/>
P13	Средняя линия Sb, параллельная стороне b	0	<input type="checkbox"/>
P14	Средняя линия Sc, параллельная стороне c	0	<input type="checkbox"/>
P15	Высота Ha, опущенная из угла A к стороне a	157	<input type="checkbox"/>
P16	Высота Hb, опущенная из угла B к стороне b	179	<input type="checkbox"/>
P17	Высота Hc, опущенная из угла C к стороне c	195	<input type="checkbox"/>
P18	Медиана Ma, опущенная из угла A к стороне a	0	<input type="checkbox"/>
P19	Медиана Mb, опущенная из угла B к стороне b	0	<input type="checkbox"/>
P20	Медиана Mc, опущенная из угла C к стороне c	0	<input type="checkbox"/>
P21	Биссектриса La угла A	0	<input type="checkbox"/>
P22	Биссектриса Lb угла B	0	<input type="checkbox"/>
P23	Биссектриса Lc угла C	0	<input type="checkbox"/>
P24	Площадь треугольника	0	<input checked="" type="checkbox"/>
P25	Расстояние от точки пересечения медианой стороны a до точки пересечения медиан MaO	0	<input checked="" type="checkbox"/>
P26	Расстояние от точки пересечения медианой стороны b до точки пересечения медиан MbO	0	<input type="checkbox"/>
P27	Расстояние от точки пересечения медианой стороны c до точки пересечения медиан McO	0	<input type="checkbox"/>
P28	Отрезок, на который биссектриса La делит сторону a - aC	0	<input type="checkbox"/>
P29	Отрезок, на который биссектриса La делит сторону a - aB	0	<input type="checkbox"/>
P30	Отрезок, на который биссектриса Lb делит сторону b - bA	0	<input type="checkbox"/>
P31	Отрезок, на который биссектриса Lb делит сторону b - bC	0	<input type="checkbox"/>
P32	Отрезок, на который биссектриса Lc делит сторону c - cA	0	<input type="checkbox"/>
P33	Отрезок, на который биссектриса Lc делит сторону c - cB	0	<input type="checkbox"/>
P34	Расстояние от вершины A до точки пересечения медиан	0	<input type="checkbox"/>
P35	Расстояние от вершины B до точки пересечения медиан	0	<input type="checkbox"/>
P36	Расстояние от вершины C до точки пересечения медиан	0	<input type="checkbox"/>
P37	Расстояние от точки пересечения медиан до стороны a	0	<input type="checkbox"/>
P38	Расстояние от точки пересечения медиан до стороны b	0	<input type="checkbox"/>
P39	Расстояние от точки пересечения медиан до стороны c	0	<input checked="" type="checkbox"/>

Рисунок 6 – Задание исходных данных цифрами в столбце "значение" и выделение "галочками" требуемых значений в столбце "найти"

На рисунке 7 показаны результаты работы программы по решению заданной задачи: все полученные значения показаны в столбце "значение". Далее, на рисунках 8, 9 и 10 показаны результаты работы программы по автоматическому конструированию алгоритма решения заданной задачи: в окне "Графический вывод работы логики" показан алгоритм решения задачи, который разбит на 3 фрагмента.

На рисунке 8 показаны результаты работы программы по решению заданной задачи: в окне "Графический вывод работы логики" показан алгоритм решения задачи - фрагмент номер 1 (начало алгоритма).

На рисунке 9 показаны результаты работы программы по решению заданной задачи: в окне "Графический вывод работы логики" показан алгоритм решения задачи - фрагмент номер 2 (середина алгоритма).

Таблица исходных данных:			
id	описание переменных	значение	найти
P1	Угол A, противолежащий стороне a (град)	72,805	<input type="checkbox"/>
P2	Угол B, противолежащий стороне b (град)	0	<input type="checkbox"/>
P3	Угол C, противолежащий стороне c (град)	0	<input type="checkbox"/>
P4	Сторона a	232,727	<input type="checkbox"/>
P5	Сторона b	204,124	<input type="checkbox"/>
P6	Сторона c	187,376	<input type="checkbox"/>
P7	Радиус описанной окружности R	0	<input type="checkbox"/>
P8	Радиус вписанной окружности r	0	<input type="checkbox"/>
P9	Расстояние между центрами вписанной и описанной окружностями d	0	<input type="checkbox"/>
P10	Периметр треугольника P	0	<input type="checkbox"/>
P11	Полупериметр треугольника p	0	<input type="checkbox"/>
P12	Средняя линия Sa, параллельная стороне a	0	<input type="checkbox"/>
P13	Средняя линия Sb, параллельная стороне b	0	<input type="checkbox"/>
P14	Средняя линия Sc, параллельная стороне c	0	<input type="checkbox"/>
P15	Высота Ha, опущенная из угла A к стороне a	157	<input type="checkbox"/>
P16	Высота Hb, опущенная из угла B к стороне b	179	<input type="checkbox"/>
P17	Высота Hc, опущенная из угла C к стороне c	195	<input type="checkbox"/>
P18	Медиана Ma, опущенная из угла A к стороне a	157,632	<input type="checkbox"/>
P19	Медиана Mb, опущенная из угла B к стороне b	0	<input type="checkbox"/>
P20	Медиана Mc, опущенная из угла C к стороне c	197,831	<input type="checkbox"/>
P21	Биссектриса La угла A	0	<input type="checkbox"/>
P22	Биссектриса Lb угла B	0	<input type="checkbox"/>
P23	Биссектриса Lc угла C	0	<input type="checkbox"/>
P24	Площадь треугольника	18 269,07	<input type="checkbox"/>
P25	Расстояние от точки пересечения медианой стороны a до точки пересечения медиан MaO	52,545	<input type="checkbox"/>
P26	Расстояние от точки пересечения медианой стороны b до точки пересечения медиан MbO	0	<input type="checkbox"/>
P27	Расстояние от точки пересечения медианой стороны c до точки пересечения медиан McO	65,944	<input type="checkbox"/>
P28	Отрезок, на который биссектриса La делит сторону a - aC	0	<input type="checkbox"/>
P29	Отрезок, на который биссектриса La делит сторону a - aB	0	<input type="checkbox"/>
P30	Отрезок, на который биссектриса Lb делит сторону b - bA	0	<input type="checkbox"/>
P31	Отрезок, на который биссектриса Lb делит сторону b - bC	0	<input type="checkbox"/>
P32	Отрезок, на который биссектриса Lc делит сторону c - cA	0	<input type="checkbox"/>
P33	Отрезок, на который биссектриса Lc делит сторону c - cB	0	<input type="checkbox"/>
P34	Расстояние от вершины A до точки пересечения медиан	0	<input type="checkbox"/>
P35	Расстояние от вершины B до точки пересечения медиан	0	<input type="checkbox"/>
P36	Расстояние от вершины C до точки пересечения медиан	131,898	<input type="checkbox"/>
P37	Расстояние от точки пересечения медиан до стороны a	0	<input type="checkbox"/>
P38	Расстояние от точки пересечения медиан до стороны b	0	<input type="checkbox"/>
P39	Расстояние от точки пересечения медиан до стороны c	92,828	<input type="checkbox"/>

Рисунок 7 – Результаты работы программы по решению заданной задачи: все полученные значения показаны в столбце "значение"

На рисунке 10 показаны результаты работы программы по решению заданной задачи: в окне "Графический вывод работы логики" показан алгоритм решения задачи - фрагмент номер 3 (конец алгоритма).

Практическая польза: преподаватель может готовить индивидуальные задания и проверять контрольные задания, а ученики сами получают объяснения в программе.

Таким образом, Миварная экспертная система "УДАВ. Геометрия" в реальном времени решает все задачи для 39 объектов и 177 правил, что помогает и учителям, и ученикам.

В настоящее время ведутся работы по добавлению новых объектов и правил из различных математических разделов. Кроме того, будут добавлены объекты и правила, описывающие физические процессы.

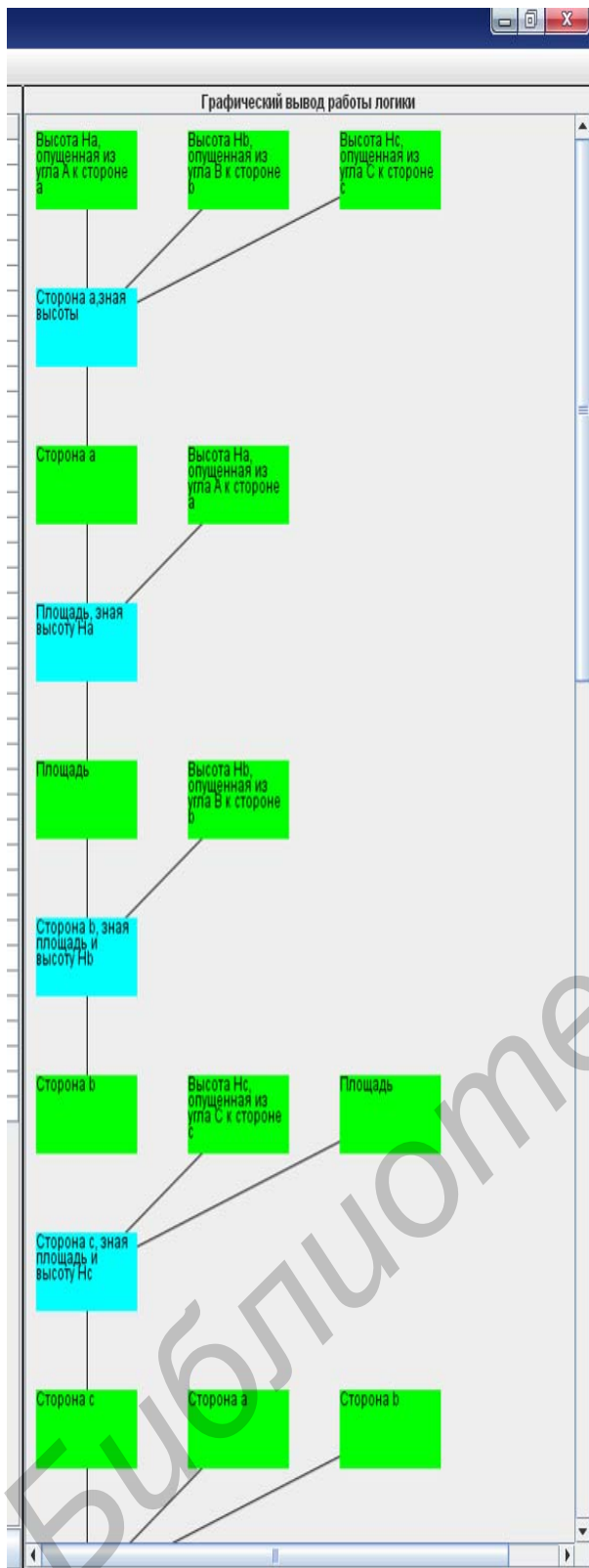


Рисунок 8 – Результаты работы (начало алгоритма)

На основе миварных технологий реализуются инновационные проекты [Варламов, 2011], [Мивар, 2011]. На Конкурсе русских инноваций наши проекты принимают участие в номинациях "Белая книга", "Перспективный проект" и "Инновационный проект".

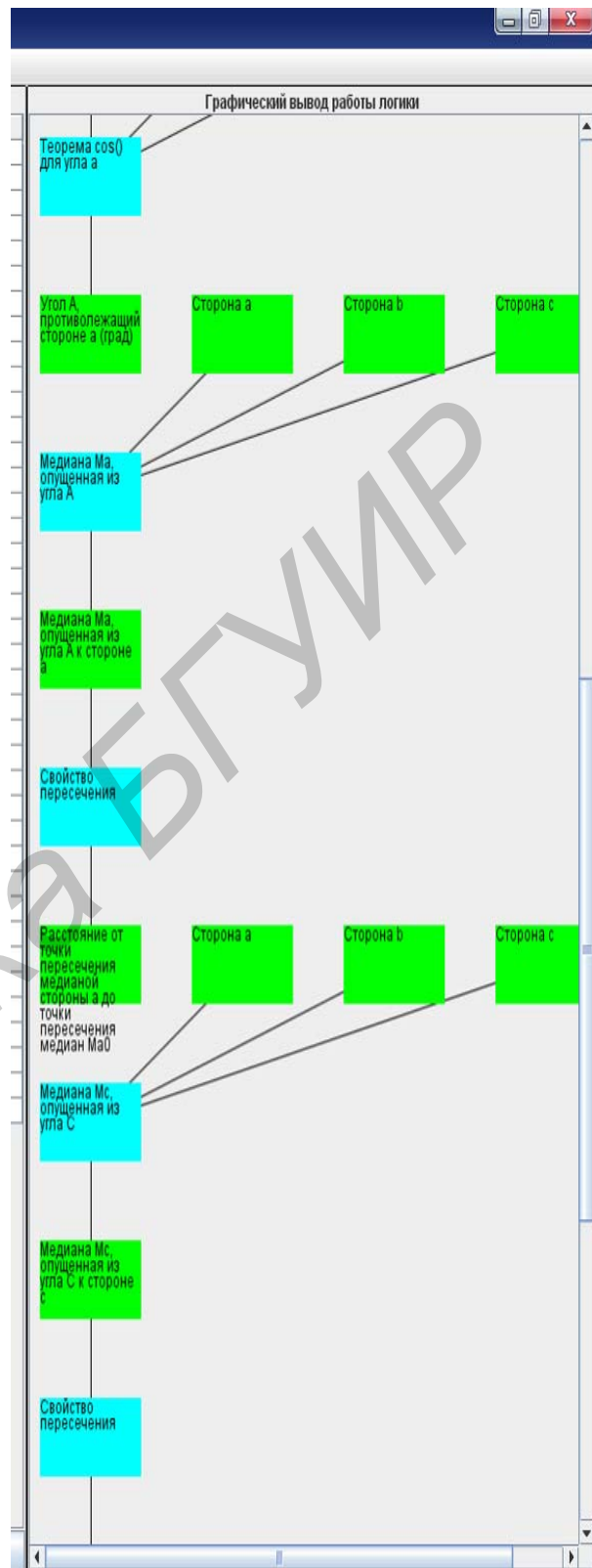


Рисунок 9 – Результаты работы (середина алгоритма)

Наши проекты в области создания искусственного интеллекта, разработки миварной информационной инфраструктуры российской электроэнергетики и другие доходили до Второго тура и становились финалистами Конкурса русских инноваций в разные годы.

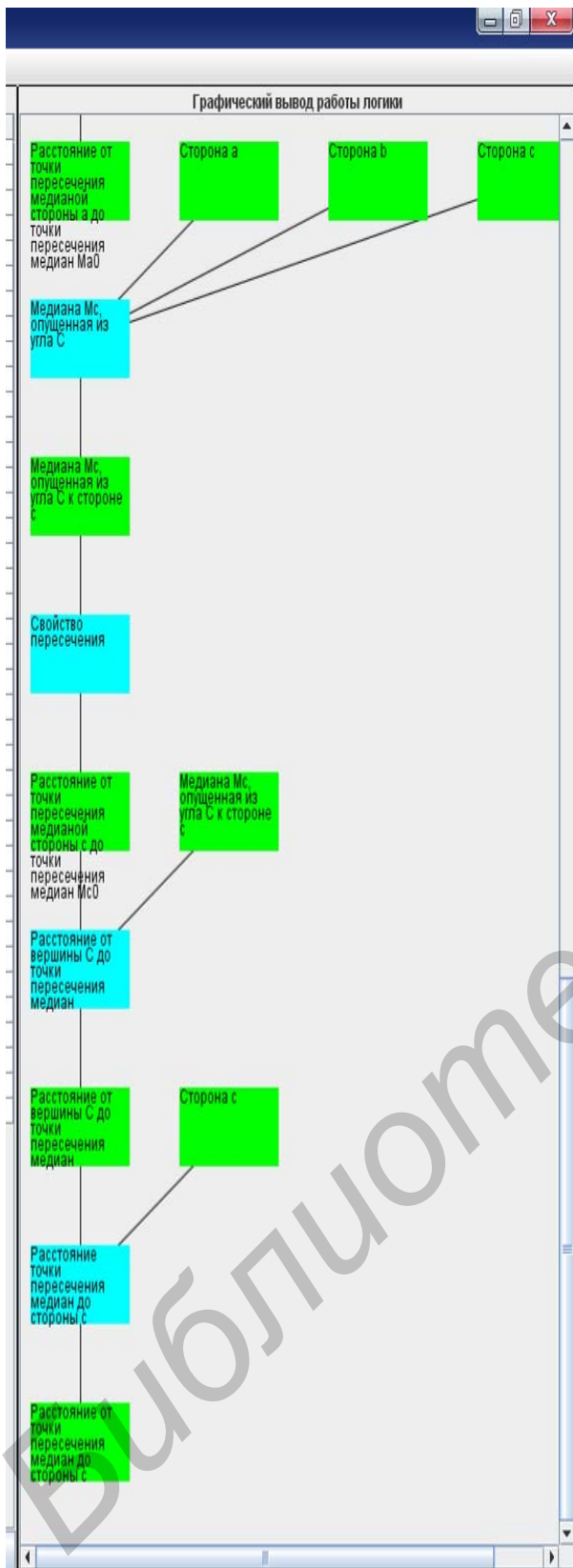


Рисунок 10 – Результаты работы (конец алгоритма)

В перспективе получится глобальная Миварная активная интернет-энциклопедия, в которой будут собраны объекты и правила для всех известных предметных областей и мы получим глобальную мультипредметную экспертную систему.

5. Моделирование контекста, понимание смысла и создание логического искусственного интеллекта на основе миварных технологий

Итак, обосновано, что миварные технологии позволяют эволюционно накапливать любые объемы данных и логически обрабатывать их с линейной вычислительной сложностью. На основе миваров создан программный комплекс УДАВ, который обрабатывает более 1,17 млн переменных и более 3,5 млн правил на обычных компьютерах и ноутбуках. Следовательно, миварный подход позволяет на новом уровне перейти к работе с большими контекстами и информационными моделями предметных областей.

В большинстве современных задач в различных областях, включая распознавание образов, поиск информации, перевод и аннотирование текстов, а также в других проблемах области искусственного интеллекта ключевую роль играет необходимость работы с огромными и структурно сложными контекстами.

По существу, контекст - это информационная модель предметной области, в которой есть данные и правила их обработки. В определенном смысле, адекватный контекст и есть "смысл информации". Одновременность накопления фактов и обработки миллионов правил в миварных системах позволяют перейти к созданию сложных и многоуровневых контекстов и адекватных информационных моделей предметных областей, что проиллюстрировано на рисунке 11.

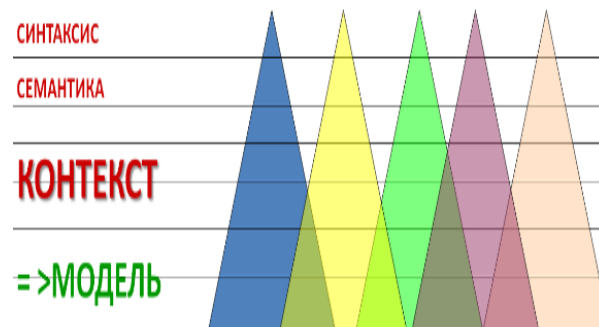


Рисунок 11 – Иллюстрация необходимости постепенного наращивания по уровням и переход от синтаксиса и семантики к обработке контекста

Такой переход можно осуществлять путем создания миварных информационных циклов и перехода от обработки синтаксиса и семантики к обработке контекста и информационных моделей предметных областей, что проиллюстрировано на рисунке 12. Сложные контексты, в свою очередь, позволят на новом уровне решать все интеллектуальные задачи на основе работы "компьютеров со смыслом информации". Как только компьютеры начнут работать со смыслом, можно говорить о создании полноценного логического искусственного интеллекта.

МИВАРНЫЕ ИНФОРМАЦИОННЫЕ ЦИКЛЫ => КОНТЕКСТ = ИНФОРМАЦИОННАЯ МОДЕЛЬ

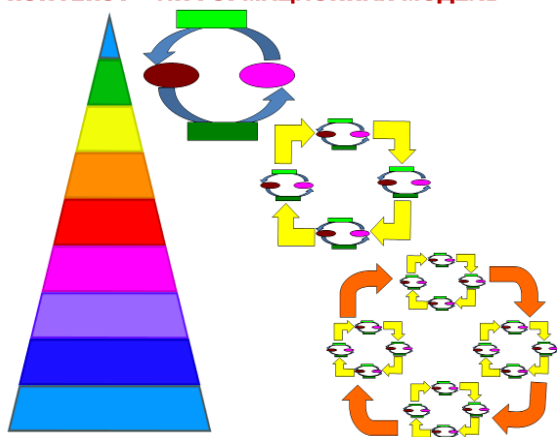


Рисунок 12 – Иллюстрация создания миварных информационных циклов и перехода от обработки синтаксиса и семантики к обработке контекста и информационных моделей предметных областей

В настоящий момент ограничения на возможности миваров по созданию сложных контекстов не обнаружены, т.к. возможности существующих компьютерных систем, объединенных сетями и ГРИД-технологиями, практически безграничны.

Таким образом, пролучаем следующую логическую цепочку: Мивары => БД + Логика => Контекст => Смысл => Искусственный интеллект!

Следовательно, мивары позволяют создавать новое поколение информационных систем, накапливающих огромные объемы данных и обрабатывающих неограниченное количество логических правил. Такие системы позволяют реализовать работу со сложным и неограниченным контекстом. А это, в свою очередь, позволит компьютерам постепенно, по мере их обучения, т.е. наращивания контекста и усложнения "миварными циклами" информационных моделей, работать со смыслом информации и породит полноценный логический искусственный интеллект!

ЗАКЛЮЧЕНИЕ

Целесообразно выделить 3 уровня научных исследований в области искусственного интеллекта. Миварный подход на логическом уровне позволяет создавать активные глобальные обучаемые логически рассуждающие автоматические познающе-диагностические системы реального времени. Для экспертных систем нового поколения вместо продукций и однодольных графов целесообразно использовать двудольные графы миварных сетей.

На основе миварных сетей создан программный комплекс УДАВ, который обрабатывает более 1,17 млн объектов и более 3,5 млн правил на обычных компьютерах и ноутбуках. Приведены результаты практических расчетов и решений различных

прикладных задач, которые на практике подтверждают линейную вычислительную сложность логического вывода и/или автоматического конструирования алгоритмов в формализме двудольных миварных сетей.

Впервые предоставлены возможности создания мультимедийных экспертных систем, способных работать с миллионами объектов и продукционных правил в формализме двудольных миварных сетей.

В работе обоснована следующая логическая цепочка: Мивары => БД + Логика => Контекст => Смысл => Искусственный интеллект!

Мивары позволяют создать новое поколение интеллектуальных систем, накапливающих огромные объемы данных и обрабатывающих очень большое (практически неограниченное) количество логических правил, сервисов и вычислительных процедур.

Миварные системы позволяют работать компьютерам с огромным контекстом в реальное время. Следовательно, компьютеры начинают работать со смыслом информации, что и порождает полноценный искусственный интеллект на логическом уровне научных исследований.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- [Варламов, 2002] Варламов О.О. Эволюционные базы данных и знаний для адаптивного синтеза интеллектуальных систем. Миварное информационное пространство. - М.: Радио и связь, 2002. - 288 с.
- [Варламов, 2003] Варламов О.О. Основы многомерного информационного развивающегося (миварного) пространства представления данных и правил // Информационные технологии. 2003. № 5. С.42-47.
- [Варламов и др., 2010] Варламов О.О., Санду Р.А., Владимиров А.Н., Носов А.В., Оверчук М.Л. Миварный подход к созданию мультимедийных активных экспертных систем в целях обучения информационной безопасности и управления инновационными ресурсами в образовании // Известия Южного федерального университета. Технические науки. 2010. № 11. С. 226 - 232.
- [Варламов, 2011] Веб-сайт // <http://www.ovar.narod.ru>. 2011.
- [Владимиров и др., 2010] Владимиров А.Н., Варламов О.О., Носов А.В., Потапова Т.С. Программный комплекс "УДАВ": практическая реализация активного обучаемого логического ввода с линейной вычислительной сложностью на основе миварной сети правил // Труды научно-исследовательского института радио. - 2010. - № 1. С. 108-116.
- [Джарратано и др., 2007] Джарратано Дж., Райли Г. Экспертные системы: принципы разработки и программирование, 4е издание. - М.: Вильямс, 2007. 1152 с.
- [Коголовский, 2005] Коголовский М.Р. Энциклопедия технологий баз данных. - М.: Финансы и статистика, 2005. - 800 с.
- [Кузнецов, 2009] Кузнецов О.П. Дискретная математика для инженера. 6-е изд., стер. - СПб: Издательство "Лань", 2009. - 400 с.
- [Люгер, 2005] Люгер Дж. Ф. Искусственный интеллект: стратегии и методы решения сложных проблем, 4-е издание.: Пер. с англ. - М.: Издательский дом "Вильямс", 2005. - 864 с.
- [Мивар, 2011] Веб-сайт компании МИВАР // <http://www.mivar.ru>. 2011.
- [Поспелов, 1989] Поспелов Д.А. Моделирование рассуждений. Опыт анализа мыслительных актов. - М.: Радио и связь, 1989. - 184 с.
- [Санду и др., 2010] Санду Р.А., Варламов О.О. Миварный подход к созданию интеллектуальных систем и искусственного интеллекта. Результаты 25 лет развития и ближайшие перспективы. - М.: Стандартинформ, 2010. - 339 с.

**THE MIVAR TECHNOLOGIES OF LOGICAL
ARTIFICIAL INTELLIGENCE CREATING:
LOGICAL INFERENCE WITH LINEAR
COMPUTATIONAL COMPLEXITY WITH
MORE THAN 3 MILLIONS PRODUCTION
RULES AND POSSIBILITY TO UNDERSTAND A
MEANING THROUGH MODELING OF A BIG
CONTEXT**

Varlamov O.O.

*Professor of Department of Applied Mathematics
at Moscow State Automobile and Road Technical
University (MADI), Professor of Department of
Radio and Information Technology FRTK Moscow
Institute of Physics and Technology (State
University), Moscow, Russia,*

Ovar@narod.ru, OVarlamov@gmail.ru

It is reasonable to define 3 levels of scientific researches in the sphere of Artificial Intelligence. The MIVAR method on the logical level allows to create automatic systems which have the following qualities: activeness, globality, learning capability, ability to make logical inferences and give conclusions in real time. It is reasonable for new generation of expert systems to use bipartite graphs of the MIVAR nets instead of production rules and monocots graphs.

Keywords: MIVAR, MIVAR net, logical inference, computational complexity, artificial intelligence, intelligent systems, expert systems, General Problem Solver.

The theoretical transition from the graphs of production systems to the bipartite graphs of the MIVAR nets is shown. Examples of the implementation of the MIVAR nets in the formalisms of matrixes and graphs are given. The linear computational complexity of algorithms for automated building of objects and rules of the MIVAR nets is theoretically proved. On the basis of the MIVAR nets the UDAV software complex is developed, handling more than 1.17 million objects and more than 3.5 million rules on ordinary computers. The results of experiments that confirm a linear computational complexity of the MIVAR method of logical inference. The MIVAR approach unifies and develops achievements from different scientific domains: databases, computational problems, logic processing, and includes two main technologies:

1. The MIVAR technology of information accumulation – is a method of creating of global evolutionary bases of data and rules (knowledge) with changeable structure based on the adaptive discrete MIVAR information space of unified representation of data and rules which bases on three main definitions: “Thing, Property, Relation”.
2. The MIVAR technology of information processing - is a method of creation of the system of logic inference or “automatic construction of algorithms from modules, services and procedures” based on the active MIVAR net of rules with linear computational complexity.

The MIVAR technology of information accumulation is designed for keeping any information with possible evolutionary change of its structure and without any restrictions of its volume and the form of representation. The MIVAR technology of information processing is designed for the processing of information, including logic inference, computational procedures and services. In fact, MIVAR nets allow to develop production approach and to create an automatic learning logically thinking system. The MIVAR approach unifies and develops production systems, ontology, semantic nets, service-oriented architectures, multi-agent systems and other modern information technologies.

There are three basic stages of MIVAR data processing:

- 1) Creation of the MIVAR matrix for the description of subject domain;
- 2) Working with the matrix and the construction (designer) of the algorithm for solving of the required problem;
- 3) Execution of all computations basing on the acquired algorithm.

The first stage can be seen as the formalization of subject domain in the form of productions with the following transition to the MIVAR rules:

"input objects – rules/procedures/services – output objects".

Currently, it is the most difficult stage that requires the participation of a human-specialist (expert) for creation of the MIVAR model of subject domain. On the second stage, automated construction of the algorithm and logic inference is being implemented. The input data is represented in the form of the MIVAR matrix of the description of subject domain and specified input (“GIVEN”) and required (“TO FIND”) objects-variables. On the third stage, the process of solution basing on the obtained algorithm is being executed. For the moment, in the software complex UDAV the work of the second and the third stages is combined. Currently, there are more than 7 different realizations of the MIVAR method. In some of them, three main stages are processed separately, but in this article the work of the UDAV software complex is described where all three stages can be combined.

Our paper is important for other researchers in the sphere of AI, because we present a new formalism of knowledge representation through which logical processing of data is increased significantly. Observing some restrictions, our formalism (MIVAR nets) provides linear computational complexity of algorithms or automated building (logical inference) on production systems base. The theoretical transition from AND/OR graphs of production systems to the bipartite graphs of the MIVAR (Multidimensional Informational Variable Adaptive Reality) nets is shown. Our results can be used for creation of intelligent systems, expert systems, General Problem Solver in different spheres.