

УДК 004.8

"ОБЛАЧНАЯ" РЕАЛИЗАЦИЯ МИВАРНОГО УНИВЕРСАЛЬНОГО РЕШАТЕЛЯ ЗАДАЧ НА ОСНОВЕ АДАПТИВНОГО АКТИВНОГО ЛОГИЧЕСКОГО ВЫВОДА С ЛИНЕЙНОЙ СЛОЖНОСТЬЮ ОТНОСИТЕЛЬНО ПРАВИЛ "ЕСЛИ-ТО-ИНАЧЕ"

Чибирова М.О., Сергушин Г.С., Елисеев Д.В., Варламов О.О.

Аннотация

Целью наших исследований является создание логического искусственного интеллекта и его применение в различных предметных областях. Предложено развитие миварной теории путем перехода от формализма "правила-переменные" к формализму "отношения-правила-переменные" в миварных сетях и введением многоуровневого и многомерного представления правил. Представлены результаты создания новой версии миварного универсального решателя задач УДАВ, способного выполнять адаптивный активный логический вывод на двухуровневом представлении "отношения-правила-переменные". Теперь УДАВ реализован на основе облачных технологий и размещен на сайте www.mivar.org. Экспериментально подтверждена теоретическая линейная вычислительная сложность логического вывода и автоматического конструирования алгоритмов решения различных задач на основе миварных сетей и продукций вида "если - то-иначе". В перспективе УДАВ практически полностью может заменить человека-оператора в автоматизированных системах обработки информации и АСУТП.

***Ключевые слова:** мивар; миварные сети; искусственный интеллект; универсальный решатель задач; логический вывод с линейной вычислительной сложностью; экспертные системы; активные базы данных.*

Введение

В областях автоматизированных системах обработки информации (АСОИ), автоматизированных систем управления технологическими процессами (АСУТП) и информационной безопасности (ИБ) достижения информатики, кибернетики и искусственного интеллекта всегда используются в первоочередном порядке. Это отражает необходимость автоматизации и ускорения выполнения сложных функций в автоматизированных системах обработки информации и АСУТП. Отметим, что в процессе проведения автоматизации и развития информатики за последние тридцать лет многие задачи в автоматизированных системах обработки информации и АСУТП хорошо теоретически описаны и с ними справляются операторы систем. Эти операторы действуют по заранее разработанным инструкциям и алгоритмам принятия решения, которые описываются в виде причинно-следственных зависимостей - продукций "если-

то". Конечно, не все задачи в автоматизированных системах обработки информации и АСУТП можно описать в таком виде, но те задачи, которые можно описать продукциями "если-то", требуют логического подхода к своему решению.

Важно подчеркнуть, что начиная с 2002 года в публикациях [1-17, 24-28] доказано, что существует линейной вычислительной сложности логический вывод на правилах "ЕСЛИ-ТО" в формализме миварных сетей (MIVAR net's). Этот метод реализован в виде программного комплекса УДАВ [2, 6]. Подробно миварный метод логического вывода с линейной вычислительной сложностью, относительно количества правил, описан в наших научных статьях, размещенных на общедоступных ресурсах в сети Интернет [3-5]. Таким образом, опровергнут очень устойчивый среди некоторых ученых миф о том, что логический вывод является NP-полной задачей. Кардинальное снижение вычислительной сложности логического вывода открыло новые возможности по созданию сложных гетерогенных АСОИ и АСУТП реального времени.

В работах [7-11] миварные технологии уже применялись для успешного решения некоторых задач в области информационной безопасности, автоматизированных системах обработки информации и АСУТП. В работе [12] описано построение нового миварного информационного пространства данных и правил, которое является основой эволюционного накопления информации.

Особо необходимо отметить предложенные в работах [15-16] новые миварные самоорганизующиеся программно-аппаратные комплексы оперативной диагностики (СПАКОД), которые решают сложные и важные задачи в информационной безопасности, автоматизированных системах обработки информации и АСУТП с выполнением требований своевременности, полноты и достоверности обработки информации. Учитывая новые возможности применения мобильных роботов, в работе [17] были показаны варианты встраивания миварных технологий в робототехнические системы.

Отдельным направлением развития является разработка модели представления знаний при создании адаптивной информационной системы на основе миварного информационного пространства [18-23].

Следовательно, развитие миварных технологий и "он-лайн" реализация универсального решателя задач на основе адаптивного активного логического вывода с линейной вычислительной сложностью относительно правил в виде причинно-следственных связей "если-то" является актуальной и практически значимой задачей для решения многих проблем в автоматизированных системах обработки информации и АСУТП.

Развитие миварных технологий и решение задач треугольников

Сразу отметим, что непосредственно ознакомиться с примерами практической работы "он-лайн" версии программного комплекса УДАВ можно на сайте www.mivar.ru. Сейчас там реализовано, например, решение геометрических задач с треугольниками.

Суть решения состоит в том, чтобы из известных формул составить конкретный алгоритм решения поставленной задачи в формате: "ДАНО - НАЙТИ", затем подставить значения и провести необходимые вычисления. Ранее, подобные задачи относили к классу задач интеллектуальных пакетов прикладных программ (ИППП). Суть проблемы состояла в том, что есть набор прикладных программ (сейчас наиболее адекватное название - "сервисы"), которые могут решать различные задачи. На вход поступали некоторые значения переменных и надо было найти значения других выделенных переменных в формате "Дано. Найти". В некотором роде, подобные задачи называли: "управляемые потоком входных данных", т.к. в зависимости от поступления входных данных, надо было строить различные алгоритмы нахождения требуемых переменных.

Важно подчеркнуть, что решение задач треугольников является модельной задачей для создания и тестирования целого класса программных комплексов (ИППП) в области информационной безопасности, в автоматизированных системах обработки информации и АСУТП при решении задач диагностики, оценки состояния объектов наблюдения и прогнозирования их деятельности в реальном масштабе времени. Как известно, например, подробные описания задач информационной безопасности являются конфиденциальными и их не выдают исполнителям. Характерными чертами подобных задач информационной безопасности является представление предметной области в виде набора объектов, их признаков и взаимосвязей между объектами.

Для расширения возможностей подобного моделирования и было предложено использовать миварную модель данных "вещь-свойство-отношение". Взаимосвязи между объектами предметной области в информационной безопасности описываются причинно-следственными зависимостями формата "если-то, иначе", который может быть сведен в простейшем варианте к представлению в виде продукции "если-то". Подобный класс задач, с точки зрения искусственного интеллекта, относится к экспертным системам, универсальным решателям задачам, автоматизированным системам управления и т.п. Основную сложность составляет создание алгоритма, что по существу является классической задачей логического вывода на продукциях.

Рассмотрим, для примера, решение задач школьной геометрии про треугольники. Известно много зависимостей между сторонами, медианами, высотами, углами и другими характеристиками треугольника: теоремы синусов, косинусов, Пифагора и др.- а также формулы получения периметра треугольника, его площади и т.д. Поэтому известно много разных задач в геометрии: найти площадь по двум сторонам и углу между ними; по высоте и стороне с углом определить другие стороны, построить медианы и т.п.

В нашем случае, показанном на указанном сайте, используется: 72 объекта (углы, стороны, высоты и т.д.) и 237 правил (в любой момент времени можно добавлять объекты и правила).

Решение задач состоит в том, что по заданному набору входных параметров ("дано") надо найти выходные параметры ("найти").

Известно два "традиционных" решения:

- 1) искать логический вывод от "дано" к "найти" (на предикатах);
- 2) заранее построить все алгоритмы для любых наборов "дано - найти".

В первом случае сложность определяется количеством правил, если вспомнить что логический вывод считался NP-полной задачей, то получаем: $237!$ (факториал).

Для второго случая под каждый набор входных и выходных данных требуется заранее составить до $72!$ программ. Общеизвестно, что оба этих 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. Программа "УДАВ" решает все задачи для 72 объектов и 237 правил в реальном времени.

Переход от формализма "правила-переменные" к формализму "отношения-правила-переменные" в миварных сетях

На сайте www.mivar.org размещена новая версия программы УДАВ и использованы "облачные технологии". В новой версии добавлена возможность объединять группы правил в отношения. С научной точки зрения это означает расширение возможностей по обработке информации и переход от формализма "правила-переменные" к формализму "отношения-правила-переменные" в миварных сетях.

В области АСУТП в настоящее время можно внедрять интернет-технологии в производство. Появляются интегрированные системы управления, объединяющие все уровни управления производством. Будем понимать под системами управления – совокупность различных объектов, имеющих некую взаимосвязь между собой, позволяющую производить (так же упрощать, облегчать) действия, направленные на объекты (как части системы, так и других систем и отдельных объектов) для получения определённого результата.

С целью создания прототипа АСУТП такой системы управления в качестве примера было использовано описание технологических процессов в нефтепереработке и нефтехимии, предоставленное Муравьевой Е.А. Эти описания выполнены в формализме миварных сетей.

Основой создания прототипа системы управления послужил логический вывод на правилах "ЕСЛИ-ТО-ИНАЧЕ" в формализме миварных сетей (MIVAR net's). В режиме реального времени решаются задачи с большой размерностью, например, задача "треугольники" с 237 правилами продукционного вида "если - то". Однако в задачах такого типа значения были фиксированы (значение либо есть, либо нет), в зависимости от этого срабатывало определенное правило. Для логического универсального решателя задач этого было достаточно, но не совсем подходило для применения в решении задач в области АСУТП. Основным ограничением было то, что АСУТП использует аналоговые датчики, в то время как миварный подход на примере треугольников имитирует дискретные логические системы (модели): сигнал или есть, или его нет.

Особенностью предметной области АСУТП является то, что используются диапазоны некоторых величин. Например, при проведении химических процессов необходимо поддерживать в прокаточной печи значение температуры в заданном диапазоне: не ниже нижней границы и не выше верхней границы. Если температура падает ниже нижней границы, то надо выполнять некоторые действия. Если температура превышает верхнюю границу, то выполняются другие действия. Если же температура находится в заданном диапазоне, то никаких действий производить не требуется. С формальной точки зрения, это означает, что все возможные значения разбиваются на три значения: "ниже диапазона"; "норма" и "выше диапазона".

Итак, в автоматизированных системах управления от диапазона значений зависит выбранное действие. Первоначально для решения этой проблемы для каждого диапазона приходилось создавать отдельную переменную. Фактически получалось, что вместо одной переменной, например - температуры, создавалось 3 различных переменных:

- 1) "температура в диапазоне";
- 2) "температура ниже диапазона" и
- 3) "температура выше диапазона".

Такой подход позволил применять миварные сети для АСУТП, но ему были присущи следующие недостатки: недостаточная универсальность, увеличение количества переменных и правил, возможность получения противоречивых значений (когда трем переменным одной температуры могли присвоить разные значения) и т.д.

Кроме того, есть определенные процессы, где количество диапазонов одной переменной большое. Некоторые управляющие воздействия зависят от набора переменных, например, давление, влажность, температура. В таком случае подход "каждому диапазону по переменной" становился явно избыточным и затруднял решение поставленной задачи. Важно еще отметить, что решение в АСУТП зависело от шага дискретизации. Если шаг был большой - терялась точность решения, если же маленький - создавалось большое количество переменных. Это делало сложным использование метода в задачах с маленьким шагом дискретизации.

Для решения этой проблемы предложено собирать группы взаимосвязанных правил в отношения.

В задачах решения треугольников это выглядит следующим образом. В геометрии для треугольников применяют различные формулы - зависимости между определенными значениями. Например, формула: сумма углов треугольника равна 180 градусам. Эта формула связывает три переменные - три значения углов в треугольнике. Ранее для каждой переменной создавалось свое правило и эта формула записывалась в миварной сети в виде трех разных правил. Теперь все эти три правила можно объединить в одно отношение, т.е. отношение становится аналогом конкретной математической формулы. Такое отношение записывается в миварной сети. Потом для этого отношения формируются все необходимые правила: в нашем примере будет отношение "сумма углов" и три правила к нему, выражающие величину каждого угла.

Это делает миварную матрицу "двухуровневой", где на первом уровне записываются только отношения и это позволяет значительно сократить размерность миварной матрицы. А на втором уровне для каждого отношения записываются его правила. При анализе построения алгоритма сначала используются только отношения, а если выбирается какое-то отношение, то для конкретного алгоритма из этого отношения на втором уровне выбирается одно из возможных правил. Теперь сам алгоритм в основном работает на уровне отношений и это значительно сокращает размерность миварной матрицы и ускоряет получение решения.

Важно отметить, что введение "уровней миварной матрицы" является развитием теории миварных сетей. Если можно вводить два уровня представления матрицы "отношения" и "правила" для АСУТП, то теоретически для других предметных областей количество уровней может быть любым необходимым числом, а, возможно, даже и перейти в многомерное пространство - теория миварных технологий это позволяет выполнить достаточно просто и эффективно в многомерном информационном пространстве [6, 11-16].

В основе новой системы лежит технология представления информации в виде трехмерного миварного пространства. Каждый элемент системы находится в трех множествах:

- 1) V- множество названий вещей(объектов);
- 2) S- множество свойств объекта (правила, связывающие объекты);
- 3) O- множество отношений объекта.

Опишем это подробнее. Для построения миварных сетей и последующего применения вычислительных алгоритмов сначала формируется список объектов - это датчики или результаты измерений, промежуточные или расчетные переменные и требуемые значения параметров. Затем определяется список всех вычислительных процедур, процедур получения результатов измерений, логических правил перехода между разными процедурами и т.п.

На последней этапе выявляются и фиксируются взаимосвязи объектов и правил, т.е. формируется логическая схема проведения измерений и обработки их результатов. Кроме того, удобно использовать ветвление в зависимости от входных параметров (если-то-иначе). Алгоритм данного процесса представлен на рисунке 1.

Это привело к значительному сокращению переменных, что повлекло уменьшение продолжительности работы программы. Кроме того, преимуществами нашего нового подхода является исключение противоречивых ситуаций (нескольким переменным, соответствующим одному процессу с разными диапазонами, присваивается значение истины).

На практике получаем, что можно расширять (или создавать новый) используемый набор объектов, отношений и правил, создав или изменив "входной" XML файл (эта функция доступна после регистрации).

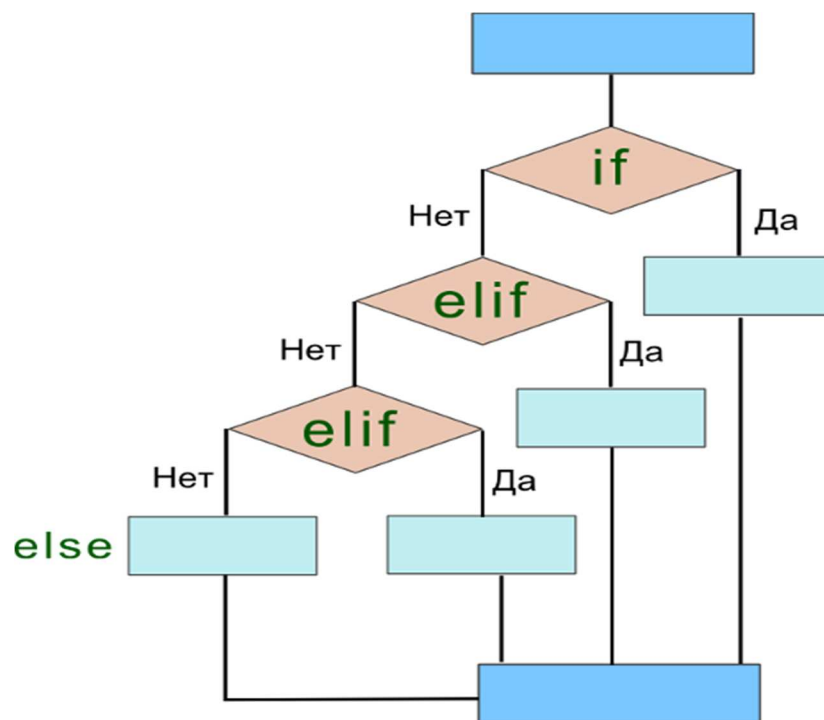


Рисунок 1

Для добавления объектов в этом файле предусмотрена секция `params`, в которую для каждого нового параметра необходимо добавить строку `paramtr` со следующими данными:

- идентификатор параметра (переменная `id`),
- значение параметра (переменная `value`),
- описание параметра для отображения в интерфейсе программы (переменная `description`).

Секция `relations` отображает отношения между объектами, где множество:

- `Y` - выходные параметры,
- `X` - входные.

Для создания нового отношения необходимо добавить строку `relation` с указанием параметров и их связи.

Для добавления нового правила предусмотрена секция `rules`, в которую для каждого нового правила необходимо добавить строку `rule` со следующими данными:

- идентификатор правила (переменная `id`);
- идентификатор отношения для данного правила и переменных, которые эти отношения определяют;
- описание правила (переменная `description`).

Почему мивары работают быстрее, чем обычные продукции?

Напомним, что мивары быстрее, чем продукции, потому что:

- в продукциях за основу поиска были взяты правила, которые перебирались для поиска решения, что порождало полный перебор, факториальную сложность и циклы;
- 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 (напомним, что миварные сети представляют собой это ориентированный двудольный граф [3-6]);
- миварная сеть может быть задана двухмерной матрицей, 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.

Особенно отметим, что представленные результаты значительно превышают "мировой уровень" научных исследований, так как впервые в мире предложено важнейшее решение по логической обработке продукционных правил с линейной вычислительной сложностью.

В настоящее время "движок" УДАВ выполняет поиск логического вывода и автоматически конструирует алгоритмы решения задач из готовых модулей-сервисов, управляемые потоком входных данных.

На обычном ноутбуке УДАВ обрабатывает более 1,17 млн переменных и 3,5 млн правил формата "если..., то..." (продукции) [3-6]. Программная реализация наглядно доказывает на практике линейную сложность поиска логического вывода, эволюционность и активность работы миварных экспертных систем нового поколения.

Для реализации метода логико-вычислительной обработки данных должно быть разработано формализованное описание предметной области. Для этого на основе миварного подхода выделяют основные объекты-переменные и правила-процедуры, а затем создают соответствующие списки "объектов" и "правил" (аналогично двудольному графу миварной логической сети). Теоретические основы миварного метода логико-вычислительной обработки информации (миварного метода) были впервые опубликованы в [1] еще в 2002 году (теоретические основы создания линейного матричного метода определения маршрута логического вывода на адаптивной сети правил). В [3-6] представлены результаты работы миварных сетей.

Этапы миварной обработки

Выделим 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 по существу является этапом синтеза концептуальной модели предметной области и ее формализации в виде продукционных правил с переходом на миварные правила. В настоящее время именно этот этап является наиболее сложным и требует участия человека-специалиста (эксперта) для создания миварной модели предметной области.

На втором этапе собственно и выполняется автоматическое конструирование алгоритма решения или логический вывод. Исходными данными для этого являются – миварная матрица описания предметной области, а также заданные входные ("ДАНО") и искомые ("НАЙТИ") объекты-переменные.

На третьем этапе должно выполняться решение по полученному алгоритму. В новой версии в программном комплексе УДАВ работы по второму и третьему этапам выполнены отдельно. Отметим, что уже создано более 7 различных реализаций миварного метода [3-6]. Часть этих реализаций в виде программных комплексов выделяет все три основных этапа и обрабатывает их отдельно. В данной работе основное внимание уделим программному комплексу УДАВ, в котором все три этапа могут быть совмещены.

Кратко опишем теоретические основы работы миварного метода логико-вычислительной обработки данных [1-6]. Напомним, что практические реализации в виде конкретных алгоритмов и программных комплексов могут отличаться от приведенного описания, которое является наиболее общим и носит теоретический характер.

Прежде всего, для миварной сети логических правил, представленной в списочной форме, строится матрица. Затем, на основе анализа этой матрицы определяется факт наличия успешного маршрута вывода, потом определяются возможные маршруты логического вывода, а на последнем этапе из этих маршрутов выбирают "кратчайший", наиболее оптимальный по заданным критериям оптимальности.

Пусть известны m правил и n переменных (входящих в правила либо в качестве исходных, активизирующих их, либо в качестве получаемых, то есть выходных). Тогда в матрице V ($m \times n$), каждая строка которой соответствует одному из правил и содержит информацию об используемых в правиле переменных, могут быть представлены все взаимосвязи между правилами и переменными. При этом в каждой строке все входные переменные этого правила на соответствующих позициях матрицы помечаются символом x , все выходные – y ; все переменные, которые уже получили в процессе вывода или задания исходных данных некоторое конкретное значение, – z ; а все искомые (выходные) переменные, то есть те, которые необходимо "вывести" из исходных (входных) данных, – w .

Кроме того, добавим в матрицу V одну строку и один столбец для хранения в них служебной информации.

Напомним, что количество служебной информации может видоизменяться для различных конкретных реализаций программных комплексов. Получаем матрицу V размерности $(m+1) \times (n+1)$, в которой отражена вся структура исходной сети правил. Сразу отметим, что структура этой логической сети может изменяться в любое время, то есть это сеть правил с изменяемой (эволюционной) структурой.

Опишем пример работы метода. Для поиска маршрута логического вывода на полученной матрице производят действия по следующим этапам [6].

В строке $(m+1)$ известные переменные помечают z , а искомые – w .

Осуществляют последовательно, например, сверху вниз, поиск таких правил, которые могут быть активизированы, то есть у которых известны все входные переменные. Если таких правил нет, то маршрута логического вывода нет и необходимо выдать запрос на уточнение (добавление) входных данных. Если такие правила, которые могут быть активизированы, есть, то у каждого из них в соответствующем месте служебной строки делается пометка, что правило может быть запущено. Например, можно ставить в этих клетках таблицы цифру 1.

Если таких правил несколько, то осуществляется выбор по заранее определенным критериям такого или таких правил, которые должны быть активизированы в первую очередь. При наличии достаточных ресурсов, одновременно могут запускаться сразу несколько правил. В таком случае логический вывод распараллеливается и это тоже важное преимущество.

Имитация запуска правила (процедуры) осуществляется путем присваивания выводимым в этом правиле переменным значений "известно", то есть – z . Запущенное правило, для удобства дальнейшей работы помечается дополнительно, например, цифрой 2 (это не обязательно).

После имитации запуска правил проводят анализ достижения цели, то есть анализируют получение требуемых значений путем сравнения служебных символов в служебной строке. Если в служебной строке $(m+1)$ осталось хоть одно значение "искомая" (то есть w), то осуществляют дальнейший поиск маршрута логического вывода. В противном случае, задача считается успешно решенной, а все задействованные правила в соответствующем порядке их запуска и образуют искомый маршрут логического вывода.

Прежде всего, определяют наличие таких правил, которые могут быть запущены после определения новых значений на предыдущем этапе. Если таких правил нет, то маршрута логического вывода нет и необходимо выдать запрос на уточнение (добавление) входных данных.

Если такие правила есть, то продолжают поиск маршрута вывода.

На следующем этапе опять аналогично запускают правила (имитируют запуск), далее аналогично выполняют необходимые действия столько раз, сколько требуется для получения результата. При необходимости, повторяют этапы до достижения результата. При этом результат может быть как положительный – маршрут вывода существует, так и отрицательный – вывода нет из-за неопределенности входных данных.

В работах [3-6] этот метод и приведенный пример описан подробно и с рисунками в формализмах матриц и графов. Напомним, что работы [3-5] находятся в открытом доступе на сайтах соответствующих журналов и препринтов. Следовательно, с ними может ознакомиться любой человек.

Применение облачных технологий для реализации комплекса УДАВ

Дополнительным преимуществом новой версии программного комплекса УДАВ является то, что он реализован на облачных технологиях. При сравнении с локальным размещением программных продуктов, облачные технологии дают существенный прирост производительности систем в целом, а также заметную экономию дискового пространства и повышение эффективности использования процессорной мощности. В качестве недостатка данной технологии можно отметить потребность в устойчивом сетевом соединении клиента с облаком, хотя данный недостаток не является существенным в современных системах.

На основе облачных технологий реализована система, позволяющая найти алгоритм и вычислить необходимые параметры модели по массиву входных переменных, т.е. - это ядро комплекса УДАВ. Эта реализация базируется на основных принципах миварного подхода и использует все преимущества облачных технологий.

Отметим, что реализованная система является универсальным блоком для построения экспертных систем, систем искусственного интеллекта и автоматизированных систем обработки информации и управления (АСОИУ).

На вход системы поступает HTTP запрос типа "POST" с массивом входных данных и списком искомых параметров. Далее, согласно выбранной модели, составляется оптимальный алгоритм для вычисления требуемых параметров. После чего происходит математическое вычисление согласно составленному ранее алгоритму.

Результат работы комплекса УДАВ оформляется в виде ответа на HTTP запрос в виде массива всех известных данных, а также XML кодом с обозначенным порядком запуска необходимых правил.

Комплекс УДАВ расположен на виртуальном сервере одного из крупнейших провайдеров облачных технологий в России - компании DataLine. Текущий сервер соответствует высоким стандартам надежности, как в плане бесперебойности электропитания, так и в плане дублирования сетевого канала связи. Выделенный "облачный сервер" физически имеет в распоряжении два процессора Intel Xeon с восемью доступными ядрами и частотой 2.67ГГц, 24 Гб оперативной памяти и 150 Гб дискового пространства с непрерывным периодическим резервированием данных на обособленный носитель. Используется операционная система Microsoft Windows Server 2008R2, 64-bit которая обеспечивает надежное управление сервером с небольшими трудозатратами. В качестве внешнего интерфейса для коммуникаций был выбран сервер Apache, в сочетании с интерпретатором языка PHP.

Сервер доступен из сети интернет, но имеет несколько степеней защиты от несанкционированного доступа.

Для общественного доступа к ограниченному функционалу портала с целью тестирования, организован доступ по web – адресу: <http://www.mivar.org> .

В качестве важного преимущества стоит отметить масштабируемость виртуального облака. При необходимости, за минимальное время, ресурсы можно перераспределить как в сторону увеличения производительности, так и в сторону экономии затрат на сервис. Все это применимо и для нашей реализации программного комплекса УДАВ: по мере роста количества запросов от пользователей будут выделяться дополнительные ресурсы.

Преимущества и новизна миварного метода обработки

Для универсального решателя задач и экспертных систем на основе продукционного подхода и вместо традиционных однодольных графов необходимо и достаточно использовать двудольные графы миварных сетей. С научно-практической точки зрения основную сложность при использовании миварных систем составляет именно концептуальное, продукционное, описание предметной области и формирование необходимых двух списков: объектов и правил для миварных логических сетей.

Непосредственная обработка проводится на основе универсального механизма. Выделенные правила-процедуры при необходимости могут быть представлены в виде продукции, что соответствует традиционному подходу, но в то же время позволяет реализовать многомерное эволюционное представление данных. Универсальные возможности миварного подхода обусловлены тем, что он обобщает все известные модели данных, включая семантические сети, модель "сущность-связь" и онтологии [4, 10, 15, 18-20].

Преимуществами миварного подхода являются:

- линейная вычислительная сложность и реальное время работы;
- решение логических и вычислительных (и других) задач;
- управление потоком входных данных и оперативная диагностика;
- адаптивное описание и непрерывное решение задач;
- активная работа с запросами или уточнениями входных данных на эволюционной сети правил и объектов (самообучение).

Вычислительная сложность миварного метода обработки

Прежде всего отметим, что миварные сети основаны на том, что каждое правило при решении конкретной задачи запускается максимум один раз. Правило может не запускаться совсем, т.е. 0 (ноль) раз. Но более одного раза его нет смысла запускать и это отслеживается в алгоритме миварного метода обработки.

Есть отдельные модификации ("два прохода") миварного метода, когда сначала выполняется поиск "сверху-вниз" и активизируются все возможные правила, а затем выполняется дополнительный проход по матрице "снизу-вверх" и отсекаются лишние правила и остаются только такие правила, которые необходимы для решения задачи. В таком случае правила рассматриваются 2 раза.

Общее количество действий при миварном методе (в классическом варианте "одного прохода сверху-вниз" по матрице) определяется суммой действий на каждом этапе:

- присваивание известных z и искомым w значений клеткам служебной строки $(m+1)$ (общее количество таких действий не более n);
- присвоение признака обработки правил в служебном столбце $(n+1)$ (количество действий не более $2m$, но может быть не более числа m);
- присваивание признака известности (z) клеткам служебной строки $(m+1)$ (общее количество таких действий не более n);
- определение новых значений клеткам строки $(m+1)$ (количество таких действий не более n).

Действия на первых этапах выполняются над одним массивом данных – служебной строкой матрицы $(m+1)$. При этом общее, суммарное, количество действий на всех этих этапах не должно превышать общего количества клеток в этой строке, так как обработанные значения "вычеркиваются" и более не обрабатываются. Получаем, что общее количество действий (KD) при миварном методе обработки, то есть его вычислительная сложность, не превышает количества клеток в служебных частях матрицы:

$$O(m+n), \text{ то есть } KD \leq (m+n)$$

В том случае, когда нельзя реализовать предложенные сокращения вычислений, этот метод решает задачу поиска маршрута логического вывода с вычислительной сложностью:

$$KD = O(m*n) \Rightarrow \{KD = O(m*Const) \text{ или } KD = O(Const*n) \text{ или } KD = O(Const*Const)\}$$

то есть имеет место линейная зависимость либо от количества правил ($n=Const$), либо от количества переменных ($m=Const$).

Важно отметить, что в худшем случае по каждой клетке "проходят" один раз, что исключает повторы и циклы. Кроме того, каждое правило может запускаться только один и именно этим определяется линейная вычислительная сложность миварного метода поиска логического вывода.

Ведь если каждое правило запускается только один раз, то и максимальное количество операций будет равно количеству правил - здесь именно линейная зависимость. Конечно, у нас есть варианты реализации миварного метода, когда для оптимизации полученного алгоритма решения выполняется два прохода: сначала "сверху-вниз" (от заданных к искомым переменным) ищутся все возможные запускаемые правила, а потом (на втором этапе) выполняется проход по матрице "снизу-вверх" (от искомым к заданным переменным) и отсекаются все лишние ветви алгоритма, которые не используются для решения требуемой задачи. В такой реализации вычислительная сложность будет квадратичной, относительно количества

правил. Но это уже выполняется оптимизация, которая реализуется дополнительным этапом и усложняет алгоритм логического вывода.

В работах [3-6] изложены результаты практической реализации в программном комплексе "УДАВ" миварных сетей.

В работе [5] миварные технологии представлены в зарубежных электронных изданиях. Именно в исследовании вопросов описания, обработки и представления контекста ведутся перспективные разработки миваров в новых областях их применения. Следовательно, доказано, что существует достаточно универсальный миварный подход для решения многих практических задач применения информационных систем и процессов с одновременным проведением как логической, так и вычислительной обработки данных.

Существовавшее ранее противоречие между логическим выводом и вычислительной обработкой успешно преодолено с помощью миварных сетей и миварных баз данных и правил.

Таким образом, "компьютеры значительно поумнели" и можно использовать эти достижения миваров для создания нового поколения информационных технологий в области информационной безопасности, автоматизированных системах обработки информации и АСУТП.

Выводы

В работе представлена "он-лайн" версия программного комплекса УДАВ, который реализован на основе облачных технологий и миварного метода логико-вычислительной обработки данных. В режиме реального времени решаются задачи с большой размерностью, например, задача "треугольники" с 237 правилами продукционного вида "если - то".

Логический вывод с 2002 года НЕ является NP-полной задачей, т.к. для него существует линейной вычислительной сложности алгоритм логического вывода на продукциях "ЕСЛИ-ТО" в формализме миварных сетей (MIVAR net's).

Предложено развитие миварной теории создания логического искусственного интеллекта путем перехода от формализма "правила-переменные" к формализму "отношения-правила-переменные" в миварных сетях и введением многоуровневого и многомерного представления правил.

Представлены результаты создания новой версии миварного универсального решателя задач УДАВ, способного выполнять адаптивный активный логический вывод на двухуровневом представлении "отношения-правила-переменные".

Универсальный решатель задач УДАВ реализован на основе облачных технологий и размещен на сайте <http://www.mivar.org/>. Экспериментально подтверждена теоретическая линейная вычислительная сложность логического вывода и автоматического конструирования алгоритмов решения различных задач на основе миварных сетей.

Показано, что универсальный решатель задач УДАВ может быть использован для создания АСУТП. В близкой перспективе УДАВ практически полностью может

заменить человека-оператора в автоматизированных системах обработки информации и АСУТП.

Можно утверждать, что теоретические основы создания логического искусственного интеллекта созданы. Особенно отметим, что представленные результаты значительно превышают "мировой уровень" научных исследований, так как впервые в мире предложено важнейшее решение по логической обработке продукционных правил с линейной вычислительной сложностью. Ознакомиться с результатами нашей работы можно на сайте www.mivar.org, реализованном на облачных технологиях.

Список информационных источников

- [1] Варламов О.О. Разработка линейного матричного метода определения маршрута логического вывода на адаптивной сети правил // Известия вузов. Электроника. 2002. № 6. С. 43-51.
- [2] Владимиров А.Н., Варламов О.О., Носов А.В., Потапова Т.С. Программный комплекс "УДАВ": практическая реализация активного обучаемого логического вывода с линейной вычислительной сложностью на основе миварной сети правил // Труды НИИР. 2010. Т. 1. С. 108-116.
- [3] Варламов О.О. Миварные технологии: переход от продукций к двудольным миварным сетям и практическая реализация автоматического конструктора алгоритмов, управляемого потоком входных данных и обрабатывающего более трех миллионов продукционных правил // Искусственный интеллект. 2012. № 4. С. 11-33.
- [4] Варламов О.О. Практическая реализация линейной вычислительной сложности логического вывода на правилах "ЕСЛИ-ТО" в миварных сетях и обработка более трех миллионов правил // Автоматизация и управление в технических системах. – 2013. – № 1(3); [Электронный ресурс]. URL: <http://auts.esrae.ru/3-66> (дата обращения: 26.03.2013).
- [5] Varlamov O.O. MIVAR: Transition from Productions to Bipartite Graphs MIVAR Nets and Practical Realization of Automated Constructor of Algorithms Handling More than Three Million Production Rules // ARXIV.ORG. 05.11.2011. [Электронный ресурс] URL: <http://arxiv.org/abs/1111.1321>. (дата обращения: 31.01.2013).
- [6] Варламов О.О. Логический искусственный интеллект создан на основе миварного похода! МИВАР: активные БД с линейным логическим выводом > 3 млн правил => понимание смысла+ сингулярность в виртуальной реальности. - Саарбрюкен, Германия: LAP LAMBERT Academic Publishing Gmbh & Co. KG, 2012. - 700 с. ISBN: 978-3-8473-1953-5.
- [7] Варламов О.О. О системном подходе к созданию модели компьютерных угроз и ее роли в обеспечении безопасности информации в ключевых системах информационной инфраструктуры // Известия Таганрогского государственного радиотехнического университета. 2006. Т. 62. № 7. С. 216-223.
- [8] Варламов О.О., Амарян М.Р., Адамова Л.Е. и др. Подход к защите информации в АСУ оператора связи на основе миварных баз данных и правил // Известия Таганрогского государственного радиотехнического университета. 2003. Т. 33. № 4. С.174-175.
- [9] Варламов О.О., Амарян М.Р., Адамова Л.Е. Особенности защиты персональных данных и информации в АСУ регионального оператора связи // Известия

- Таганрогского государственного радиотехнического университета. 2003. Т. 33. № 4. С. 238-239.
- [10] Варламов О.О., Амарян М.Р., Лысаковский В.А., Адамова Л.Е. Подход к защите информации на основе локальных корректировок вычислений и обработки данных // Известия Таганрогского государственного радиотехнического университета. 2003. Т. 33. № 4. С. 239-240.
- [11] Варламов О.О. Обзор 25 лет развития миварного подхода к разработке интеллектуальных систем и создания искусственного интеллекта // Труды НИИР. 2011. № 1. С. 34-44.
- [12] Варламов О.О. Основы многомерного информационного развивающегося (миварного) пространства представления данных и правил // Информационные технологии, 2003. № 5. С. 42-47.
- [13] Варламов О.О. Параллельная обработка потоков информации на основе виртуальных потоковых баз данных // Известия высших учебных заведений. Электроника. 2003. № 5. С. 82.
- [14] Варламов О.О. Разработка метода распараллеливания потокового множественного доступа к общей базе данных в условиях недопущения взаимного искажения данных // Информационные технологии. 2003. №1. С. 20.
- [15] Варламов О.О. Системный анализ и синтез моделей данных и методы обработки информации для создания самоорганизующихся комплексов оперативной диагностики // Искусственный интеллект. 2003. № 3. С. 299.
- [16] Варламов О.О. Системный анализ и синтез моделей данных и методы обработки информации в самоорганизующихся комплексах оперативной диагностики : дис. ... док. техн. наук. – М.: МАРТИТ, 2003. 307 с.
- [17] Варламов О.О. Системы обработки информации и взаимодействие групп мобильных роботов на основе миварного информационного пространства // Искусственный интеллект. 2004. № 4. С. 695.
- [18] Тоноян С.А., Балдин А.В., Елисеев Д.В. Методика модернизации стандартных модулей типовой конфигурации на базе технологической платформы "1С:предприятие 8" с минимальными доработками // Наука и образование: электронное научно-техническое издание. 2012. № 08. С. 17.
- [19] Балдин А.В., Елисеев Д.В., Агаян К.Г. Обзор способов построения темпоральных систем на основе реляционной базы данных // Наука и образование: электронное научно-техническое издание. 2012. № 08. С. 20.
- [20] Балдин А.В., Елисеев Д.В. Адаптация темпоральной реляционной модели данных в многомерном пространстве // Наука и образование: электронное научно-техническое издание. 2011. № 09. С. 1.
- [21] Елисеев Д.В., Балдин А.В. Алгебра многомерных матриц для обработки адаптируемой модели данных // Наука и образование: электронное научно-техническое издание. 2011. № 07. С. 4.
- [22] Балдин А.В., Елисеев Д.В. Адаптируемая модель данных на основе многомерного пространства // Наука и образование: электронное научно-техническое издание. 2010. № 10. С. 1.
- [23] Елисеев Д.В. Модель представления знаний при создании адаптивной информационной системы // Наука и образование: электронное научно-техническое издание. 2010. № 03. С. 2.
- [24] Варламов О.О. Подход к формированию многоуровневой модели мультиагентной системы с использованием миваров / А.В. Остроух, А.И. Белоусова, М.Н. Краснянский, О.О. Варламов // Перспективы науки. – 2011. - № 5(20). - С. 57 – 61.

- [25] Варламов О.О. Анализ возможностей миварного подхода для систем искусственного интеллекта и современной робототехники / А.В. Остроух, М.Н. Краснянский, Т.Л. Давыдова, О.О. Варламов // Вестник ТГТУ. - 2011. - Т.17. - № 3. - С.687-694.
- [26] Варламов О.О. Миварные автоматизированные системы управления технологическими процессами для нефтяной промышленности России / А.В. Остроух, Р.А. Санду, О.О. Варламов // Автоматизация, телемеханизация и связь в нефтяной промышленности. – 2011. - №11. - С.37-41.
- [27] Остроух, А.В. Основы построения систем искусственного интеллекта для промышленных и строительных предприятий / А.В. Остроух - М.: ООО «Техполиграфцентр», 2008. - 280 с. - ISBN 978-5-94385-033-2.
- [28] Остроух, А.В. Информационные технологии в научной и производственной деятельности / [ред. А.В. Остроух] - М: ООО "Техполиграфцентр", 2011. - 240 с. - ISBN 978-5-94385-056-1.