

Шамшев А. Б.

[МЕХАНИЗМЫ РЕАЛИЗАЦИИ МНОГОПОТОКОВОГО БАЙЕСОВСКОГО ЛОГИЧЕСКОГО ВЫВОДА](#)

Адрес статьи: www.gramota.net/materials/1/2008/12/74.html

Статья опубликована в авторской редакции и отражает точку зрения автора(ов) по данному вопросу.

Источник

[Альманах современной науки и образования](#)

Тамбов: Грамота, 2008. № 12 (19). С. 236-239. ISSN 1993-5552.

Адрес журнала: www.gramota.net/editions/1.html

Содержание данного номера журнала: www.gramota.net/materials/1/2008/12/

[© Издательство "Грамота"](#)

Информация о возможности публикации статей в журнале размещена на Интернет сайте издательства: www.gramota.net

Вопросы, связанные с публикациями научных материалов, редакция просит направлять на адрес: almanac@gramota.net

19. Сингер С. Природа шаровой молнии. - М.: Мир, 1973. - 239 с.
20. Смирнов Б. М. Проблема шаровой молнии. - М.: Наука, 1988. - 208 с.
21. Стаханов И. П. О физической природе шаровой молнии. - М.: Энергоатомиздат, 1985. - 209 с.
22. Холмид Л., Маныкин Э. // ЖЭТФ. - 1997. - Т. 111. - № 5. - С. 1601-1610.
23. Шавлов А. В. Параметры шаровой молнии, вычисляемые с помощью двухтемпературной плазменной модели. – Тюмень: Институт криосферы Земли СО РАН, 2008. - 17 с. Ил. 3. Библиогр. 17. Назв. рус. Деп. в ВИНТИ 04.03.08 № 195-В2008.

МЕХАНИЗМЫ РЕАЛИЗАЦИИ МНОГОПОТОКОВОГО БАЙЕСОВСКОГО ЛОГИЧЕСКОГО ВЫВОДА

Шамшев А. Б.

Ульяновский государственный технический университет

Необходимость реализации многопоточного вывода

Современная тенденция развития вычислительной техники состоит в увеличении количества вычислительных ядер в процессорах. На сегодняшний день максимальное количество ядер в одном процессоре – 4, однако производители процессоров анонсируют процессоры с большим количеством ядер (например, AMD планирует через 2 – 3 года выпустить процессор с 12 ядрами). Также не следует забывать про графические процессоры. Например, процессор производства фирмы nVidia GTX 280 обладает двумястами сорока процессорными ядрами (младшая версия данного процессора – GTX 260 – сто девяносто ядер) которые благодаря технологии CUDA можно использовать для неграфических вычислений. Использование параллельного программирования позволяет использовать вычислительные способности процессоров гораздо более эффективно, чем в традиционном однопоточковом программировании. Очевидно, что рациональное использование многоядерности позволяет получить существенный выигрыш в производительности.

Сложности реализации многопоточных приложений

Основной проблемой при реализации многопоточных приложений является необходимость взаимодействия потоков. Взаимодействие происходит с использованием общих переменных. Т.е. несколько потоков должны ждать, пока другой поток не сформирует необходимое значение. Так же действия потоков не должны приводить к искажению общих данных. Из этого следует, что для реализации многопоточного приложения необходимо также реализовать механизм синхронизации действий потоков, что приводит к возникновению накладных расходов на управление потоками. Современные среды программирования, в частности платформа .NET Framework, разработанная компанией Microsoft, имеет в своём составе средства для организации многопоточковых приложений. Из вышесказанного можно сделать следующий вывод: для реализации задачи байесовского логического вывода необходимо разбить задачу на подзадачи таким образом, чтобы они как можно меньше зависели друг от друга и как можно меньше взаимодействовали друг с другом.

Байесовская сеть доверия

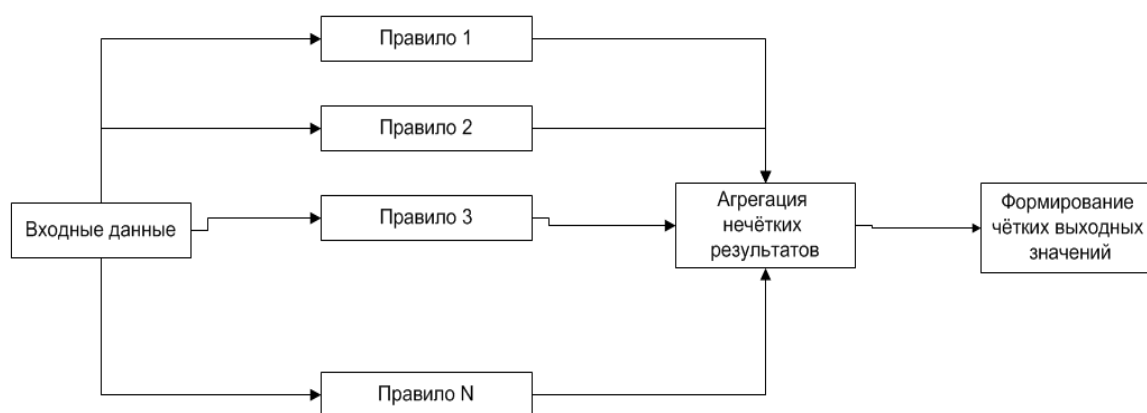
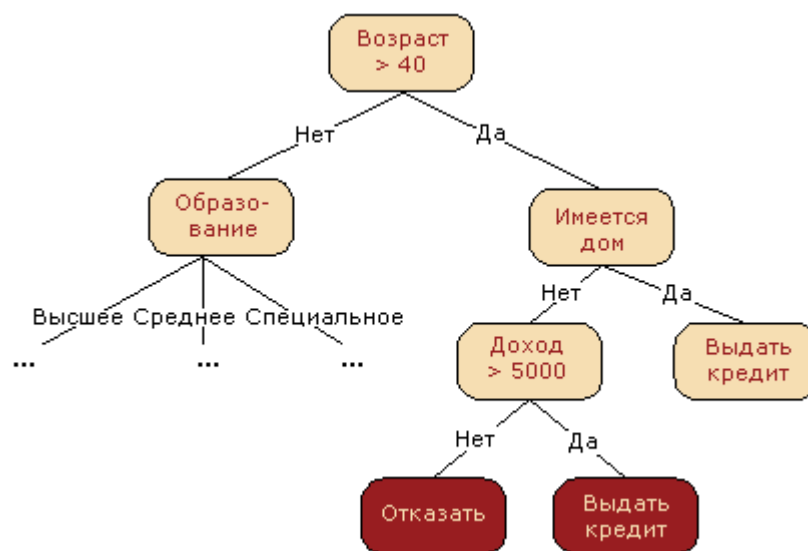
Байесовская сеть доверия (БСД) представляет собой направленный ациклический граф, структура которого задаёт причинно следственные связи между значимыми переменными задачи, заданными вершинами графа. Каждая вершина представлена набором состояний, в которых она может находиться с различной степенью вероятности. Таким образом, каждая вершина графа может быть истолкована как соответствующее нечёткое понятие, имеющее некоторое количество соответствующих нечётких значений. Каждая вершина (кроме начальных) имеет матрицу условных вероятностей, связывающую вероятности нахождения вершины в том или ином состоянии в зависимости от состояния вершин – предков. Таким образом, каждая матрица условных вероятностей может быть истолкована как набор правил «Если - То». Следовательно, логический вывод по сети доверия можно интерпретировать как нечёткий логический вывод.

Многопоточковый нечёткий вывод

Для того, чтобы выполнить логический вывод по сети доверия при помощи нечёткого логического вывода, сеть доверия необходимо преобразовать в дерево решений. Одним из вариантов такого преобразования является использование библиотеки MSBNx. Деревья решений – это способ представления правил в иерархической, последовательной структуре, где каждому объекту соответствует единственный узел, дающий решение. Под правилом понимается логическая конструкция, представленная в виде "если ... то ...".

Таким образом, задача логического вывода по сети доверия сводится к иерархическому нечёткому выводу. Для реализации механизма многопоточкового нечёткого вывода удобно брать один уровень дерева решений, так как само дерево можно получить, последовательно соединив несколько механизмов.

Данный механизм должен произвести логический вывод по набору правил, поэтому логичным является решение выделить каждому правилу отдельный вычислительный поток (или в одном потоке производить вывод по нескольким правилам). Схема такого вывода представлена на рисунке ниже:



Распараллеливание логического вывода возможно потому, что выходные данных потоки только читают, но не изменяют. Два последних этапа необходимо выполнять в однопоточном варианте, так как они связаны с обработкой данных всех потоков, обрабатывающих правила.

Автором по данной схеме реализован механизм нечёткого вывода на языке программирования C# в среде Visual Studio 2005. Тестовые примеры показали, что на двуядерном процессоре многопоточный механизм работает в среднем в два раза быстрее, чем однопоточный. Из схемы процесса видно, что прирост производительности пропорционален количеству вычислительных ядер.

Недостатки данного механизма следующие:

1. Результатом вывода по сети доверия является распределение вероятностей, характеризующее вероятности нахождения узлов сети в определённых состояниях. Создание правил, имитирующих такое распределение, приводит к резкому увеличению объёма базы правил.

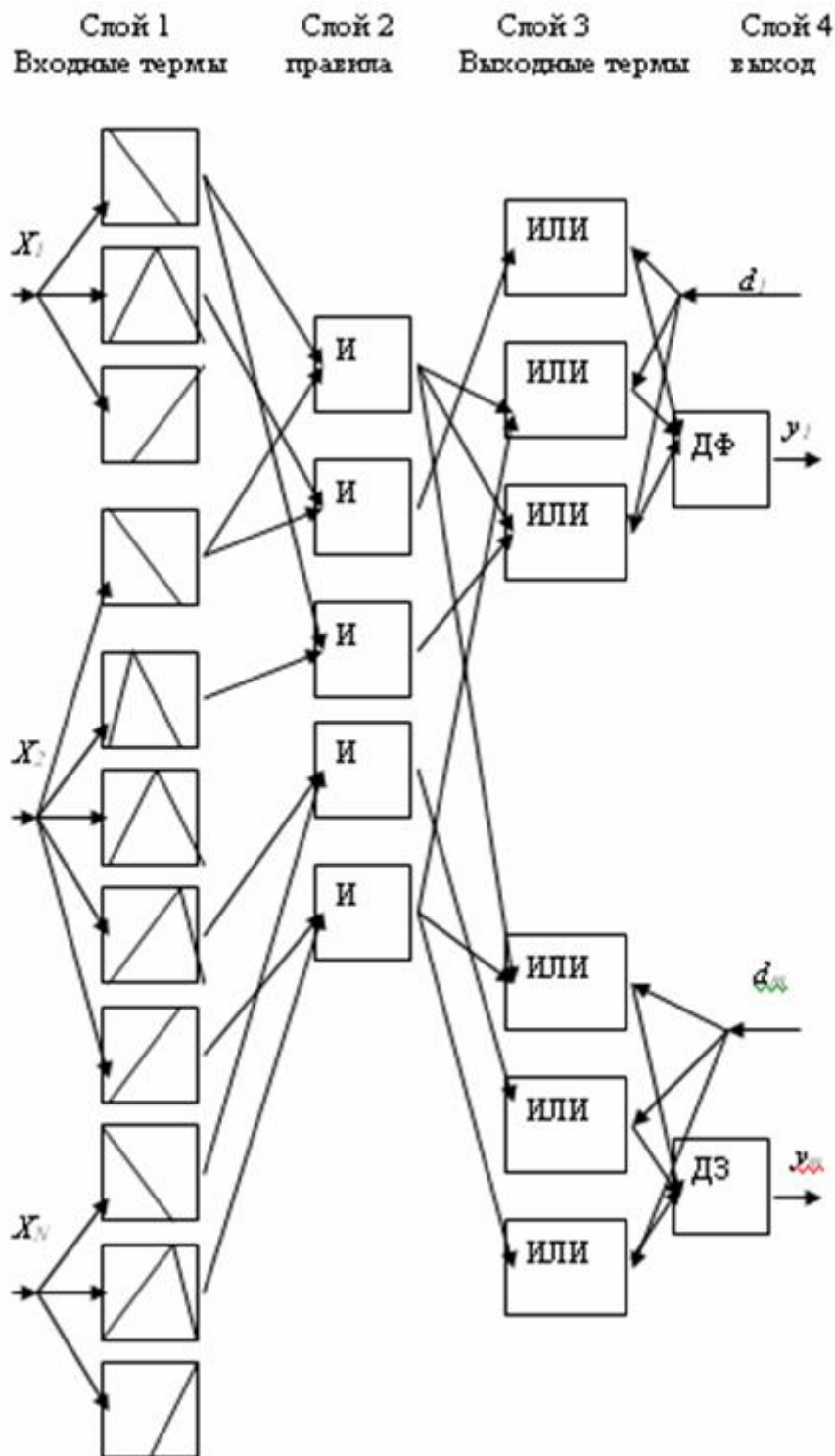
2. Если сеть доверия обладает сложной структурой с большим количеством связей между вершинами, то дерево решений получается очень широким и объёмным. Из этого следует, что, база правил также получается сложной и объёмной. Вывод по такой базе представляется весьма затруднительным. Эта проблема в MSBNx также не решена.

Таким образом, можно сделать вывод о том, что механизм нечёткого многопоточного вывода способен заменить логический вывод по БСД сравнительно небольшого размера и простой структуры.

Нечёткая нейронная сеть

Одним из способов преодоления частичного преодоления указанных недостатков может быть использование вместо механизма нечёткого вывода нечёткой нейронной сети. Основным преимуществом нейронной сети является то, что её структура и алгоритмы функционирования изначально предполагают высокую степень параллельности работы.

Структура нечёткой нейронной сети в определённой степени повторяет структуру механизма нечёткого вывода и может быть представлена следующим образом:



Нечёткая нейронная сеть отличается от чёткой нейронной сети функциями, которые выполняют нейроны различных слоёв.

Основной недостаток нечёткой нейронной сети заключается в том, что для правильного функционирования нечёткую сеть необходимо предварительно обучить. Для этого необходима априорная обучающая выборка наиболее типичных данных и правильные результаты вывода для отобранных данных. Получить такую выборку можно далеко не во всех случаях, что резко ограничивает возможности применения нечёткой нейронной сети.

Сети Петри

Сети Петри – одна из наиболее популярных моделей параллельных систем, применяемых для исследований и моделирования объектов, как в теоретических, так и в практических областях. К достоинствам сетей Петри относятся их наглядное графическое представление и возможность автоматизированного анализа их свойств. Наиболее наглядным представлением сети Петри может быть двудольный ориентированный граф, аналогичный графу сети доверия. Функционирование сети Петри моделируется перемещением фишек или

тоkenов между узлами сети. Очевидно, что если узел сети Петри будет представлять конкретный узел сети доверия, а фишка будет переносить информацию об изменившихся вероятностях, то сеть Петри может стать аналогом соответствующей байесовской сети доверия и логический вывод по сети доверия может быть представлен как перемещение фишек в сети Петри. Преимуществом подобного решения является то, что сеть Петри изначально ориентирована на параллельность выполнения передвижения фишек и не требует предварительного обучения. Следовательно, именно сеть Петри наилучшим образом подходит для реализации многопоточного нечёткого вывода из всех рассмотренных в статье механизмов.

Для того, чтобы реализовать байесовский логический вывод при помощи сети Петри необходимо выполнить следующие действия:

1. Сопоставить узлам сети Петри узлы сети доверия. Каждому узлу сети Петри приписывается информация о возможных состояниях и их вероятностях, а также матрица условных вероятностей.

2. При изменении вероятностей в некоторых узлах, они начинают пересылать фишки связанным с ними узлам. Подобная пересылка возможна потому что количество фишек в сети Петри не является постоянной величиной. Каждая фишка несёт информацию о новом состоянии узла, её выпустившую.

3. Узел, получивший фишку, производит пересмотр информации о собственных вероятностях. Если вероятности изменились, то узел рассылает связанным с ним узлам информацию об изменениях при помощи новых фишек. Если изменения не произошло, то полученная фишка поглощается.

Таким образом, итерационный процесс, порожаемый введением в сеть Петри новой информации, приводит к обновлению всей информации в сети. Единственным местом, в котором возникает взаимодействие различных потоков, является получение узлами всех необходимых фишек для обновления информации. Очевидно, что количество взаимодействий будет существенно меньше, чем для байесовской сети доверия. Таким образом, реализация многопоточного нечёткого вывода на базе сетей Петри может оказаться очень перспективной и способствовать повышению эффективности использования вычислительных возможностей современных процессоров.

Список использованной литературы

1. **Шамшев А. Б.** Автоматизированное топологическое проектирование вычислительных сетей на основе байесовских сетей доверия: Автореф. дис. канд. техн. наук. - Ульяновск, 2006.
2. **Шамшев А. Б.** Применение байесовских сетей доверия в задачах нечёткого контроля (42 НТК) // Вузовская наука в современных условиях. – Ульяновск: УлГТУ, 2008. – Ч. 1.
3. **Horvitz E., Hovel D., Kadie C.** MSBNx: A Component-Centric Toolkit for Modeling and Inference with Bayesian Networks // MSDN. - July 2001.

ЯЗЫК МАТЕМАТИЧЕСКОЙ НАУКИ И ЯЗЫК ОБУЧЕНИЯ МАТЕМАТИКЕ

Шармин Д. В.

ГОУ ВПО «Тюменский государственный университет»

Б. В. Гнеденко писал: «Для общения и выражения своих мыслей люди создали величайшее средство – живой разговорный язык и его письменную запись... Несмотря на свою многогранность и гибкость, в ряде случаев язык человека оказывается недостаточным и более того – неудовлетворительным средством общения. Поэтому в различных областях деятельности вырабатываются свои собственные языки, специально приспособленные для точного и краткого выражения мыслей, системы действий, правил поведения, свойственных определенным видам человеческой активности» [Гнеденко 1991: 29]. В приведенном высказывании речь идет о языках различных наук, одним из которых и является язык математики.

Построение искусственных языков науки связано с формализацией. В математике под формализацией обычно понимают отображение содержательного знания в формализованном языке, иначе говоря, в виде знаков (символов). Полная формализация какой-либо математической теории достигается лишь тогда, когда отвлекаются от содержательного смысла исходных понятий и аксиом теории и полностью перечисляют правила логического вывода теорем из аксиом.

Важно отметить, что формализация – это не просто замена отдельных представлений описательной теории, выраженных словами естественного языка, символическими выражениями. Главное заключается в построении системы нового языка со своим особым синтаксисом. «Это и нужно считать, – пишет А. Черч, – главной отличительной частью формализованного языка, а вовсе не то, что оказалось удобным заменить определенными буквами и различными специальными символами слова, которые в письменности большинства обычных языков составляются из многих букв. Эта замена, хотя она и бросается сразу в глаза, теоретически менее важна» [Кикель 1989: 31].

А. А. Столяр [Столяр 1965] отмечает, что язык математики – явление далеко не однородное. По существу, каждый раздел математики пользуется своим особым языком. Язык каждой математической теории, и язык математики в целом, есть язык логико-математический ($Я_{ЛМ}$). Он состоит из двух компонентов: $Я_M$ и $Я_Л$. $Я_M$ – язык данной математической теории, состоящий из специфических терминов и символов, обозначающих объекты, свойства и отношения объектов множества, структура которого описывается этой теорией. $Я_Л$ – логический язык, состоящий из терминов и символов, обозначающих логические операции, исполь-