УДК 681.326:519.713

В.И. ХАХАНОВ, О.А. ГУЗЬ, NGENE CHRISTOPHER UMERAH,
В. ОЛЬХОВОЙ

ПРОЦЕСС-МОДЕЛИ АНАЛИЗА АССОЦИАТИВНЫХ СТРУКТУР ДАННЫХ

Параллельные векторные мультипроцессорные неарифметические процесс-модели, представленные в исследованиях, ориентированы на новые эффективные решения практических задач синтеза и анализа: минимизация булевых функций, поиск дефектов, восстановление работоспособности, распознавание образов, принятие решений, разработка цифровых фильтров, создание дружественных серверов, сайтов и порталов.

1. Введение

Мозгоподобность (функциональная) предполагает исключение арифметики и всех нелогических критериев (вероятность, функция принадлежности), которые до сих пор исполняют роль фигового листка на голом теле нашего логического невежества. Мозг не знает арифметики, кроме нелогической таблицы умножения, но ежесекундно (ежеминутно) в реальном масштабе времени принимает логически обоснованные решения. Цель работы заключается в создании параллельных векторных мультипроцессорных неарифметических процесс-моделей для существенного уменьшения времени анализа ассоциативных структур данных. Задача исследования – разработка моделей, методов и средств, включающих быстродействующие мозгоподобные компьютеры, критерия качества взаимодействия объектов в пространстве для реализации эффективного поиска, распознавания и принятия решений.

Источники: 1. Технологии параллельных вычислений на основе специализированных мультипроцессорных систем [1, 2, 10, 11, 15]. 2. Алгебраические структуры, ориентированные на создание математического аппарата параллельных вычислений [3, 4, 7 – 10]. 3. Процесс-модели для решения задач реального времени на основе эффективных параллельных вычислений [5, 6, 11, 13].

2. Параллельные векторные мультипроцессорные неарифметические процесс-модели

Вектор есть ассоциация. Алгебра ассоциативной логики есть алгебра векторной логики, в которой заданы операции: and, or, not. Векторизация есть конкатенация переменных. Девекторизация есть автоматная, в общем случае, процедура формирования двоичного решения на основе определенных двоичных значений вектора (векторов) существенных переменных. Процесс-модель девекторизации логических условий, подготовленных для принятия решения, имеет две альтернативные структуры. Первая – комбинационная, формирует мгновенно двоичное решение, например, путем наличия единичных значений на n входах элемента and. Вторая – последовательностная или автоматная, – накапливает совокупность единичных условий n переменных во времени с помощью, например, счетной структуры. Возможна комбинация в виде параллельно-последовательной или последовательно-параллельной схемы для принятия решения. В общем случае, девекторизатор, как цифровой формирователь решения, может быть представлен в виде автоматной модели первого рода:

$$Y(t) = f[m(t), Z(t-1)];$$
$$Z(t) = g[m(t), Z(t-1)].$$

Здесь Y, Z, m – переменные выхода, состояния и входов автомата девекторизации логических значений входного вектора.

Автомат вырождается в тривиальный или комбинационный, если значение выхода девекторизатора зависит только от входного вектора:

$$Y(t) = f[m(t)] \approx Y = f(m_1, m_2, ..., m_i, ..., m_n).$$

Девекторизатор комбинационного типа может быть простым или сложным. Простой реализован вектор-бит операцией and, or, and-not, or-not:

$$Y = \Delta(m) = (m_1 \Delta m_2 \Delta ... \Delta m_i \Delta ... \Delta m_n),\ \Delta = \{and, or, and - not, or - not\}.$$

Сложные конструкции, использующие логическое взаимодействие двух векторов, представлены выражениями:

$$Y = \{(a \wedge b);\ (a \vee b);\ a \wedge \overline{(a \wedge b)};\ b \wedge \overline{(a \wedge b)};\ (a \vee b) \wedge \overline{(a \wedge b)}\}.$$

Каждому варианту взаимодействия можно поставить в соответствие теоретико-множественную диаграмму, а также логическую схему, которая соответствует процесс-модели принятия решения, рис. 1.

Как правило, процедуры принятия критических и ответственных решений определены в пространстве и во времени, что исключает ошибку, но при этом затягивается процесс формирования вывода. Для принятия решения конструктивный мозг синтезирует только минимальное число существенных переменных, исключая несущественные по мере приобретения опыта. Решение всегда двоичное, следовательно, переменные ассоциативного вектора, даже лингвистические, участвующие в формировании вывода, должны быть приведены к двоичной норме.

На рынке электронных технологий наиболее распространенными являются две альтернативные модели данных: явная табличная и неявная аналитическая. Таблица истинности есть совокупность векторов, задающих поведение дискретного объекта $Y = f(X)$ в многозначном (двоичном) алфавите с выраженным отношением координат входных и выходных переменных:

$$C = (C_1, C_2, ..., C_i, ..., C_n), C_i = (X_i, Y_i), C_{ij} \in \{\alpha_1, \alpha_2, ..., \alpha_r, ..., \alpha_k\}$$

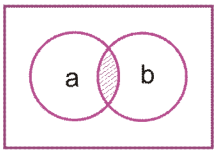
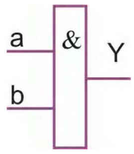
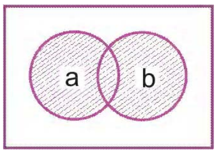
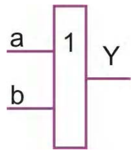
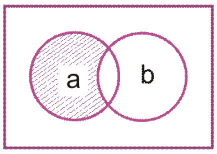
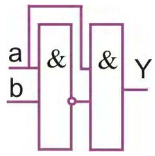
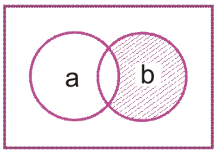
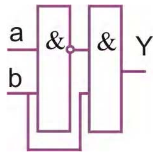
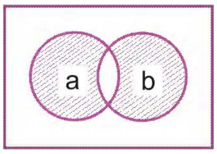
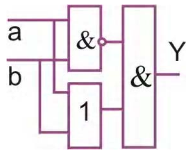
$Y = (a \wedge b)$		
$Y = (a \vee b)$		
$Y = a \wedge \overline{(a \wedge b)}$		
$Y = b \wedge \overline{(a \wedge b)}$;		
$Y = (a \vee b) \wedge \overline{(a \wedge b)}$;		

Рис. 1. Примитивные процесс-модели принятия решений

Двоичный алфавит $X_{ij} \in \{\alpha_1, \alpha_2, ..., \alpha_r, ..., \alpha_k\}, Y_{ij} \in \{0,1\}$ определения выходных координат формирует алгебру логики, где каждое высказывание принимает значение истинности или ложности: $f(X_i) = Y_i = \{1,0\}$. Тождественно истинные высказывания $f(X_i) = Y_i = \{1\}$, формируют алгебру конечных предикатов (АКП).

Таким образом, любая таблица истинности C может быть преобразована в две матрицы, нулевую и единичную относительно состояния выходной переменной, которые могут отличаться количеством строк или векторов:

$$C = \{M^1 = M^1_{ij};\ M^0 = M^0_{rj}\}, i = \overline{1, n_0}, r = \overline{1, n_1}, j = \overline{1, k-1}, n_0 + n_1 = n.$$

Здесь единичная матрица формирует структуру, которая соответствует табличному заданию конечного предиката (DNF), рис. 2:

$$P = f(X) = 1 \approx M^1 = M^1_{ij}.$$

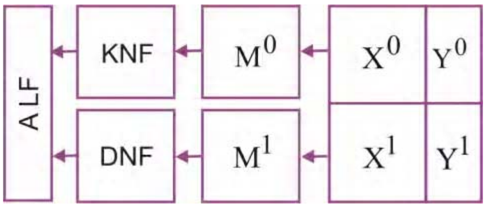


Рис. 2. Формы задания функций алгебры логики

Все строки матрицы формируют только тождественно истинные решения, заданные в явном виде. Достоинство матричного (табличного) задания предиката заключается в исключении сложных процедур анализа матрицы для формирования решения.

Вычислительные процедуры анализа таблиц (матриц) и уравнений существенно различаются по своей сложности.

1. Анализ таблицы истинности – моделирование исправного поведения. Определение решения как реакции выходной двоичной (троичной) переменной таблицы истинности на входной вектор заключается в поиске непротиворечивых результатов пересечения вектора с каждой строкой таблицы:

$$Y = Y \cup C_{ik} \leftarrow (m \cap C_i \neq \varnothing).$$

Данной модели процесса ставится в соответствие изоморфная алгебро-логическая структура, формирующая решение в двоичном алфавите:

$$Y = Y \vee C_{ik} \leftarrow (m \wedge C_i = m).$$

Учитывая, что в таблице истинности, заданной в двоичном алфавите, может существовать только одно решение, удовлетворяющее условию $m \wedge C_i = m$, то функционал вычисления состояния выхода Y, трансформируется к более простому выражению:

$$Y = C_{ik} \leftarrow (m \wedge C_i = m).$$

2. Анализ единичной матрицы M^1 – все переменные равнозначны. Формируется вектор позитивного или негативного взаимодействия входного запроса со строками матрицы. Затем определяется решение A путем объединения или группирования непротиворечивых (позитивных) результатов пересечения вектора с каждой строкой матрицы:

$$A = A \cup M_i^1 \leftarrow (m \cap M_i^1 \neq \varnothing).$$

Теоретико-множественной модели процесса ставится в соответствие изоморфная алгебро-логическая структура, формирующая решение в двоичном алфавите:

$$A = M_i^1 \leftarrow (m \wedge M_i^1 = m).$$

3. Более сложная процесс-модель связана с анализом матрицы M на основе сформированного выходного m-вектора позитивного или негативного взаимодействия запроса со строками матрицы, которая представлена следующим выражением:

$$A^s = (\bigcap_{\forall m_i = 1} M_i \setminus \bigcup_{\forall m_i = 0} M_i);$$

$$A^m = (\bigcup_{\forall m_i = 1} M_i \setminus \bigcup_{\forall m_i = 0} M_i).$$

Здесь два уравнения определяют единственное и множественное решения путем анализа единичных и нулевых строк матрицы, отмеченных позитивным или негативным состоянием координат m-вектора. Изоморфные уравнения в алгебре логики, соответствующие последним теоретико-множественным решениям, имеют вид:

$$A^s = (\bigwedge_{\forall m_i = 1} M_i) \wedge (\overline{\bigvee_{\forall m_i = 0} M_i});$$

$$A^m = (\bigvee_{\forall m_i = 1} M_i) \wedge (\overline{\bigvee_{\forall m_i = 0} M_i}).$$

4. Метод неопределенных коэффициентов [4] для минимизации функций алгебры логики также сводится к анализу матрицы входных переменных по нулевым и единичным координатам состояний вектора выходных координат, путем использования последних уравнений. Пусть имеется таблица истинности функции от трех переменных, дополненная всеми возможными сочетаниями входных переменных для минимизации булевой функции:

X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$	$m \approx F$
0	0	0	00	00	00	000	1
0	0	1	00	01	01	001	0
0	1	0	01	00	10	010	1
0	1	1	01	01	11	011	0
1	0	0	10	10	00	100	1
1	0	1	10	11	01	101	0
1	1	0	11	10	10	110	0
1	1	1	11	11	11	111	1

Применение теоретико-множественной процесс-модели

$$A^m = \Big(\bigcup_{\forall m_i=1} M_i \setminus \bigcup_{\forall m_i=0} M_i\Big) = \Big(\bigcup_{\substack{j=\overline{1,7}\\ \forall m_i=1}} M_{ij} \setminus \bigcup_{\substack{j=\overline{1,7}\\ \forall m_i=0}} M_{ij}\Big)$$

к строкам таблицы истинности дает следующий результат:

X_1	X_2	X_3	X_1X_2	X_1X_3	X_2X_3	$X_1X_2X_3$	F
.	.	.	.	00	00	000	1
.	.	.	.	00	.	010	1
.	.	.	.	.	00	100	1
.	.	.	.	.	.	111	1

Покрытие единичных строк функции конъюнктивными термами минимальной стоимости, определяющее оптимальную ДНФ, имеет следующий вид: $F = X_1X_2X_3 \vee \overline{X}_2\overline{X}_3 \vee \overline{X}_1\overline{X}_3$. Квазиоптимальное покрытие может быть составлено из левых по положению в таблице термов, принадлежащих каждой единичной строке.

3. H-метод минимизации булевых функций

Интересная процесс-модель минимизации булевой функции по избыточной таблице истинности может быть получена, если использовать диаграмму Хассе [3, 4] в качестве формы, позволяющей оптимально задавать иерархию и эволюцию любой ассоциативной логической структуры. На рис. 3 представлены три графа, где первый и второй соответствуют нулевым и единичным строкам таблицы истинности F, заданной выше, а третий является результатом вычитания $G = G_i^1 \setminus G_i^0$. Достоинства диаграммы Хассе в данном примере заключаются в эффективном формировании решения на основе теоретико-множественного вычитания содержимого вершин нулевого графа из соответствующих вершин единичного графа. После чего минимальное решение определяется такими нижними гранями (вершинами) третьей диаграммы (см. рис. 3), которые покрывают все строки или кубы самой верхней грани (Supremum G).

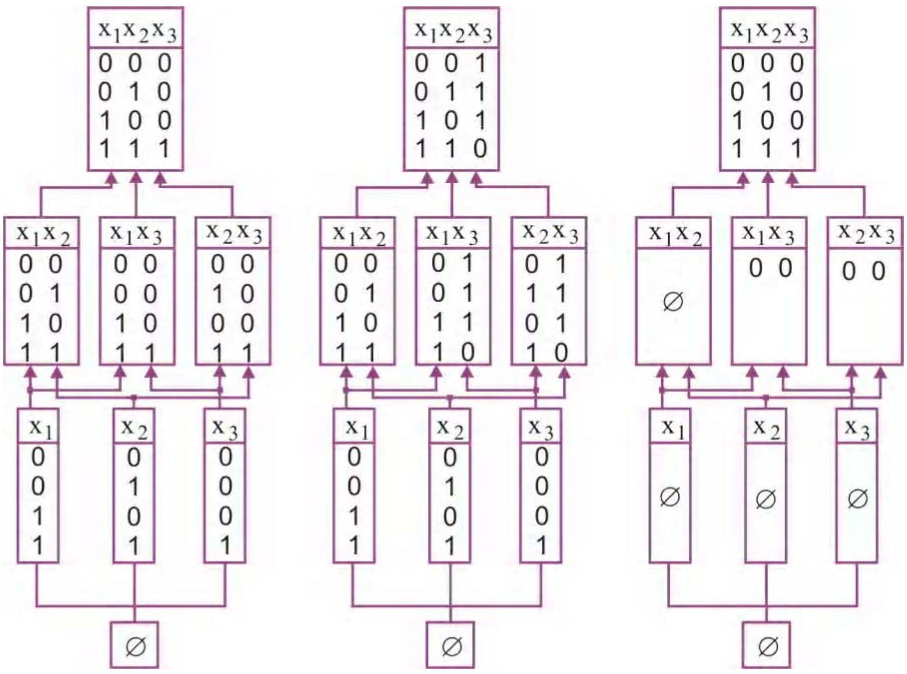


Рис. 3. Диаграммы Хассе для минимизации логической функции

Вычислительная сложность нового H-метода (Hasse) минимизации на основе использования диаграмм Хассе определяется процедурами: 1) создания двух графов G^1, G^0;

2) избирательной операцией покомпонентного теоретико-множественного вычитания содержимого вершин: $G = G_i^1 \setminus G_i^0, i = \overline{1,2^n}$, n – количество переменных булевой функции;

3) поиском квазиоптимального решения путем покрытия кубов верхней вершины наборами, принадлежащими нижним компонентам графа. Формула вычислительной сложности, включающая три слагаемых для упомянутых процедур, имеет вид:

$$Q = 4n + \frac{1}{4}n^2 + \frac{1}{4}n^2 = 4n + \frac{1}{2}n^2.$$

Стратегия получения минимальной ДНФ – за счет введения избыточности к таблице истинности G_{nj} обеспечить ее покрытие простыми кубами $G_{ij} \in G$, расположенными в нижних вершинах:

$$G = G_i^1 \setminus G_i^0;$$
$$F = F \cup G_{ij} \leftarrow G_{ij} \cap G_{nj} = G_{nj}.$$

Для получения минимальной КНФ необходимо выполнить вычитание единичного графа из нулевого, а затем покрыть нулевые кубы таблицы истинности G_{nj} более простыми кубами $G_{ij} \in G$ компонентов, расположенных как можно ниже:

$$G = G_i^0 \setminus G_i^1;$$
$$F = F \cup G_{ij} \leftarrow G_{ij} \cap G_{nj} = G_{nj}.$$

Диаграмма Хассе есть идеальная, замкнутая, иерархическая, мозгоподобная структура логических теоретико-множественных отношений в форме булеана примитивов, где решаются все задачи взаимодействия ассоциаций переменных. В целях уменьшения размерности диаграммы Хассе для ее понимания человеком необходимо выполнять разбиение числа переменных и строить двух или иерархическую многоуровневую структуру по возможности с одинаковым числом переменных или примитивов. Например, при n=16, следует построить 4 диаграммы Хассе по 4 примитива в каждой из них, которые объединяются графом такой же мощности, где примитивами выступают диаграммы нижнего уровня иерархии.

4. Выводы

Параллельные векторные мультипроцессорные неарифметические процесс-модели, представленные в исследованиях, ориентированы на новые эффективные решения практических задач синтеза и анализа: минимизация булевых функций, поиск дефектов, восстановление работоспособности, распознавание образов, принятие решений, разработка цифровых фильтров, создание дружественных серверов, сайтов и порталов.

Рыночная привлекательность исследований заключается в ориентации предложенной математической и технологической культуры на создание метрики кибернетического пространства, инфраструктуры сервисного обслуживания в виде моделей, методов и средств, включающих быстродействующие мозгоподобные компьютеры, критерии качества взаимодействия объектов в пространстве при поиске, распознавании и принятии решений, а также новые сервисы со стороны кибернетического пространства и индивидуального интеллектуального компьютера.

Список литературы: 1. *Бондаренко М.Ф.* О мозгоподобных ЭВМ / М.Ф. Бондаренко, З.В. Дударь, И.А. Ефимова, В.А. Лещинский, С.Ю. Шабанов–Кушнаренко // Радиоэлектроника и информатика. Харьков: ХНУРЭ. 2004, № 2. С. 89–105. **2.** *Cohen A.A.* Addressing architecture for Brain-like Massively Parallel Computers / Euromicro Symposium on Digital System Design (DSD'04). 2004. P. 594-597. **3.** *Кузнецов О.П.* Быстрые процессы мозга и обработка образов // Новости искусственного интеллекта. 1998. №2. **4.** *Васильев С.Н., Жерлов А.К., Федосов Е.А., Федунов Б.Е.* Интеллектуальное управление динамическими системами.– М.: Физико-математическая литература. 2000. 352 с. **5.** *Липаев В.В.* Программная инженерия. Методологические основы. Учебник. Москва: Теис. 2006. 608 с. **6.** *А.С.* №1439682. 22.07.88. Регистр сдвига / Какурин Н.Я., Хаханов В.И., Лобода В.Г., Какурина А.Н. 4с. **7.** *Гайдук С.М., Хаханов В.И., Обризан В.И., Каменюка Е.А.* Сферический мультипроцессор PRUS для решения булевых уравнений // Радиоэлектроника и информатика. Харьков. 2004. № 4(29). С.107-116. **8.** *Проектирование* и тестирование цифровых систем на кристаллах / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ, 2009. 484с. **9.** *Проектирование* и верификация цифровых систем на кристаллах. Verilog &

System Verilog / В.И. Хаханов, И.В. Хаханова, Е.И. Литвинова, О.А. Гузь. Харьков. Новое слово. 2010. 528с. **10.** *Акритас А.* Основы компьютерной алгебры с приложениями: Пер. с англ. / А. Акритас. М.: Мир. 1994. 544 с. **11.** *Аттетков А.В.* Методы оптимизации / А.В. Аттетков, С.В. Галкин, В.С. Зарубин. Москва: Издательство МГТУ им. Н.Э. Баумана. 2003. 440 с. **12.** *Abramovici M.* Digital System Testing and Testable Design / M. Abramovici, M.A. Breuer and A.D. Friedman. Comp. Sc. Press. 1998. 652 p. **13.** Densmore D. A Platform-Based taxonomy for ESL Design / Douglas Densmore, Roberto Passerone, Alberto Sangiovanni–Vincentelli // Design & Test of computers. 2006. P. 359–373. **14.** *Автоматизация* диагностирования электронных устройств/ Ю.В.Малышенко и др./ Под ред. В.П.Чипулиса. М.: Энергоатомиздат, 1986. 216с. **15.** Трахтенгерц Э.А. Компьютерные методы реализации экономических и информационных управленческих решений. СИНТЕГ. 2009. 396 с.

Поступила в редколлегию 11.09.2010

Хаханов Владимир Иванович, декан факультета КИУ ХНУРЭ, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Увлечения: баскетбол, футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326, e-mail: hahanov@kture.kharkov.ua.

Ngene Christopher Umerah, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем и сетей. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-421, e-mail: kiu@kture.kharkov.ua.

Ольховой Виталий, студент факультета КИУ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326.