МЕТОДИКА ОБРАБОТКИ ЦИФРОВЫХ МАТРИЧНЫХ СТРУКТУР

Львов С.А.
Восточноукраинский национальный университет имени В. И. Даля,
г. Луганск, Украина

Abstract
Lvov S.A. The method of digital matrixes structures processing. *In the given work the principles of the organization of parallel data processing on an example of multiplication of matrixes are considered. The synthesis complete focused the column multipate of matrixes is shown. An example of the realization based on the neuronlike elements is given.*

Одной из главных проблем современной вычислительной техники является разработка принципов построения эффективных средств обработки информации, которые позволили бы увеличить производительность вычислительных средств и уменьшить их стоимость.

Уровень производительности современных универсальных ЭВМ составляет 10^8-10^9 оп/сек, супер-ЭВМ и проблемно-ориентированных систем – до 10^{11} оп/сек, что явно недостаточно для решения сложных задач в реальном времени. Поэтому проводятся интенсивные исследования в области создания систем обработки информации принципиально нового типа, в основе функционирования которых лежат принципы обработки информации в нейронных сетях мозга. В нейронных сетях вычислительные элементы имеют множество параллельных соединений, причем каждый элемент соединен почти с каждым, поэтому входной сигнал как бы «разливается» по всей сети и все элементы сети работают параллельно, реализуя тем самым массированно-параллельные вычисления.

Для соединения процессоров в сеть существует много методов и топологий [1], таких например как : кубически связанные циклы, т.е. гиперкуб; альфа-сеть, т.е. гиперкуб размерности **R**; ячеистая сеть, т.е. систолические структуры; топология гипердерева; структура мультидерева; полностью связанная цепь и др. Каждая из вышеперечисленных топологий представляется в виде *графа вычислительных ресурсов*, в котором каждая его вершина обозначает процессор или ЭВМ сети.

Наиболее гибкой структурой является полностью связанная цепь или, другими словами, полносвязная вычислительная среда (в дальнейшем ПВС), которая позволяет реализовать любую модель нейронной сети независимо от вида представляющего ее графа и способа отображения, т.к. ПВС представляет собой пространственный или формируемый во времени полный ориентированный граф из **N** вершин, в котором каждая вершина соединена двумя противоположно-ориентированными ребрами.

Разработка и создание систем типа ПВС отпугивала разработчиков, т.к. стоимость соединений процессоров растет как $\mathbf{n^2}$, где **n** – число вершин в графе, поэтому развитие вычислительных средств пошло в другом направлении, а именно, в направлении создания проблемно-ориентированных высокопроизводительных многопроцессорных систем, насчитывающих до 32 тысяч процессорных элементов с производительностью каждого до 1000 млн. оп. в секунду.

Современный процессор содержит порядка 2 миллионов условных логических элементов. Процессор типа RISC содержит порядка 30-100 тысяч элементов, т.е. возможности одного процессора велики, но использование его в качестве вершины графа сети типа ПВС сводит возможности процессора к минимуму, т.к. вычислительная энергия системы будет максимальна, а консенсус минимален.

С другой стороны, вычислительная система, состоящая из 32 тысяч процессоров, выполнит за один параллельный цикл всего 32 тысячи умножений. Но для того, чтобы перемножить две матрицы порядка 256x256, необходимо выполнять 16777216 умножений и 16711680 сложений за один цикл. Ясно, что применить такое количество процессоров, каждый из которых содержит более 100 тысяч условных логических элементов – нереально, т.к. все процессоры необходимо запрограммировать, да и обмен между ними сложен и занимает много времени.

Поэтому предлагается в качестве вершин полного ориентированного графа использовать не процессоры, а нейроподобные элементы (в дальнейшем НП, или для краткости – нейроны). Структура нейрона значительно проще процессора, отсутствует программирование, количество условных логических элементов на несколько порядков меньше, а вычислительные возможности такие же, а могут быть и больше, т.к. нейрон занимается только вычислениями. Например, нейрон на четыре входа и один выход, в качестве носителя информации в котором используется плотность потока импульсов единичной амплитуды, содержит примерно 400 логических элементов, но выполняет за один цикл четыре умножения и три сложения одновременно, т.е. параллельно. Таким образом при использовании процессоров на одно умножение за один цикл приходится 100 тысяч и более условных логических элементов. При использовании же нейронов на одно умножение приходится всего 100 условных логических элементов, что в 1000 раз меньше, и что, в свою очередь, позволяет на одном кристалле разместить от нескольких тысяч до десятков тысяч нейронов (в зависимости от типа кристалла) [2].

Кроме того, увеличение количества умножителей в одном нейроне увеличивает количество условных логических элементов не пропорционально, а логарифмически, причем, производительность нейрона не зависит от числа умножителей в нем, т.е. от числа синапсов на его входе.

Полносвязная вычислительная среда на нейроподобные элементах позволяет проводить массировано-параллельную цифровую обработку информации при решении сложных задач в реальном времени.

Метод синтеза многосвязной нейроархитектуры ориентирован на решение задач линейной алгебры. Наиболее трудоемкими задачами (операциями) линейной алгебры являются произведение матриц и их обращение. В качестве примера рассмотрим произведение двух квадратных матриц порядка 4×4. Для этого распишем подробно произведение строк на столбцы.

$$C_{11} = (a_{11}, a_{12}, a_{13}, a_{14}) \times \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \\ b_{41} \end{bmatrix} \quad (1)$$

Из произведения двух матриц следует, что необходимо произвести сложение всех попарных произведений каждого элемента строки α_{ij} на каждый элемент столбца b_{ij}. Для проведения синтеза полного графа связей нейронов для произведения двух матриц необходимо построить схемы связей нейронов для произведения каждой строки на каждый столбец.

На рис. 1 представлены схемы связей нейронов для вычисления каждого элемента матрицы **C**, связь осуществляется согласно (1) выделенной строки матрицы **A** и выделенного столбца матрицы **B** (сами линии связей не показаны). Таких схем 16, каждая из которых отображает произведение строки на столбец и является как бы элементом матрицы графов. Так например, произведение первой строки на первый столбец с последующим суммирова-

нием дает элемент C_{11} . Произведение вектора строки на матрицу дает в результате вектор строку

$$(a_{11}, a_{12}, a_{13}, a_{14}) \times \begin{bmatrix} b_{11}, b_{12}, b_{13}, b_{14} \\ b_{21}, b_{22}, b_{23}, b_{24} \\ b_{31}, b_{32}, b_{33}, b_{34} \\ b_{41}, b_{42}, b_{43}, b_{44} \end{bmatrix} = (c_{11}, c_{12}, c_{13}, c_{14}) \quad (2)$$

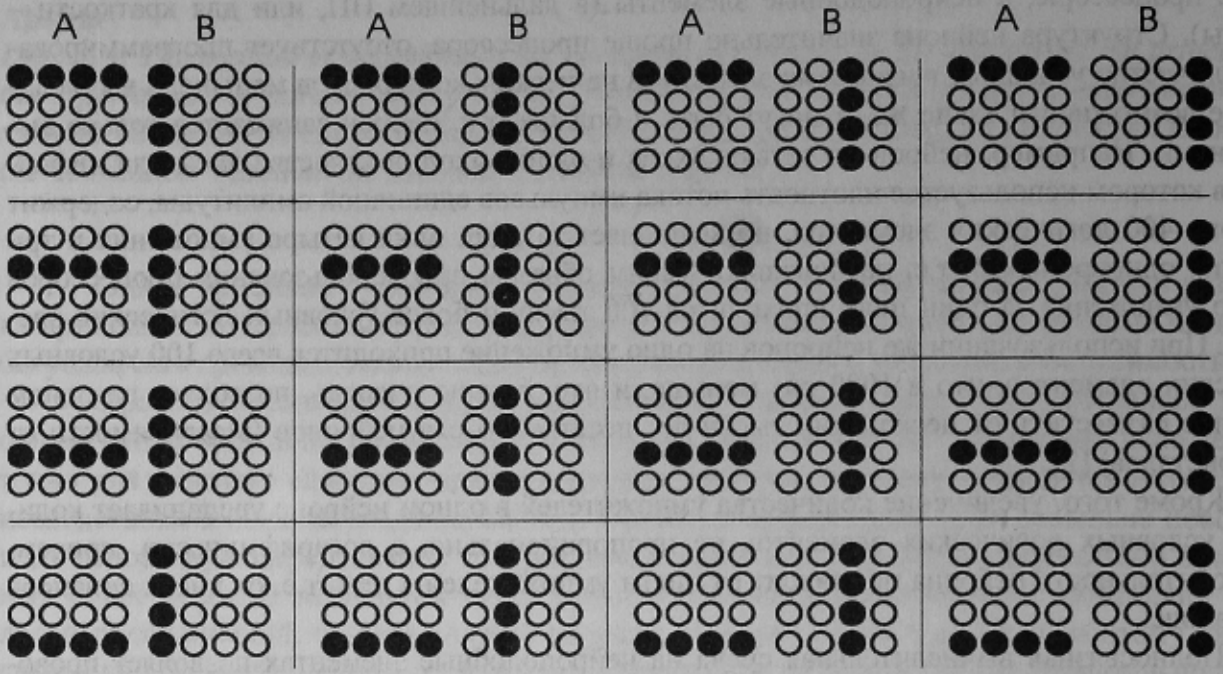

Рис.1. Синтез графа произведения двух матриц порядка 4×4.

Для получения результата согласно (1) необходимо выполнить попарные произведения и просуммировать результат этих произведений. Поэтому каждый граф произведения строки на столбец должен отображать три операции «умножения» и одну операцию «сложения», но это нецелесообразно, т.к. при выполнении параллельных умножений большая часть нейронов простаивает. Поэтому метод синтеза предусматривает оптимизацию графов произведения строк на столбцы так, чтобы не было простаивающих нейронов. Оптимизация осуществляется путем объединения взаимоперпендикулярных диагоналей симметрично-зеркальных графов (рис. 1). Объединять графы можно как относительно главной диагонали, так и относительно вспомогательной диагонали. Например, объединяя (соединяя) графы первой диагонали, параллельной вспомогательной диагонали, получим граф связей нейронов первого этапа (рис.2). Объединяя графы второй диагонали , параллельной вспомогательной, получим граф связей нейронов второго этапа и т.д. Количество этапов равно порядку матрицы. Так, например, перекрестные связи первого этапа вычисляют элементы результирующей матрицы $c_{11}, c_{24}, c_{33}, c_{42}$.

Синтезируем четыре графа (четыре этапа) вычисления попарных произведений и четыре графа вычисления сумм попарных произведений. У верхнего ряда этих графов в качестве вершин должны быть **«умножители»**, причем каждый **«умножитель»** должен иметь свое место, т.е. координаты, отображающие индексы элементов матрицы, а связи между умножителями должны отображать получение попарных произведений. При суммировании попарных произведений (нижний ряд графов) нейроны выстраиваются по связям в линию (в строку). Анализ графов показывает, что все они могут быть реализованы независимо друг от

друга и представлять собой автономные слои с простейшими нейронами в вершинах, которые выполняют только одно умножение.

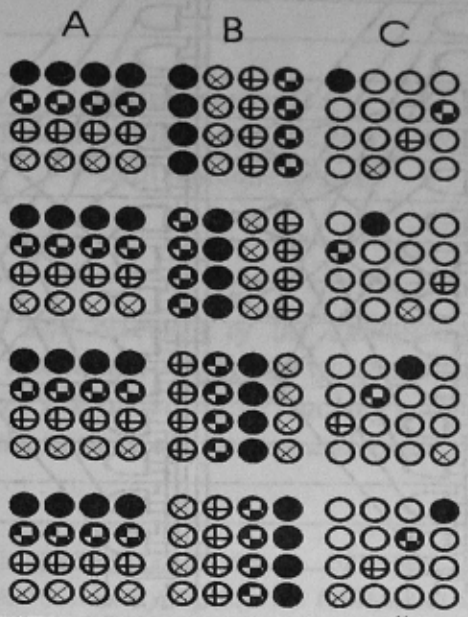

Рис. 2. Объединение взаимоперпендикулярных диагоналей симметрично-зеркальных графов (линии связей не показаны)

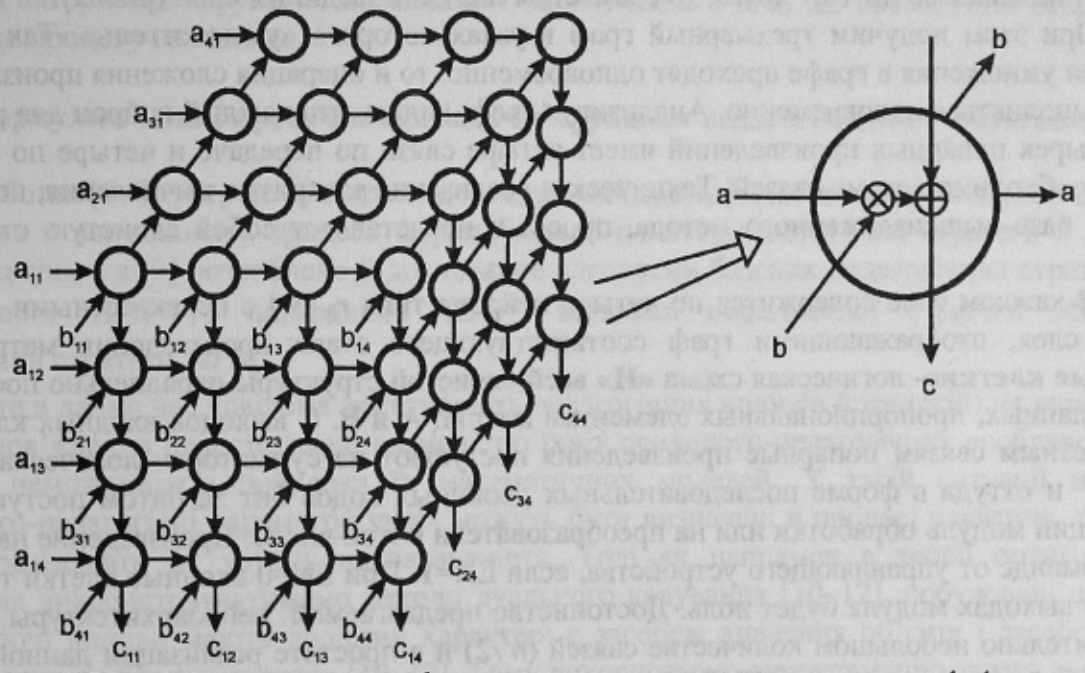

Рис. 3. Трехмерный граф произведения двух матриц порядка 4×4.

Если в качестве носителя информации в нейроне взять плотность потока импульсов единичной амплитуды, то для реализации одного попарного произведения достаточно одной «входной клетки»- логической схемы **«И»**. Для реализации же суммы попарных произведений в одной строке каждого графа достаточно одной «входной клетки»- логической схемы **«ИЛИ»** на четыре входа. Таким образом, синтезированы модели как бы двух специализированных типов простейших клеток.

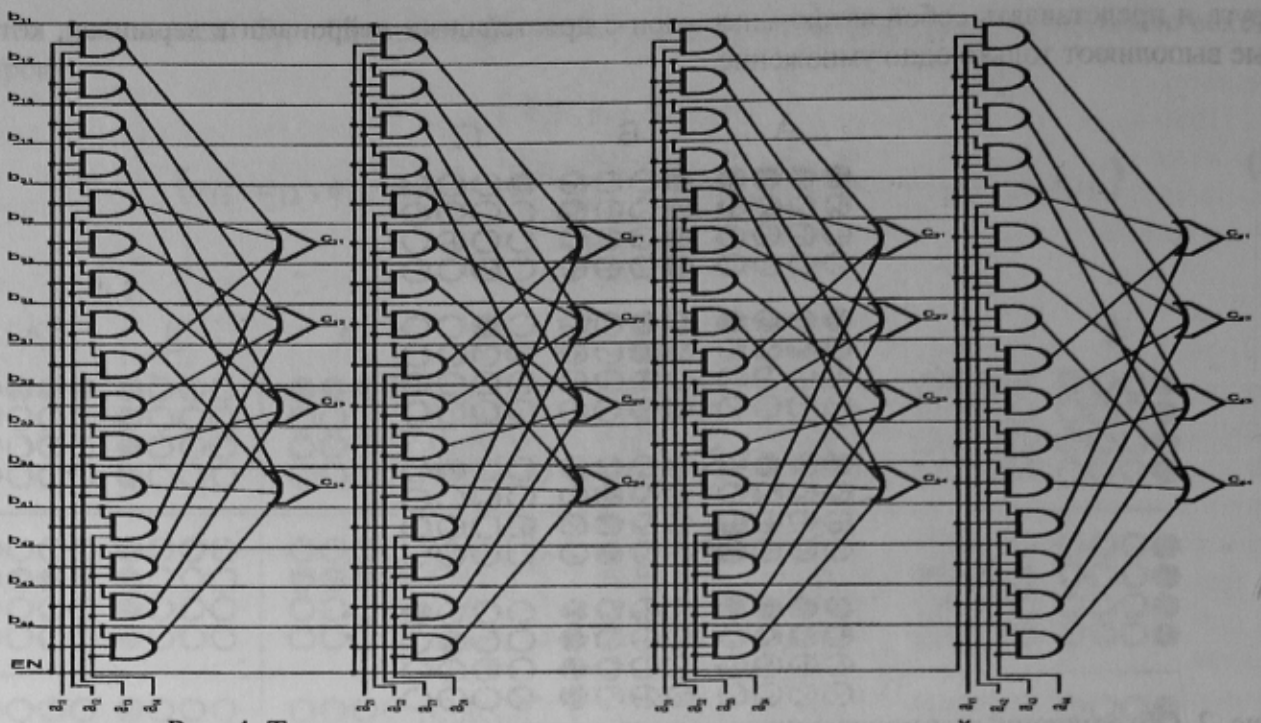

Рис. 4. Техническая реализация алгоритма перемножений матриц.

Так как четыре этапа нахождения элементов матрицы **C** могут выполняться параллельно и независимо друг от друга, то совместим их, т.е. объединим слои (рис. 3).

При этом получим трехмерный граф в узлах которого **«умножитель»**. Так как все операции умножения в графе проходят одновременно, то и операция сложения произведений будет выполняться одновременно. Анализируя граф, видим, что каждый нейрон для реализации четырех попарных произведений имеет четыре связи по передаче и четыре по приему. Итого необходимо восемь связей. Техническая реализация алгоритма вычисления, построенного на базе вышеизложенного метода, проста и представляет собой слоистую структуру (рис.4).

В каждом слое содержится по четыре нейрона типа r_0 [3] с перекрестными связями внутри слоя, отображающими граф соответствующего этапа произведения матриц. На **«входные клетки»**- логическая схема **«И»** всей слоистой структуры параллельно поступают потоки данных, пропорциональных элементам матриц **A** и **B**. С выходов входных клеток по перекрестным связям попарные произведения поступают на сумматоры- логическая схема **«ИЛИ»** и оттуда в форме последовательных двоичных кодов бит за битом поступают на следующий модуль обработки или на преобразователи поток-число. Произведение начинается по команде от управляющего устройства, если EN=**1**. При EN=0 входные клетки тормозятся и на выходах модуля будет ноль. Достоинство предлагаемой нейроархитектуры состоит в сравнительно небольшом количестве связей ($n^2/2$) и в простоте реализации данной структуры любой разрядности на ПЛИС. Недостатком же является отсутствие динамической перестройки.

Литература

1. Клини С.К. Представление событий в нервных сетях и конечных автоматах. Нейрокомпьютер: научно-технический журнал № 1,2 с. 45, Москва, 1993.
2. Львов С.А. Реализация потоковых вычислителей на ПЛИС. // Інформаційні технології. / Міжвід. наук.-техн. зб.- вип.30(1) 03. Дніпропетровськ. 2003. С.17-22.
3. Маматов Ю.А., Булычев С.Ф. и др. Потоковый нейрон на цифровых элементах. Нейрокомпьютер: научно-технический журнал № 3,4 с. 23, Москва, 1993.