

ОЦЕНКА РЕЗУЛЬТАТОВ МОДЕЛИРОВАНИЯ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ НА БАЗЕ НЕЙРОПРОЦЕССОРОВ

В.А.Романчук, В.Н.Ручкин

Рассмотрены вопросы разработки математического аппарата оценки результатов моделирования вычислительных систем на базе нейропроцессоров. Подробно описан процесс теоретико-множественного анализа вычислительной системы, классификации структур нейропроцессорных систем и аналитические выражения оценок результатов моделирования нейропроцессорных систем. Исследование выполнено при финансовой поддержке РФФИ в рамках проекта № НК-12-07-97516/13.

Ключевые слова: моделирование, нейропроцессор, система.

В настоящее время для процессоров наступил так называемый "технологический предел", означающий что они достигли максимального уровня повышения быстродействия. Одним из выходов из данной ситуации является новая элементная база, например использование нейрокомпьютеров. В области нейрокомпьютеров в настоящее время ведутся разработки с использованием новых технологий, перспективными можно назвать технологии создания оптических нейрокомпьютеров, нейрокомпьютеров на пластине, молекулярных и нанонейрокомпьютеров [1].

Специфика нейропроцессорных устройств заключается в том, что это устройства, обладают специальными возможностями (функции активации, взвешенного суммирования и др.), ориентированные на эмуляцию и работу с нейронными сетями. Кроме этого они высокопараллельны, что делает их особенно эффективными для специализированной обработки информации (обработки изображений, распознавания, криптографии).

Для дальнейшего развития в области нейропроцессорных технологий существует ряд проблем, например невысокая производительность нейропроцессорных устройств в связи с небольшой частотой нейрочипов (30-150 МГц). Одним из способов решения первой проблемы является организация многопроцессорных систем. В настоящее время в области нейропроцессорных технологий ведутся исследования в части многопроцессорности, уже разработаны модули, включающие несколько процессоров с различными связями (плата ВМ1, плата МЦ4.04, плата МЦ4.13, МЦ9.01, разработанные в НТЦ "Модуль"; модули SMT302, SMT344, SMT313, SMT315, SMT316 на базе 1,2 и 4 модулей семейства TMS320C4x).

Но имеются проблемы, мешающие созданию эффективных мультимикропроцессорных структур на базе нейропроцессоров.

1. Нейропроцессоры являются пока дорогим и штучным товаром, и не каждая организация может их приобрести в нужном количестве.

2. Проектирование и анализ специализированных многопроцессор-

ных систем на базе нейропроцессоров являются очень трудоемким и сложным процессом.

Одним из способов решения перечисленных проблем является моделирование вычислительных систем различной структуры на базе нейропроцессоров. Математический аппарат моделирования таких систем подробно показан в [4,5]. Завершающим этапом моделирования является оценка его результатов.

Целью работы является разработка математического аппарата оценки результатов моделирования вычислительных систем на базе нейропроцессоров.

Пусть $A^{(j)} \in A$ - некоторый алгоритм обработки информации, который относится в классу алгоритмов $A = \langle A_1, A_2, \dots, A_n, \dots, A_N \rangle$, предназначенных для реализации на нейропроцессорной системе и представляет собой множество искусственных нейронных сетей, функционирующих в определенном порядке. Тогда, с учетом задачи исследования, объектом теоретико-множественного анализа является вычислительная система на базе нейропроцессора для реализации алгоритма $A^{(j)}$.

Алгоритм $A^{(j)}$ представляет собой кортеж операций вида $y = f(g) = f(\sum_{i=1}^n a_i x_i + a_0)$ и операций управления. Введем множество операций $O = \{O_1, O_2, \dots, O_p, \dots, O_P\}$. С учетом введенного множества операций этот алгоритм обработки информации представляет собой кортеж операций $\langle O_1, O_2, \dots, O_m, \dots, O_M \rangle$ длиной $l_j = |A^{(j)}|$; $j = \overline{1, N}$, где $O_1, O_2, \dots, O_m, \dots, O_M \in O$:

$$A^{(j)} = \langle O_1, O_2, \dots, O_m, \dots, O_M \rangle; j = \overline{1, N}. \quad (1)$$

На этапе разработки программы каждой операции ставится в соответствие микрокоманда процессора MK_p ; $\forall p = \overline{1, K_i}$:

$$\forall O_l \in O : O_m \rightarrow \{MK_p\}; \forall p = \overline{1, K_i}; \forall m = \overline{1, M}, \quad (2)$$

где $MK = \{MK_1, MK_2, \dots, MK_i, \dots, MK_I\}$ - множество микрокоманд, написанных на внутреннем языке процессора; K_i - минимальное количество микрокоманд, необходимое для реализации операции O_m ;

В зависимости от решения задачи (2) на следующем этапе необходимо каждому j -му алгоритму обработки $A^{(j)}$ поставить в соответствие некоторую программу $PR^{(j)}$, то есть определить отображение φ :

$$\varphi : A^{(j)} \rightarrow PR^{(j)}, j = \overline{1, N}. \quad (3)$$

Под программой $PR^{(j)}$ понимается кортеж микрокоманд:

$$PR^{(j)} = \langle MK_1, MK_2, \dots, MK_l, \dots, MK_L \rangle, \quad (4)$$

где $MK_1, MK_2, \dots, MK_l, \dots, MK_L \in MK$.

Далее производится разделение кода программы $PR^{(j)}$ на множество подпрограмм $\{RO_l^{(j)}\}; l = \overline{1, L}$ и загрузка подпрограмм на каждый процессорный модель (ПМ).

Математически данный этап можно описать следующим образом:

$$PR^{(j)} \rightarrow \{RO_l^{(j)}\}; l = \overline{1, L}; \forall j = \overline{1, N}. \quad (5)$$

Введем понятие структуры S_w , под которым понимается отношение параллельности выполнения подпрограмм R_{i1}, R_{i2} между i_1 -м и i_2 -м процессорными модулями $R_{i1} S_w R_{i2}; \forall R_{i1}, R_{i2} \in PR^{(j)}$. Отношение параллельности понимается как выполнение одновременно двух или несколько подпрограмм на разных процессорных модулях.

Тогда необходимо определить структуру $S_w \in S; w = \overline{1, W}$ из множества всевозможных структур $S = \{S_1, S_2, \dots, S_w, \dots, S_W\}$, позволившую j -й программе $PR^{(j)}$ (4) поставить в соответствие множество подпрограмм:

$$S_w : PR^{(j)S_w} \rightarrow \{RO_l^{(j)}\}; l = \overline{1, L}; \forall j = \overline{1, N}. \quad (6)$$

Введем понятие равенства подпрограмм обработки информации $RO_j = RO_k$, под которым будет подразумеваться равенство длин подпрограмм $/RO_j/ = /RO_k/$, под которым понимается равенство число микрокоманд подпрограмм, и совпадение подпрограмм с точностью до микрокоманды $MK_l^{(j)} = MK_l^{(k)}, \forall l = \overline{1, |RO_j|}$, т.е:

$$RO_j = RO_k = \begin{cases} /RO_j/ = /RO_k/ \\ MK_l^{(j)} = MK_l^{(k)}, \end{cases} \quad (7)$$

где $l = \overline{1, |RO_j|}$.

Рассмотрим отношение $RO_l S_w RO_k$ структуры S_w , которое указывает на то, что любые две произвольно взятые подпрограммы $/RO_j/$ и $/RO_k/$, удовлетворяющие (7), могут выполняться одновременно на разных ПМ, то есть:

$$\forall RO_l, RO_k \in PR^{(j)} : RO_l S_w RO_k. \quad (8)$$

Тогда справедливо утверждение о том, что отношение структуры обработки S_w есть отношение эквивалентности [4].

Введенное отношение структуры обработки информации позволяет разбить всю программу $PR^{(j)}$, определяемую согласно (4) на классы эквивалентности, т.е. классы неравных между собой подпрограмм согласно

(5). Число классов эквивалентности обозначим через L , а порядок класса $|a_l|, l = \overline{1, L}$. Каждый класс эквивалентности имеет своего представителя в виде подпрограммы RO_l , имеющей число микрокоманд равное порядку подпрограммы $|RO_l|$.

Таким образом, структура S_w по определению ставит в соответствие некоторой j -й программе обработки информации $PR^{(j)}$ совокупность подпрограмм RO_l , число которых равно числу классов эквивалентности L , а кратность $(RO_l)^q$ определяется порядком класса эквивалентности $|a_l|$:

$$\forall j = \overline{1, N}, PR^{(j)S_w} \rightarrow \{(RO_l)^q\}. \forall q = \overline{1, |a_l|}; \forall l = \overline{1, L}. \quad (9)$$

Если рассмотреть все возможные структуры, полученные в результате решения задачи (9), то могут получиться структуры обработки: конвейерная, векторная, конвейерно-векторная, векторно-конвейерная.

Если число классов эквивалентности $L = 1$ и порядок класса $|a_q|$. Тогда j -му алгоритму обработки информации соответствует набор $|a_q|$ совершенно одинаковых программ. Дальнейшие рассуждения зависят от построения j -го алгоритма. Если выходная информация каждой подпрограммы RO_i является входной для следующей $RO_{i+1}; i = 1, q$; входная первой программы RO_1 является входной, а последней RO_q - выходной всего устройства, тогда назначая q процессорных модулей на обработку информации, имеем структуру конвейерного типа. Если входная информация требуется одновременно для всех подпрограмм, тогда назначая q процессорных модулей на обработку информации, получаем структуру векторного типа, в которой все q ПМ будут одновременно функционировать по одной и той же подпрограмме RO_l . Число ПМ в этом случае $q = |a_q|$. Аналогичные рассуждения для случая, когда число классов эквивалентности равно $i = 1, K$, порядок каждого класса равен $|a_l| = 1, l = \overline{1, L}$. Число ПМ в этом случае $q = L$.

Если число классов эквивалентности равно $i = 1, K$ и порядок каждого класса равен $|a_l|, l = \overline{1, L}$, то имеем наиболее общий случай, характеризующий структуру типа MIMD, так как предлагает отношения не только между подпрограммами внутри класса, но и между подпрограммами различных классов.

Рассматриваемые ниже варианты структур обработки информации могут быть реализованы в зависимости от построения конкретного j -го алгоритма $A^{(j)}$, а именно от передачи данных между подпрограммами.

Объединим все смежные подпрограммы, в которых информация передается параллельно, в K множеств. Тогда $G = \{G_1, \dots, G_K\}$, $G_i = \{RO_{iM}^{(j)}, \dots, RO_{iN}^{(j)}\}, \forall i = \overline{1, K}$. Число подпрограмм в каждой группе различно для каждого множества $N_{G_i} = |\{G_i\}|, \forall i = \overline{1, K}$. Тогда, если все подпрограммы $RO_{iM}^{(j)}, \dots, RO_{iN}^{(j)}$ внутри множеств $G_i, i = \overline{1, K}$, обмениваются информацией последовательно, то структура является конвейерно-векторной и представляет собой структуру, в которой по принципу конвейерной системы обрабатываются векторные структуры. Число ПМ будет равно $q = \sum_{i=1}^K |\{G_i\}|$. Если в каждом множестве G_i число подпрограмм равно 1, т.е. $|G_1| = |G_2| = \dots = |G_K| = 1$ то имеем структуру конвейерного типа.

Объединим все смежные подпрограммы, в которых информация передается последовательно, в V множеств. Тогда $G = \{G_1, \dots, G_V\}$, $G_i = \{RO_{iM}^{(j)}, \dots, RO_{iN}^{(j)}\}, \forall i = \overline{1, V}$. Число подпрограмм в каждой группе различно для каждого множества $N_{G_i} = |\{G_i\}|, \forall i = \overline{1, V}$. Тогда, если всем подпрограммам $RO_{iM}^{(j)}, \dots, RO_{iN}^{(j)}$ внутри множества $G_i, i = \overline{1, V}$ информация необходима одновременно, то структура является векторно-конвейерной и представляет собой структуру, в которой по принципу векторной системы обрабатываются конвейерные структуры. Число ПМ будет равно $q = \sum_{i=1}^V |\{G_i\}|$. Если в каждом множестве G_i число подпрограмм равно 1, т.е. $|G_1| = |G_2| = \dots = |G_V| = 1$ то имеем структуру векторного типа.

Исходя из поставленных целей исследования, были выбраны следующие критерии [1].

1. Время проигрыша $T_n^{(j)}$ - время, равное разности времени выполнения программы на системе и времени выполнения программы на системе, состоящей из одного ПМ.

2. Время выигрыша $T_g^{(j)}$ - время, показывающее выигрыш по времени, получаемый за счет обработки параллельных потоков данных.

Вспомогательными оценками являются:

1. Время выполнения $T_o^{(j)}$ - время, в течение которого обрабатывается программа $PR^{(j)}$ на многопроцессорной системе архитектуры S_w .

2. Время простоев $T_{np}^{(j)}$ - суммарное для всех ПМ время, в течение которого ПМ простаивает в ожидании данных.

3. Время обработки $T_p^{(j)}$ - суммарное время, равное времени, в течение которого ПМ обрабатывает данные.

Рассмотрим конвейерную структуру НПС. Цикл конвейерной структуры определяется как продолжительность максимального во времени этапа обработки, т.е.:

$$T_c^{(j)} = \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}. \quad (10)$$

Первый результат обработки информации будет получен на выходе конвейера через время $t^{(j)} = T_c^{(j)} * q$. Эта величина и является временем выполнения, т.е.

$$T_o^{(j)} = \max_{l \in L} TO_l^{(j)} * q, \forall l = \overline{1, L}. \quad (11)$$

Каждый последующий результат обработки будет получаться на выходе через $T_c^{(j)}$, тогда рассчитать время получения последующих результатов обработки можно следующим образом:

$$T_o^{(j)} = \max_{l \in L} TO_l^{(j)} * q + (N - 1) * \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}, \quad (12)$$

где N - порядковый номер необходимого результата обработки.

Тогда проигрыш по времени будет равен:

$$T_n^{(j)} = q * \max_{l \in L} TO_l^{(j)} - \sum_{l=1}^q TO_l^{(j)}, \forall l = \overline{1, L}. \quad (13)$$

С другой стороны, при обработке каждого последующего слова получается выигрыш по времени относительно однопроцессорного варианта системы. Тогда время выигрыша будет равно:

$$T_s^{(j)} = \sum_{k=1}^q ((k - 1) * TO_{q-k+1}^{(j)}). \quad (14)$$

Тогда время простоев можно оценить следующим образом:

$$T_{np}^{(j)} = \sum_{k=1}^{q-1} (k * (\max_{l \in L} TO_l^{(j)} - TO_{q-k+1}^{(j)})), \forall l = \overline{1, L}. \quad (15)$$

Время обработки равно:

$$T_p^{(j)} = \sum_{k=1}^q (k * TO_{q-k+1}^{(j)}). \quad (16)$$

Рассмотрим векторную структуру НПС. Время выполнения равно:

$$T_o^{(j)} = \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}. \quad (17)$$

Время проигрыша можно оценить:

$$T_n^{(j)} = \max_{l \in L} TO_l^{(j)} - \sum_{l=1}^q TO_l^{(j)}, \forall l = \overline{1, L}. \quad (18)$$

Время выигрыша будет равно нулю, так как за время обработки те-

кущего потока не будет обработано никаких других потоков данных:

$$T_e^{(j)} = \sum_{i=1}^q TO_i^{(j)} - \sum_{i=1}^q TO_i^{(j)} = 0. \quad (19)$$

Время простоев будет равно:

$$T_{np} = q * \max_{l \in L} TO_l^{(j)} - \sum_{l=1}^q TO_l^{(j)}, \forall l = \overline{1, L}. \quad (20)$$

Время обработки равно:

$$T_p^{(j)} = \sum_{l=1}^q TO_l^{(j)}. \quad (21)$$

Рассмотрим конвейерно-векторную структуру НПС. Время цикла определяется как продолжительность максимального по времени этапа:

$$T_c^{(j)} = \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}. \quad (22)$$

Время выполнения равно:

$$T_o^{(j)} = K * \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}. \quad (23)$$

Тогда время выигрыша равно:

$$T_e^{(j)} = \sum_{l=1}^K ((K-l) * \sum_{i=lM}^{lN} TO_i^{(j)}). \quad (24)$$

Время проигрыша можно определить:

$$T_n^{(j)} = K * \max_{l \in L} TO_l^{(j)} - \sum_{l=1}^q TO_l^{(j)}, \forall l = \overline{1, L}. \quad (25)$$

Время простоев равно:

$$T_{np}^{(j)} = \sum_{l=1}^K ((K-l) * (\sum_{i=lM}^{lN} (\max_{l \in L} TO_l^{(j)} - TO_i^{(j)}))), \forall l = \overline{1, L}. \quad (26)$$

Тогда время обработки равно:

$$T_p^{(j)} = \sum_{l=1}^K ((K-l+1) * \sum_{i=lM}^{lN} TO_i^{(j)}). \quad (27)$$

Рассмотрим векторно-конвейерную структуру НПС. Цикл равен:

$$T_c^{(j)} = \max_{l \in L} TO_l^{(j)}, \forall l = \overline{1, L}. \quad (28)$$

Время выполнения программы $T_o^{(j)}$ равно:

$$T_o^{(j)} = \max_{l \in L} TO_l^{(j)} * \max_{G_i \in G} |G_i|, i = \overline{1, V}, \forall l = \overline{1, L}. \quad (29)$$

Время выигрыша:

$$T_e^{(j)} = \sum_{i=1}^K \sum_{k=iM}^{iN} (k * TO_{iN-k+1}^{(j)}) - \sum_{l=1}^q TO_l^{(j)}. \quad (30)$$

Время проигрыша:

$$T_n^{(j)} = \max_{G_i \in G} |G_i| * \max_{l \in L} TO_l^{(j)} - \sum_{l=1}^q TO_l^{(j)}, i = \overline{1, V}. \quad (31)$$

Время простоев:

$$T_{np}^{(j)} = \sum_{i=1}^K \sum_{k=iM}^{iN} ((k * (\max_{l \in L} TO_l^{(j)} - TO_{iN-k+1}^{(j)})), \forall l = \overline{1, L}. \quad (32)$$

Время обработки:

$$T_p^{(j)} = \sum_{i=1}^K \sum_{k=iM}^{iN} (k * TO_{iN-k+1}^{(j)}). \quad (33)$$

На основе полученных аналитических выражений оценки результатов моделирования был разработан модуль анализа в многофункциональном программном комплексе "Нейро-КС" (см. рисунок).

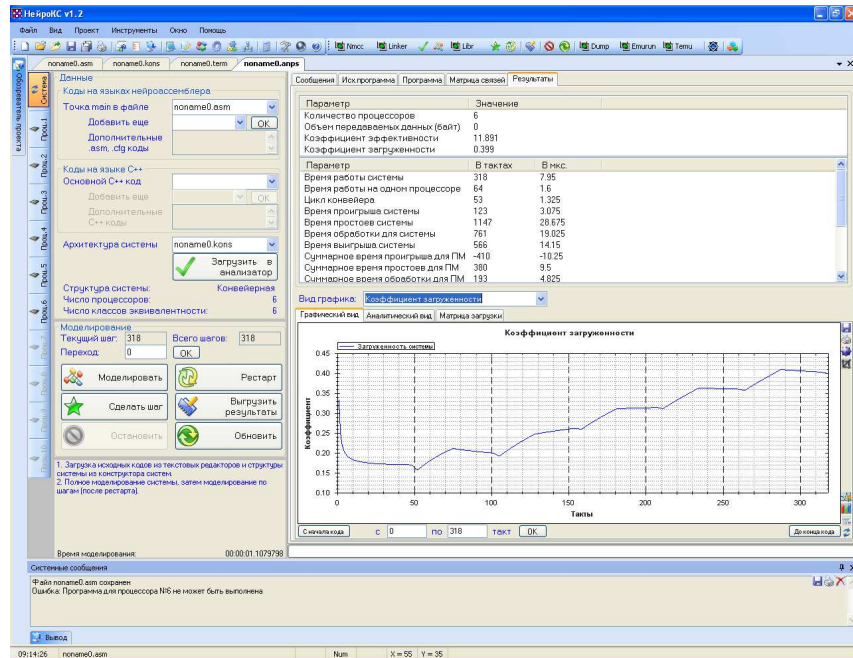


Рисунок. Модуль анализа результатов моделирования

Таким образом, цель исследования была выполнена полностью. Результатами являются:

1. Классификация многопроцессорных вычислительных систем на базе нейропроцессоров.
2. Аналитические выражения оценок результатов моделирования вычислительных систем на базе нейропроцессоров.
3. Алгоритмы и программные средства моделирования и оценок результатов моделирования.

Список литературы

1. Галушкин А.И. Нейрокомпьютеры. Кн.3. М: ИПРЖР, 2000. 528 с.

2. Злобин В.К., Ручкин В.Н. Нейросети и нейрокомпьютеры: Учеб. пособие. СПб.: БХВ-Петербург, 2011. 256 с.

3. Головкин Б.А. Вычислительные системы с большим числом процессоров. М.: Радио и связь, 1995. 320 с.

4. Романчук В.А., Ручкин В.Н. Разработка программных средств анализа нейропроцессорных систем // Вестник РГРТУ. №2. Вып.32. 2010. С.61–67.

5. Романчук В.А., Ручкин В.Н., Фулин В.А. Разработка модели сложной нейропроцессорной системы // Цифровая обработка сигналов. №4. 2012. С.70–74.

Романчук Виталий Александрович, канд. техн. наук, v.romanchuk@rsu.edu.ru, Россия, Рязань, Рязанский государственный университет имени С.А.Есенина,

Ручкин Владимир Николаевич, д-р. техн. наук, проф., v.ruchkin@rsu.edu.ru, Россия, Рязань, Рязанский государственный университет имени С.А.Есенина.

EVALUATION OF THE SIMULATION COMPUTING SYSTEMS BASED ON NEUROPROCESSOR

V.A.Romanchuk, V.N.Ruchkin

The problems of the development of the mathematical apparatus of evaluation of the results of computer modeling of systems based on neuroprocessors. Detail but describes how to set-theoretic analysis of computer systems, neuroprocessors systems classification and analytical expressions estimates of simulation results neuroprocessor systems.

Key words: modeling, neuroprocessor, system.

Romanchuk Vitaliy Aleksandrovich, candidate of technical science, v.romanchuk@rsu.edu.ru, Russia, Ryazan, Ryazan State University named by Esenin,

Ruchkin Vladimir Nikolaevich, doctor of technical science, professor, v.ruchkin@rsu.edu.ru, Russia, Ryazan, Ryazan State University named by Esenin.