ОБЪЕДИНЕННЫЙ ИНСТИТУТ
ЯДЕРНЫХ ИССЛЕДОВАНИЙ

На правах рукописи

ТИУНЧИК
Александр Александрович

УДК: 519.6 + 51:681.3.012

ИЕРАРХИЧЕСКИЙ МЕТОД ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ ФОРМ ВЫЧИСЛИТЕЛЬНЫХ АЛГОРИТМОВ ДЛЯ ОТОБРАЖЕНИЯ НА МАТРИЧНЫЕ ПРОЦЕССОРЫ С СИСТОЛИЧЕСКОЙ АРХИТЕКТУРОЙ

(01.01.07 — вычислительная математика)

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата физико-математических наук

Дубна 1995

Работа выполнена в Институте математики Академии наук Беларуси

Научный руководитель: доктор физико-математических наук Соболевский Павел Иосифович

Официальные оппоненты: доктор физико-математических наук Хоромский Борис Николаевич

кандидат физико-математических наук Ланеев Евгений Борисович

Ведущая организация: Московский инженерно-физический институт, г. Москва

Защита диссертации состоится «13» октября 1995 года в 10 час. 30 мин. на заседании Специализированного совета Д047.01.04 при лаборатории вычислительной техники и автоматизации Объединенного института ядерных исследований по адресу: 141980, г.Дубна Московской области.

С диссертацией можно ознакомиться в библиотеке ОИЯИ.

Автореферат разослан «11» сентября 1995 г.

Ученый секретарь
Специализированного совета
кандидат физико-математических наук

Иванченко
Зинаида Мироновна

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность. Быстрое развитие технологии производства вычислительных устройств обусловило создание высокоэффективных специализированных многопроцессорных систем, проектируемых с учетом особенностей реализуемых на них алгоритмов.

Появление многопроцессорных вычислительных устройств ведет к необходимости разработки специальных средств вычислительной математики, ориентированных на многопроцессорные устройства различной архитектуры: построения параллельных вычислительных алгоритмов и проектирования многопроцессорных вычислительных устройств, в максимальной степени адекватных реализуемым на них алгоритмам. Все это делает разработку и исследование параллельных вычислительных алгоритмов для реализации на многопроцессорных ЭВМ с заданой архитектурой одним из наиболее актуальных и перспективных направлений вычислительной математики.

Среди множества известных в настоящее время типов архитектур многопроцессорных вычислительных устройств следует выделить архитектуры, ориентированные на реализацию на основе сверхбольших интегральных схем (СБИС). СБИС-технология является одной из наиболее высокоэффективных и перспективных технологий производства вычислительных устройств, однако она накладывает ряд ограничений на проектируемые устройства, таких как локальность соединений между процессорными элементами, регулярность, небольшое число портов ввода-вывода и др.

Требованиям СБИС-технологии в наибольшей степени отвечают матричные процессоры с архитектурой систолического типа (систолические процессоры). Под систолическими процессорами понимают специализированные многопроцессорные устройства, характеризующиеся регулярностью внутренней структуры, пространственной локальностью связей, временной локальностью, наличием небольшого числа типов процессорных элементов, расположением портов ввода-вывода только в граничных процессорных элементах, ритмичностью обработки и распространения данных, конвейерностью и параллельностью вычислений.

Целью диссертационной работы является создание единой формальной методики построения специальных классов параллельных форм вычислительных алгоритмов и их отображения на многопроцессорные вычислительные устройства, ориентированные на СБИС-технологию.

Научная новизна. Предлагаемые в диссертационной работе методы являются новыми и обобщают известные подходы к построению некоторых параллельных форм вычислительных алгоритмов для отображения на отдельные типы систолических процессоров. Результаты работы имеют теоретический характер, сформулированы в виде утверждений, теорем и описаний конкретных алгоритмов и методов. Достоверность научных результатов обеспечена полными и строгими доказательствами. Применение разработанных методов иллюстрируется на примерах.

Практическая ценность. Разработанные методы могут быть использованы при проектировании специализированных матричных процессоров и при решении задач отображения параллельных алгоритмов на многопроцессорные устройства. Использование разработанных методов построения и отображения параллельных форм вычислительных алгоритмов на систолические структуры позволило спроектировать высокопроизводительные устройства, защищенные авторскими свидетельствами, для решения различных актуальных задач линейной алгебры, вычислительной математики и обработки сигналов.

Апробация работы. Основные результаты работы докладывались на международных конференциях "Parallel Computing Technologies-91", г. Новосибирск, 1991 г., "Parallel Computing Technologies-93", г. Обнинск, 1993 г., "Real Time Data Distributed Front-End Processing", г. Дубна, 1991 г., "Parallel Processing and Applied Mathematics", г. Ченстохова, Польша, 1994 г., на I Всесоюзной конференции "Однородные вычислительные среды и систолические структуры", г. Львов, 1990 г., VI конференции математиков Беларуси (г. Гродно, 1992 г.), заседаниях отдела математических проблем автоматизации проектирования Института математики АН Беларуси (1989–1995 гг.) и лаборатории вычислительной техники и автоматизации ОИЯИ (1995 г.).

Публикации. По теме диссертации опубликована 21 работа, список которых помещен в конце автореферата.

Объем и структура диссертации. Диссертация состоит из введения, трех глав, разбитых на параграфы, и списка литературы из 170 наименований. Объем основного текста диссертации – 159 страниц, включая 27 иллюстраций.

СОДЕРЖАНИЕ РАБОТЫ

Во введении сформулированы цели работы, дан обзор литературы по рассматриваемой теме, раскрыто содержание и приведены основные результаты диссертации.

В первой главе разработана теория построения специализированных алгоритмов, приспособленных для отображения на матричные процессоры с систолической архитектурой.

В параграфе 1.1 приведено описание специальных параллельных форм вычислительных алгоритмов, приспособленных для отображения на СБИС-процессоры, а также даны основные определения и обозначения.

Решетчатым алгоритмом будем называть алгоритм, представленный в следующей форме:

$$x_i(v)[\uparrow \vartheta_i^q] = f_i^q(x_1(v-\varphi_1^q), x_2(v-\varphi_2^q), \ldots, x_p(v-\varphi_p^q)),$$
$$1 \le i \le p,\ 1 \le q \le Q,\ v \in V^q \subset Z^m, \qquad (1)$$

где f_i^q – функция p переменных $x_1, x_2, \ldots, x_p$; $\varphi_1^q, \varphi_2^q, \ldots, \varphi_p^q$, $\vartheta_1^q, \vartheta_2^q, \ldots, \vartheta_p^q$ – векторы из пространства Z^m, определяющие координаты точек $(v-\varphi_i^q)$ и $(v+\vartheta_i^q)$ в которых, соответственно, вычислена и используется в качестве аргумента функции величина x_i. Область $V = \cup_q V^q$ будем называть областью вычислений. Вектор, соединяющий точки области вычислений и соответствующий пересылке данного x_i между этими точками, будем называть вектором зависимости, а вектор, соответствующий вводу или выводу этого данного – внешним вектором. Совокупность всех векторов обозначим через E.

Граф $G = (V, E)$ будем называть графом зависимостей, представляющим решетчатый алгоритм (1), если его вершины идентифицируются с точками $v \in V$, а множество дуг характеризуется векторами из множества E. Граф зависимостей можно рассматривать как геометрический объект, отождествляя при этом его вершины с точками пространства Z^m, в которых они расположены, а дуги зависимости – с соответствующими векторами, что дает возможность применять операторы аффинного отображения для преобразования графов зависимостей.

Между решетчатыми алгоритмами и представляющими их графами зависимостей существует взаимно-однозначное соответствие, что позволяет отождествлять задачу построения алгоритмов и задачу построения графов зависимостей, представляющих эти алгоритмы.

Пусть задан некоторый алгоритм и $G = (V, E)$ — его граф. Предположим, что множество вершин V разбито на такие непересекающиеся подмножества $V_1, V_2, \ldots, V_H$, что если $v \in V_i$, $u \in V_j$ и существует дуга (v, u), то $i < j$. Тогда разбиение $V = \cup_{k=1}^{H} V_k$ называется параллельной формой алгоритма, подмножества $V_1, V_2, \ldots, V_H$ — ее ярусами, а число H — высотой параллельной формы.

Граф зависимостей G будем называть строго направленным, если существует такой вектор $n = (n_1, n_2, \ldots, n_m) \in \mathrm{Z}^m$, который образует острые углы со всеми векторами φ_i^q, $1 \le q \le Q$, $1 \le i \le p$. Множество всех таких векторов для графа G обозначим через $\mathrm{K}(G)$.

Граф зависимостей будем называть квазисистолическим, если он является строго направленным, а вершины ввода-вывода расположены на границе области вычислений. Решетчатый алгоритм будем называть квазисистолическим, если он может быть представлен квазисистолическим графом зависимостей.

Квазисистолический граф зависимостей G будем называть систолическим, если все его векторы зависимости локальны, а число типов вершин этого графа невелико. Решетчатый алгоритм, представляемый систолическим графом зависимостей, будем называть систолическим алгоритмом.

Функцию вида

$$t(v) = \langle n, v \rangle - \min_{v \in V} \langle n, v \rangle + t_0,$$

где $\langle \cdot, \cdot \rangle$ означает скалярное произведение, t_0 — некоторая неотрицательная целочисленная константа, будем называть таймирующей функцией.

Таймирующая функция при фиксированном $n \in \mathrm{K}(G)$ задает одну из параллельных форм алгоритма, представленного квазисистолическим или систолическим графом зависимостей $G = (V, E)$, и определяет режим работы систолического процессора.

В параграфе 1.2 изложена единая методика построения специальных параллельных форм алгоритмов, реализуемых проектируемыми систолическими спецпроцессорами. С этой целью введено разделение вычислительных алгоритмов на иерархические уровни: уровень разрядов (реализация арифметической операции над числами как множества логических операций над их разрядами), уровень чисел (реализация одной операции над массивами чисел как множества арифметических операций над числами), уровень массивов (реализация множества операций над массивами чисел как совокупность отдельных операций уровня чисел), уровень

блоков (реализация всего исходного алгоритма как совокупности преобразований над промежуточными результатами отдельных блоков алгоритма).

По степени детализации алгоритмы этих уровней могут быть иерархически разделены на локальные и глобальные: алгоритм локального уровня описывает выполнение операции, которая используется при реализации макрооперации, описываемой, в свою очередь, алгоритмом глобального уровня. Так алгоритм уровня чисел является глобальным по отношению к алгоритму уровня разрядов и локальным по отношению к алгоритму уровня массивов. Алгоритмы уровня блоков являются глобальными по отношению к алгоритму любого другого более низкого уровня. Обозначим графы зависимостей алгоритмов локального и глобального уровней как G^{loc} и G^{gl} соответственно.

Пусть графы зависимостей G^{gl} и G^{loc} вложены в пространства Z^D и Z^d соответственно, а их вершины определяются наборами координат $v^{gl} = (i_1^{gl}, \ldots, i_D^{gl})$ и $v^{loc} = (i_1^{loc}, \ldots, i_d^{loc})$ соответственно. Каждой вершине v^{gl} графа G^{gl} можно поставить в соответствие граф G^{loc}, представляющий реализуемую на локальном уровне операцию, приписанную вершине v^{gl}. Такой граф будем обозначать (v^{gl}, G^{loc}).

Суперпозиция $G^{sup} = (G^{gl}, G^{loc})$ представляет собой граф зависимостей, полученный заменой вершин v^{gl} на графы (v^{gl}, G^{loc}), размещенные в пространстве Z^{D+d} и реализующие приписанные этим вершинам операции, с последующим соединением информационно зависимых вершин дугами. Алгоритм, представленный суперпозицией графов зависимостей локального и глобального уровней, будем называть двухуровневым алгоритмом, а суперпозицию — двухуровневым графом зависимостей.

Пусть $(v_\xi^{gl}, G_\xi^{loc})$ — локальные графы, реализованные в пространстве Z^{D+d} и соответствующие вершинам v_ξ^{gl}. И пусть глобальный граф G^{gl} представляет собой совокупность s информационно связанных вершин v_ξ^{gl}, $1 \le \xi \le s$. Тогда суперпозицию графов G^{gl} и G^{loc} будем обозначать также $G^{sup} = \Xi_{\xi=1}^{s} (v_\xi^{gl}, G_\xi^{loc})$.

Теорема 1.1. Суперпозиция $G^{sup} = (G^{gl}, G^{loc}) = \Xi_{\xi=1}^{s} (v_\xi^{gl}, G_\xi^{loc})$ квазисистолических графов зависимостей глобального и локального уровней представляет квазисистолический алгоритм, если

$$\cap_{\xi=1}^{s} \mathrm{K}(G_\xi^{loc}) \neq \emptyset.$$

Многоуровневые алгоритмы, получаемые на основе суперпозиции графов зависимостей, обеспечивают построение систолических алгоритмов для реализации на спецпроцессорах:

задача построения систолических алгоритмов для реализации на процессорах с поразрядной обработкой информации сводится к задаче построения многоуровневых систолических графов зависимостей на основе суперпозиций, в которых в качестве графов зависимостей локального уровня используются графы систолических алгоритмов уровня разрядов;

задача построения систолических алгоритмов для реализации на систолических процессорах для итерационных алгоритмов сводится к задаче построения суперпозиций графов зависимостей, в качестве глобальных графов в которых используются систолические графы алгоритмов уровня массивов;

задача построения систолических алгоритмов для отображения на процессоры фиксированного размера сводится к задаче построения систолических алгоритмов на основе суперпозиций, в которых в качестве графов зависимостей глобального уровня используются систолические графы уровня блоков.

Построенные суперпозиции могут быть использованы в качестве новых локальных или глобальных графов зависимостей для построения трех- или четырехуровневых систолических графов и отображения их на процессоры (например, систолический процессор с поразрядной обработкой информации для итерационных алгоритмов, систолический процессор фиксированного размера для итерационных алгоритмов, систолический процессор с поразрядной обработкой информации фиксированного размера и т.д.).

Следующие три параграфа (§1.3 – §1.5) посвящены разработке методов преобразования квазисистолических графов зависимостей, получаемых в результате построения суперпозиции, в систолические. Такие преобразования должны обеспечивать равенство и локальность векторов зависимости, соединяющих граничные информационно связанные вершины соседних локальных графов.

Параграф 1.3 посвящен получению многоуровневых систолических алгоритмов на основе последовательных трансформаций графов зависимостей локального уровня аффинными преобразованиями, обеспечивающими изменение взаимного пространственного расположения соответствующих друг другу входных и выходных вершин.

Определим преобразование каждой вершины $v = (v_\xi^{gl}, v^{loc})$ суперпозиции $\overset{s}{\underset{\xi=1}{\Xi}}(v_\xi^{gl}, G_\xi^{loc})$ оператором $\Upsilon_\xi : Z^{D+d} \to Z^{D+d}$ аффинного преобразования

$$\Upsilon_\xi v = \Psi_\xi v + \psi_\xi,$$

где Ψ_ξ – оператор линейного преобразования, ψ_ξ – вектор параллельного переноса, имеющие вид

$$\Psi_\xi = \begin{pmatrix} E & O_{D\times d} \\ O_{d\times D} & \hat{\Psi}_\xi \end{pmatrix}, \quad \psi_\xi = (O_D,\ \hat{\psi}_\xi),$$

E – матрица тождественного преобразования размерности $D \times D$, $O_{D\times d}$ и $O_{d\times D}$ – нулевые матрицы размерности $D \times d$ и $d \times D$ соответственно, $\hat{\Psi}_\xi$ – матрица размерности $d \times d$, O_D – D-мерный нулевой, а $\hat{\psi}_\xi$ – d-мерный векторы. Обозначим через $\{\Upsilon_\xi\}_{\xi=1}^s$ множество таких аффинных преобразований $\Upsilon_\xi x = \Psi_\xi x + \psi_\xi$, $1 \le \xi \le s$, что $\Psi_1 x = Ex$, $\psi_1 = 0$, а через $\overset{s}{\underset{\xi=1}{\Xi}} \Upsilon_\xi(v_\xi^{gl}, G_\xi^{loc})$ обозначим граф зависимостей, полученный в результате отображения суперпозиции $\overset{s}{\underset{\xi=1}{\Xi}}(v_\xi^{gl}, G_\xi^{loc})$ множеством операторов $\{\Upsilon_\xi\}_{\xi=1}^s$.

Теорема 1.2. Граф зависимостей $\overset{s}{\underset{\xi=1}{\Xi}} \Upsilon_\xi(v_\xi^{gl}, G_\xi^{loc})$ представляет тот же алгоритм, что и суперпозиция $\overset{s}{\underset{\xi=1}{\Xi}}(v_\xi^{gl}, \hat{\Upsilon}_\xi G_\xi^{loc})$.

В случае реализации глобального графа в одномерном пространстве для каждой пары соседних вершин v_{i-1}^{gl} и v_i^{gl} можно определить оператор $\Upsilon_{(\xi)} : Z^{D+d} \to Z^{D+d}$, $\Upsilon_{(\xi)} = \Psi_{(\xi)} + \psi_{(\xi)}$, а преобразование графа $(v_\xi^{gl}, G_\xi^{loc})$ задать оператором $\Upsilon_\xi = \Upsilon_{(1)}\Upsilon_{(2)}\Upsilon_{(3)}\cdots\Upsilon_{(\xi-1)}\Upsilon_{(\xi)}$. Обозначим через $(V_\alpha^{out})_{\xi-1}$ и $(V_\alpha^{in})_\xi$ информационно связанные соседние области вывода и ввода в графах $(v_{\xi-1}^{gl}, G_{\xi-1}^{loc})$ и $(v_\xi^{gl}, G_\xi^{loc})$.

Теорема 1.3. Если для графа квазисистолического алгоритма $\overset{s}{\underset{\xi=1}{\Xi}}(v_\xi^{gl}, G_\xi^{loc})$, $v_\xi^{gl} \in V^{gl} \subset Z^1$, существует такое множество аффинных преобразований $\{\Upsilon_{(\xi)}\}_{\xi=1}^s$, что выполняются следующие условия:

$$\hat{\Psi}_{(\xi)}(V_\alpha^{in})_\xi + \hat{\psi}_{(\xi)} = (V_\alpha^{out})_{\xi-1}, \quad 2 \le \xi \le s,$$

$$\forall \varphi \in G_\xi^{loc}: \ \hat{\Psi}_{(\xi)}\varphi \in \{\bar{\varphi}\},$$

$$\exists \lambda,\ \lambda \in N: \ \hat{\Psi}_{(\xi)}^\lambda = E,$$

$$\cap_{\xi=1}^{s}\mathrm{K}(\hat{\Psi}_{(1)}\hat{\Psi}_{(2)}\cdots\hat{\Psi}_{(\xi-1)}\hat{\Psi}_{(\xi)}G_{\xi}^{loc})\neq\emptyset,$$

то множество $\{\Upsilon_{(\xi)}\}_{\xi=1}^{s}$ обеспечивает отображение графа квазисистолического алгоритма $\overset{s}{\underset{\xi=1}{\Xi}}(v_{\xi}^{gl},G_{\xi}^{loc})$ в граф систолического алгоритма $\overset{s}{\underset{\xi=1}{\Xi}}(v_{\xi}^{gl},\hat{\Upsilon}_{\xi}G_{\xi}^{loc})$.

В параграфе 1.4 рассмотрены вопросы преобразования графов зависимостей алгоритмов локального уровня с целью получения линейной зависимости между координатами вершин вывода данных из одного локального графа и координатами вершин ввода этих данных в соседний информационно связанный с ним локальный граф.

Такое преобразование осуществляется за счет расширения графа зависимостей, т.е. использования дополнительных вершин, в которых не производятся (или производятся фиктивные) преобразования над входными данными. Дополнительные вершины соединены между собой необходимыми дугами. Расширение графа зависимостей приводит к изменению областей ввода-вывода данных.

Модификация локальных графов, обеспечивающая уменьшение длины критического пути в суперпозиции графов зависимостей, что приводит к уменьшению высоты параллельной формы алгоритма, рассмотрена в параграфе 1.5.

Основная идея такой модификации заключается в том, чтобы разместить вершины вывода данного x из графа G_{ξ}^{loc} в тех же точках, где размещены вершины ввода, сохранив при этом исходную рассылку данного x по графу G_{ξ}^{loc}. Такая модификация всегда возможна по отношению к данному x в локальном графе G_{ξ}^{loc}, если в соответствующей вершине v_{ξ}^{gl} не производятся преобразования над этим данным.

В результате такой модификации (которую будем называть модификацией ветвлением) количество дуг для пересылки данного x в суперпозиции графов зависимостей уменьшается с $d(x)+\sum\limits_{\xi=1}^{d(x)+1} d(x_j^{\xi})$ до $\max\limits_{1\leq\xi\leq d(x)+1}(\xi-1+d(x_j^{\xi}))$, где $d(x)$ – количество дуг для пересылки данного x в глобальном графе G^{loc} между последовательно соединенными вершинами $v_1^{gl}, v_2^{gl}, \ldots, v_{d(x)+1}^{gl}$, а $d(x^{\xi})$ – количество дуг для пересылки одного данного x в локальном графе G_{ξ}^{loc}, $1\leq\xi\leq d(x)+1$.

В ряде случаев модификация ветвлением возможна и по отношению к данным, которые в вершине v_{ξ}^{gl} преобразуются.

В параграфе 1.6. разработана процедура пространственно-временного отображения суперпозиций графов зависимостей на систолические процессоры.

Пространственное отображение суперпозиции графов зависимостей на граф систолического спецпроцессора определено посредством двух линейных операторов: $\Pi^{gl}: Z^D \to Z^r$ и $\Pi^{loc}: Z^d \to Z^r$ ($r=0$, 1 или 2). Образ $\Pi^{sup}v^{sup}$ точки v^{sup} задает формула $\Pi^{sup}v^{sup} = \Pi^{gl}v^{gl} + \Pi^{loc}v^{loc}$.

Использование различных операторов отображения приводит к получению широкого класса систолических спецпроцессоров. Так если через E обозначить тождественный оператор, а через O – нулевой, то операторы $\Pi^{gl} = E$ и $\Pi^{loc} = O$ отображают суперпозицию на локально последовательное глобально параллельное устройство, а операторы $\Pi^{gl} = O$ и $\Pi^{loc} = E$ – на глобально последовательное локально параллельное. Использование других операторов позволяет получать систолические спецпроцессоры как известных, так и новых конфигураций.

Условия, которым должен отвечать направляющий вектор n суперпозиции в случае отображения ее на граф систолического процессора, получены в Теореме 1.4.

Процедура вложения графовой модели вычислительного алгоритма в пространство меньшей размерности, упрощающая отображение алгоритма на многопроцессорное устройство и обеспечивающая повышение однородности конечного вычислительного устройства, разработана в параграфе 1.7.

Процедура вложения определяется формулой $v^{\Delta} = GP^Dv^{gl} + RP^dv^{loc}$, где $\Delta = \max(D,d)$, $P^D = (p_{ij}^D)$, $P^d = (p_{ij}^d)$ – матрицы порядка $D\times\Delta$ и $d\times\Delta$ соответственно, $p_{ij}^D, p_{ij}^d = 1$ при $i=j$ и $p_{ij}^D, p_{ij}^d = 0$ при $i\neq j$, $G = \mathrm{diag}(g_i), 1\leq i\leq\Delta$, – диагональная матрица, R – ортогональная матрица.

Для вычисления матриц P^D, P^d, G и R разработаны формальные методы.

Во второй главе осуществлено построение некоторых систолических графов локального и глобального уровней, которые могут быть использованы при построении многоуровневых систолических алгоритмов. С этой целью рассмотрены вопросы построения и использования рекуррентных соотношений для получения алгоритмов специального вида, исследованы возможности построения на их основе систолических алгоритмов, а также их модификации для использования при построении многоуровневых систолических алгоритмов.

Задача построения систолического алгоритма операции умножения и сложения целых чисел, представленых в дополнительном коде, решена в параграфе 2.1. Там же рассмотрены возможности модификации графа полученного систолического алгоритма методами расширения и ветвления.

Параграф 2.2 посвящен построению графов систолических алгоритмов для операций умножения и сложения плотных матриц, перемножения плотной и нижней треугольной матриц и умножения плотной матрицы на вектор.

В параграфе 2.3 осуществлено построение систолического алгоритма для LU-разложения матрицы и модификация полученного графа методом расширения.

В **третьей главе** рассмотрено построение многоуровневых систолических алгоритмов решения различных задач вычислительной математики. Построение таких алгоритмов осуществлено на основе суперпозиции графов зависимостей алгоритмов, построенных во второй главе. Рассмотрены вопросы преобразования полученных суперпозиций, вложения их в пространство меньшей размерности и отображения на вычислительные модели систолических процессоров.

В параграфе 3.1 рассмотрена задача построения двухуровневого систолического алгоритма умножения и сложения плотных матриц, реализованного на уровне поразрядной обработки информации. Для получения этого алгоритма использована суперпозиция графов систолических алгоритмов, построенных в параграфах 2.1 и 2.2. Полученная суперпозиция реализована в пространстве Z^5. Рассмотрено вложение этой суперпозиции в пространство Z^3. Показано, что полученный в результате такого вложения двухуровневый алгоритм является систолическим. На базе этого алгоритма разработана вычислительная модель систолического процессора.

Задача построения двухуровневого систолического алгоритма нахождения наибольшего собственного значения матрицы степенным методом решена в параграфе 3.2. Суперпозиция графов зависимостей, представляющая этот алгоритм, получена с использованием графа умножения матрицы на вектор, полученного в параграфе 2.2. Методом последовательных трансформаций построенная суперпозиция преобразована в граф, который является систолическим в пространстве Z^4. В результате вложения этого графа в пространство меньшей размерности получен граф, являющийся систолическим в пространстве Z^3. На основе этого графа получена вычислительная модель систолического процессора.

Параграф 3.3 посвящен построению двухуровневых и трехуровневых систолических алгоритмов решения полной проблемы собственных значений треугольным степенным методом. Для построения суперпозиции, представляющей такой алгоритм, использованы систолические графы алгоритмов перемножения плотной и нижней треугольной матриц, а также LU-разложения (полученные в §2.2 и §2.3 соответственно). Преобразование полученного в результате построения суперпозиции квазисистолического графа зависимостей в систолический реализовано методом последовательных трансформаций с последующим вложением полученного графа в пространство Z^3. На основе этого алгоритма построены вычислительные модели двумерного и одномерного систолических процессоров. Кроме того, рассмотрена задача построения трехуровневого алгоритма для реализации итерационного процесса на систолическом устройстве с фиксированным числом процессорных элементов и построена соответствующая вычислительная модель.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ ДИССЕРТАЦИИ

Разработана единая формальная методика построения алгоритмов для отображения на систолические процессоры с поразрядной обработкой информации, систолические процессоры для реализации итерационных алгоритмов и систолические процессоры с фиксированным числом процессорных элементов.

Предложены способы отображения графовых представлений параллельных форм многоуровневых вычислительных алгоритмов на многопроцессорные вычислительные устройства.

Разработаны методы модификации систолических графов локального уровня, обеспечивающие построение многоуровневых алгоритмов, максимально адекватных архитектуре соответствующего СБИС-процессора.

Предложена процедура вложения графовой модели вычислительного алгоритма в пространство меньшей размерности, упрощающая отображение алгоритма на многопроцессорное устройство.

Построены многоуровневые систолические алгоритмы решения различных задач, представляющие модели систолических матричных процессоров, полученные на основании предложенной методики.

СПИСОК РАБОТ, ОПУБЛИКОВАННЫХ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Косьянчук В.В., Лиходед Н.А., Соболевский П.И., Тиунчик А.А. О проектировании систолических вычислителей с двухуровневой конвейеризацией. – Львов, 1989. – 69 с. – (Препринт / АН УССР. Институт прикладных проблем механики и математики; N 20-89). – С. 42–47.
2. Лиходед Н.А., Тиунчик А.А. О синтезе двухуровневых систолических вычислителей // 1-я Всесоюз. конф. "Однородные вычислительные среды и систолические структуры", 10–20 апр., 1990: Тез. докл. Т. 1. / Львов, 1990. – С. 89–95.
3. Тиунчик А.А. Систолический вычислитель с распределенной арифметикой для умножения двух матриц // 1-я Всесоюз. конф. "Однородные вычислительные среды и систолические структуры", 10–20 апр., 1990: Тез. докл. Т. 1. / Львов, 1990. – С. 139–142.
4. Лиходед Н.А., Тиунчик А.А. О проектировании двухуровневых систолических вычислителей. – Мн., 1990. – 56 с. – (Препринт / АН БССР. Институт математики; N 26(426)).
5. Likhoded N.A., Sobolevskii P.I., Tiountchik A.A. Design of systolic arrays for iterative algorithms: eigenvalue computations // Proc. Int. Conf. "Parallel Computing Technologies – 91", Novosibirsk, USSR, Sept. 7–11, 1991. / World Scientific, Singapore, New Jersey, London, Hong Kong, 1991. – P. 129–138.
6. Tiountchik A.A. Design of multi-level processor arrays // Proc. Int. Conf. "Parallel Computing Technologies – 93", Vol.3, Obninsk, Russia, Aug. 30 – Sept. 4, 1993. / Computing Center, Novosibirsk, Russia, 1993. – P. 691–696.
7. Kotov V., Aleksandrov I., Pose R., Likhoded N., Tiountchik A. On a systematic approach to the programming of the systolic structures for fast trigger pipe-processing. // Int. Conf. on Real Time Data Distributed Front-End Processing, JINR, Dubna, Russia, June 27 - July 1, 1994. / Conference Handbook, Dubna, Russia, 1994. – P. 31–32.

8. Лиходед Н.А., Соболевский П.И., Тиунчик А.А. Синтез систолических вычислителей для решения полной проблемы собственных значений треугольным степенным методом. – Мн., 1991. – 40 с. – (Препринт / АН БССР. Институт математики; N 12(462)).
9. Тиунчик А.А. Метод построения алгоритмов для систолических процессоров уровня разрядов // 6 Конф. мат. Беларуси, 29 сент.– 2 окт., 1992: Тез. докл. Ч. 2. / Гродн. гос. ун-т. – Гродно, 1992. – С. 161.
10. А.С. 1721611 (СССР) М.Кл. G 06 F 15/347. Устройство для вычисления собтвенных значений $(n\times n)$-матрицы / Якуш В.П., Лиходед Н.А., Бондаренко Д.Е., Тиунчик А.А. // Бюллетень изобретений и открытий, 1992, N 11.
11. А.С. 1721612 (СССР) М.Кл. G 06 F 15/347. Устройство для операций над матрицами / Якуш В.П., Лиходед Н.А., Тиунчик А.А., Косьянчук В.В. // Бюллетень изобретений и открытий, 1992, N 11.
12. А.С. 1737463 (СССР) М.Кл. G 06 F 15/347 Устройство для умножения матрицы на вектор / Якуш В.П., Лиходед Н.А., Косьянчук В.В., Тиунчик А.А. // Бюллетень изобретений и открытий, 1992, N 20.
13. А.С. 1730948 (СССР) М.Кл. G 06 F 15/324. Устройство для решения систем линейных алгебраических уравнений / Якуш В.П., Лиходед Н.А., Соболевский П.И., Тиунчик А.А. // Для служебного пользования, 1993.
14. А.С. 1779180 (СССР) М.Кл. G 06 F 15/347. Устройство для умножения матриц / Якуш В.П., Лиходед Н.А., Косьянчук В.В., Тиунчик А.А. // Для служебного пользования, 1993.
15. А.С. 1819018 (СССР) М.Кл. G 06 F 15/324. Устройство для решения систем линейных алгебраических уравнений / Якуш В.П., Лиходед Н.А., Соболевский П.И., Тиунчик А.А. // Для служебного пользования, 1993.
16. А.С. 1782132 (СССР) М.Кл. G 06 F 15/347. Устройство для вычисления свертки / Якуш В.П., Лиходед Н.А., Косьянчук В.В., Тиунчик А.А., Чернега П.П. // Бюллетень изобретений и открытий, 1992, N 11.

17. Лиходед Н.А., Соболевский П.И., Тиунчик А.А. Один метод синтеза систолических структур, реализующих итерационные алгоритмы // Весці АНБ. Сер.фіз.-мат. навук. – 1992, N 3–4. С. 109–113.

18. Тиунчик А.А. О проектировании двухуровневых систолических процессоров с распределенной арифметикой // Весці АНБ. Сер.фіз.-мат. навук. – 1993, N 4. С. 101–106.

19. Тиунчик А.А. О проектировании многоуровневых систолических процессоров // Доклады АНБ. – 1994. – Т. 38, N 4. С. 16–18.

20. Likhoded N., Tiountchik A. Systolic implementation of matrix sweep method // Proc. 1-st Int. Conf. on Parallel Processing and Applied Mathematics, Czestochowa, Poland, September 14-16, 1994. / Czestochowa, Poland, 1994. – P. 270–277.

21. Sobolevski P., Tiountchik A. Systolic implementation of multidimensional Discrete Fourier Transform // Proc. 1-st Int. Conf. on Parallel Processing and Applied Mathematics, Czestochowa, Poland, September 14-16, 1994. / Czestochowa, Poland, 1994. – P. 278–285.