УДК 004.414.23:004.27

Средства проектирования высокопроизводительных потоковых вычислительных систем

Д.Н. Змеев

Институт проблем проектирования в микроэлектронике РАН, zmejevdn@ippm.ru

***Аннотация* – В статье описываются средства проектирования параллельных потоковых вычислительных систем, реализующих модель вычислений с управлением потоком данных. Эти средства в виде программного комплекса проектирования потоковых систем (ПКППС) позволяют оценивать эффективность прохождения задач из различных областей знаний, определять степень масштабируемости программ на различных конфигурациях вычислительной системы, подбирать функции распределения вычислений для локализации этих вычислений как по вычислительным ядрам, так и по времени выполнения соответствующего этапа задачи.**

***Ключевые слова* – комплекс проектирования потоковых систем, эмулятор, поведенческая модель, система оценки локализации вычислений.**

I. Введение

Для решения актуальных научных и практических задач (таких как задачи ядерной физики, гидро- и газодинамики, генной инженерии, природных геофизических явлений, исследования космоса и других) требуются суперкомпьютеры с реальной производительностью 10^{12}-10^{16} операций в секунду. При этом требования к подобной вычислительной технике постоянно возрастают.

Одним из способов увеличения производительности вычислительных систем является метод распараллеливания вычислений. Данный метод включает в себя программную (распараллеливание задач) и аппаратную (увеличение количества вычислительных модулей, из которых состоит суперкомпьютер) составляющие, причем в настоящее время преимущественно они опираются на фон-неймановскую архитектуру построения суперкомпьютеров. При решении многих актуальных задач такие вычислительные системы показывают низкий коэффициент загрузки процессоров, обусловленный необходимостью синхронизации параллельных вычислительных процессов, работающих с общими данными, а также временными потерями при обмене данными между процессами.

В результате реальная производительность таких систем практически на порядок отличается от пиковой. Поэтому актуальна задача создания системы, обеспечивающей на аппаратном уровне предельное распараллеливание вычислительных процессов и высокую эффективность использования вычислительных средств. С этой целью исследуются различные модели вычислений, в том числе и модель вычислений с управлением потоком данных. Особенностью архитектур, ее реализующих, является возможность выполнения операций, для которых в данный момент готовы все данные, что обеспечивает предельную степень параллелизма для конкретной задачи.

Однако широкого распространения подобные вычислительные системы не получили, в том числе и из-за отсутствия систем автоматизации проектирования и моделирования, в полной мере обеспечивающих необходимый набор инструментов для всестороннего исследования и анализа модели вычислений, архитектур и базовых принципов работы вычислительных систем.

Программный комплекс проектирования потоковых систем (ПКППС) состоит из следующих основных частей: поведенческой блочно-регистровой модели, эмулятора параллельной потоковой вычислительной системы, работающего на кластерной системе, программной системы оценки степени локализации вычислений задач и интегрированной среды разработки параллельных программ на языках DFL и HPL. ПКППС применяется при создании и исследовании оригинальной архитектуры параллельной потоковой вычислительной системы «Буран».

II. Архитектура параллельной потоковой вычислительной системы

Архитектура параллельной потоковой вычислительной системы (ППВС) «Буран» с высокой реальной производительностью базируется на гибридной модели вычислений с управлением потоком данных. Она объединяет традиционную фон-неймановскую (программа обработки данных в исполнительном устройстве) и потоковую (организация вычислительного процесса) модели вычислений [1].

Основным элементом структуры ППВС является вычислительное ядро (ВЯ). Оно конструктивно состоит из процессора сопоставления (ПС), исполнительного устройства (ИУ), блока выработки хэш-функций и внутренних коммутаторов токенов и пакетов [2]. Основными единицами информации, циркулирующими в системе, являются токен (структура, которая содержит передаваемое данное – операнд, ключ, а также служебные атрибуты) и пакет

(структура, объединяющая готовые к обработке на исполнительных устройствах операнды). При этом по внешней коммуникационной сети передается только токен, а пакет формируется, передается и исполняется исключительно внутри вычислительного модуля, который представляет собой многоядерный кристалл. ПС обеспечивает сопоставление токенов по определенным правилам и образование пакетов. ИУ обеспечивает исполнение программы узла (узел представляет собой последовательную программу, которая обрабатывает поступившие на ее вход операнды), в результате чего происходит генерация новых токенов, которые, пройдя через блок выработки хэш-функций и сформировав номер ВЯ, поступают в коммуникационную сеть.

Определяющими критериями оценки коммуникационной сети ППВС являются пропускная способность и задержки при передаче «коротких» сообщений (токенов) с использованием средств детерминированной или адаптивной маршрутизации. Основными параметрами коммуникационной среды являются топология сети (гиперкуб, 3D-тор и ряд других), вид маршрутизации и задержки коммутационных элементов. Следует отметить, что для ППВС топология коммуникационной сети тесно связана с используемой функцией локализации вычислений [2].

Программа, написанная в парадигме потоковой модели вычислений с динамически формируемым контекстом, представляется в виде виртуального графа. Граф состоит из узлов и дуг, по которым перемещаются данные к следующему узлу. Таким образом определяется последовательность обработки данных. В узле графа выполняется программа узла, результат ее работы направляется по ветвям графа к следующим узлам.

III. Поведенческая блочно-регистровая модель системы

Программная поведенческая блочно-регистровая модель (ПБРМ) системы была разработана для анализа (преимуществ и недостатков) работы отдельных элементов и узлов ППВС, создания и отладки программ. Работа поведенческой блочно-регистровой модели основывается на принципах событийного моделирования [3]. Конфигурация системы задается связанными между собой макрообъектами, причем каждый такой макрообъект представляет собой модель, состоящую из отдельных узлов и блоков, таких как модуль ассоциативной памяти, исполнительное устройство, коммутатор, входные и выходные интерфейсы и т. п., что позволяет реализовывать и отлаживать этот объект независимо от других.

В конфигурации моделируемой системы можно задавать глобальные переменные файла конфигурации, через которые можно менять параметры конфигурации: тип коммуникационной сети, объем памяти, задержки передачи и т.п.

Объекты, получая на вход операнды, выполняют действия над этими операндами в соответствии с имеющимся для этого входа алгоритмом. После завершения выполнения обработки данных генерируется структура – «событие» и передается на выход, откуда она попадает на вход связанных с этой структурой объектов. Эта последовательность действий выполняется до тех пор, пока в системе не реализуются последние «события». Структура «события» состоит из объекта, которому оно передается, времени начала его исполнения в объекте-получателе и передаваемых данных.

Данная модель позволяет с точностью до такта пропускать задачи на ней с широким диапазоном настраиваемых параметров. На ПБРМ были исследованы задачи различных классов, и получены данные, свидетельствующие, во-первых, о работоспособности концепции параллельной потоковой вычислительной системы, а во-вторых – о высокой эффективности масштабирования программ, значительно превосходящей по этому показателю традиционные кластерные вычислительные системы. Поскольку данная модель функционирует на рабочей станции, то у нее имеются ограничения на размерность пропускаемых задач и на производительность самой модели. Кроме того, значительный эффект от масштабирования может быть получен только при моделировании десятков тысяч вычислительных ядер системы, что на ПБРМ недостижимо.

ПБРМ планируется использовать также и при реализации идеи синтеза оптимальных архитектурных решений для различных классов задач. При этом особое значение приобретают средства, позволяющие пользователю не только отлаживать программы в этой среде, но и собирать полную статистику о работе отдельных устройств и блоков (объектов), анализировать эффективность прохождения задач, настраивая разнообразный набор параметров. Изменяя конфигурацию системы, выбирая различные критерии оценки и используя регулируемые параметры, пользователь при работе на ПБРМ имеет возможность подобрать оптимальный состав вычислительного ядра вычислительной системы, топологию коммуникационной среды, определить функцию локализации вычислений и, таким образом, создать базу для синтеза нужной ему (оптимальной для решения его задачи) архитектуры.

Основные модули поведенческой блочно-регистровой модели:

1) модуль сбора статистики;
2) модуль построения графа;
3) модуль формирования функции локализации вычислений;
4) модуль настройки параметров визуализации;
5) модуль отображения динамического прохождения задачи;
6) модуль хранения трассы событий.

IV. Эмулятор ППВС на высокопроизводительном кластере

С целью подтверждения и развития полученных ранее результатов на ПБРМ на задачах большой размерности был создан эмулятор ППВС, работающий на кластерных вычислительных системах [4].

Создание эмулятора ППВС для работы на кластерных системах предоставляет возможность программистам создавать программы в парадигме «раздачи», пропускать эти программы, используя модель вычислений с управлением потоком данных, апробировать разработанные средства «аппаратного» масштабирования программ.

Создание эмулятора параллельной потоковой вычислительной системы «Буран» преследует несколько целей:

1) предоставление пользователям возможности (запрограммировав свою задачу в модели вычислений ППВС) получения целого ряда параметров исследуемой архитектуры. Используя эти параметры, можно будет в дальнейшем реализовать спецвычислитель на базе ПЛИС или заказного кристалла.

2) Предоставление возможности «рядовым» пользователям программировать в парадигме «раздачи», пропускать параллельные программы в привычной для них среде, но с использованием модели вычислений с управлением потоком данных.

3) Использование разработчиками вычислительной системы «Буран» для «отработки» различных вариантов аппаратных решений и получения статистической информации для оценки эффективности от их внедрения.

4) Использование для исследования работы ППВС «Буран» с большим количеством вычислительных ядер для широкого круга задач.

Также следует отметить, что поскольку создаваемые в нашей парадигме программы обычно обладают хорошей масштабируемостью, то, несмотря на относительно высокие накладные расходы (на сам процесс эмуляции), при распараллеливании на большое количество вычислительных ядер эмулятор может демонстрировать вполне конкурентную эффективность по сравнению с кластерными вычислительными системами, что для некоторых задач позволит снизить время их решения.

По результатам исследований, ускорение работы средств моделирования на эмуляторе составило более 3-х порядков по сравнению с моделированием работы ППВС на программной блочно-регистровой модели системы. ПБРМ решает в первую очередь задачу отработки перехода к макетированию и аппаратной реализации системы.

Ускорения работы на эмуляторе ППВС удалось добиться как за счет параллельной реализации модели системы, так и за счет упрощения модели, отказавшись от потактового моделирования. При этом эмулятор ППВС обеспечивает максимально возможную производительность, сохраняя при этом функциональность системы. Потактовое моделирование вносит синхронность в процессы, являющиеся имманентно асинхронными, что в результате препятствует распараллеливанию и значительному замедлению из-за простоев в синхронизации и ожидании обменов. В эмуляторе было решено отказаться от потактового моделирования еще и потому, что с точки зрения оценки адекватности моделирования отсутствие учета времени (тактов) не будет являться критичным. Для оценки моделирования существуют такие параметры работы, как количество выполненных узлов, количество порожденных токенов, количество токенов, переданных по сети. Подсчет необходимых показателей можно вести опционально, указывая их непосредственно перед запуском процесса моделирования программы.

Кроме того, эмулятор ППВС обладает рядом преимуществ по сравнению с поведенческой блочно-регистровой моделью с точки зрения снятия целого ряда ограничений, которые накладывались на ПБРМ.

Многие из этих ограничений технического характера, вызванные, во-первых, 32-х разрядной реализацией ПБРМ, а, во-вторых, тем, что ПБРМ создана в среде RAD Studio на языке Delphi и работает под операционной системой Windows x86 в однопроцессорном режиме, что накладывает свои ограничения на доступную физическую память. К таким ограничениям относятся максимальный объем ассоциативной памяти ключей и памяти токенов в процессоре сопоставления, количество ядер в конфигурации вычислительной системы, а также общая размерность задачи, которая может быть пропущена на модели. Часть же ограничений носит принципиальный характер, поскольку ПБРМ должна отражать только те аппаратные решения, которые могут быть реализованы на существующей элементной базе.

V. Программный комплекс оценки масштабируемости

Для оценки степени локализации вычислений конкретной задачи была разработана программная система. Оценка степени локализации включает в себя как оценку распределения вычислений задачи по пространству (по вычислительным ядрам), так и во времени (по этапам задачи). Степень локализации вычислений оказывает существенное влияние на эффективность работы параллельной потоковой вычислительной системы.

Оценка степени локализации производится на всех уровнях коммуникационной иерархии вычислительной системы, исходя из предположения, что иерархическая близость соответствует арифметической, то есть вычислительные ядра с близкими номерами близки и в коммуникационной сети. Наилучшая равномерность распределения по пространству обеспечивается в том

случае, когда каждое из вычислительных ядер принимает одинаковое число токенов. Наилучшее временное распределение достигается в том случае, когда число токенов, перемещающихся между вычислительными ядрами при смене этапов, будет минимально.

Оценка степени локализации производится исходя из структуры контекста, размерности данных задачи, а также из выбранной пользователем функции распределения, причем это делается без моделирования задачи. Оценка степени локализации при минимальных известных параметрах возможна благодаря тому, что за локализацию вычислений в ППВС «Буран» отвечает блок вычисления хэш-функций, а распределение вычислений не связано с текстом параллельной программы. Именно расчет функции распределения в программной системе для каждого токена и позволяет оценить параметры, которые влияют на степень локализации задачи. Результаты оценки предоставляются в графическом и текстовом виде, включая в себя распределение токенов по ядрам и группам ядер, а также различные количественные показатели.

Программная система позволяет программисту выбрать наиболее эффективную для своей задачи функцию распределения, уточнить ее параметры без необходимости проведения ряда длительных экспериментальных сессий моделирования.

VI. Выводы

При увеличении количества вычислительных узлов, усложнении, а тем более при использовании новых, нетрадиционных моделей вычислений, оригинальных архитектурных решений, при резком усложнении коммуникационной среды при масштабировании современных вычислительных систем, а также при наличии трудностей, связанных с параллельным программированием повышаются требования к эффективной организации систем визуализации прохождения задач и сбора статистики и созданию развитых возможностей для быстрой отладки программ.

Существующие системы визуализации параллельных вычислений, предназначенные для кластерных систем (такие как VTK [5], AVS/Express Parallel Edititon [6] и другие) поддерживают в основном визуализацию научных данных [7]. К тому же использовать эти системы визуализации, вследствие коренных различий в моделях вычислений, системе программирования и оригинальной аппаратной реализации, которая обладает своими особенностями, затруднительно.

Созданный ПКППС позволяет отлаживать сложные параллельные программы, выбирать базовые функции локализации вычислений и создавать новые, исследовать новые конструкции языка программирования, исследовать новые программно-аппаратные решения, реализуемые в архитектуре при масштабировании вычислительной системы до нескольких тысяч вычислительных ядер, с большей степенью достоверности выбирать параметры для получения оптимального архитектурного решения той или иной аппаратной реализации системы. В дальнейшем планируется развивать возможности ПКППС, в том числе и для поддержки средств визуального программирования, которые создаются в настоящее время для ППВС «Буран». Еще одним направлением развития программного комплекса является оценка возможности выполнения задач пользователя с использованием потоковой модели вычислений и определения оптимальной архитектуры (с прогнозируемыми параметрами) для конкретной задачи. Программный комплекс в будущем будет доступен пользователям посредством сети Internet и обеспечит механизм создания спецпроцессоров на базе нашей архитектуры и модели вычислений. Получившееся на выходе решение для спецпроцессоров может быть реализовано на ПЛИС или в кристалле.

Литература

[1] Стемпковский А.Л., Левченко Н.Н., Окунев А.С, Цветков В.В. Параллельная потоковая вычислительная система — дальнейшее развитие архитектуры и структурной организации вычислительной системы с автоматическим распределением ресурсов // журнал "Информационные технологии" №10, 2008, С. 2 – 7.

[2] Стемпковский А.Л., Климов А.В., Левченко Н.Н., Окунев А.С. Методы адаптации параллельной потоковой вычислительной системы под задачи отдельных классов // журнал «Информационные технологии и вычислительные системы», 2009. №3, С. 12-21.

[3] Змеев Д.Н., Левченко Н.Н., Окунев А.С., Ходош Л.С. Средства визуализации процесса прохождения задачи в программной модели ППВС // Материалы Международной научно-технической конференции «Многопроцессорные вычислительные и управляющие системы» (МВУС-2009), Таганрог, ТТИ ЮФУ. Т. 1, с. 49-52.

[4] Змеев Д.Н., Климов А.В., Левченко Н.Н., Окунев А.С., Стемпковский А.Л.. Эмуляция аппаратно-программных средств параллельной потоковой вычислительной системы «Буран» // «Информационные технологии», 2015. Т. 21, №10, С. 757-762.

[5] Marcio Dutra, Paulo Rodrigues, Gilson Giraldi, Bruno Schulze, Distributed visualization using VTK in Grid Environments, pros. conf. Seventh IEEE International Symposium on Cluster Computing and Grid (CCGrid’07), 2007.

[6] Система визуализации AVS/Express Parallel Edition. URL: http://www.avs.com/solutions/express/ (дата обращения: 30.03.2016).

[7] Джосан О.В., Попова Н.Н. «Параллельная визуализация для высокопроизводительных систем обработки данных на суперкомпьютере BlueGene/P» // Труды Третьей Всероссийской научной конференции «Методы и средства обработки информации», Москва, МАКС Пресс, 2009. С. 317-323.

Design tools of high-performance dataflow computing systems

D.N. Zmejev

Institute for Design Problems in Microelectronics of RAS, zmejevdn@ippm.ru

***Keywords* – design complex of dataflow systems, emulator, behavioral model, estimating system of computation localization.**

ABSTRACT

The article describes the software complex for dataflow systems development, presents its structure, basic features and functions; various components are compared by their functionality. The complex allows to evaluate the effectiveness of tasks execution (from various fields of knowledge), to determine the degree of scalability of the task on a variety of configurations of the computing system, to select the distribution functions for the computation localization in the computing cores and in the execution time of corresponding stage of the task. Programs for the complex are created in the paradigm of "distribution" on the parallel language of DFL.

This complex has been successfully used in the development of the parallel dataflow computing system "Buran", which implements the dataflow computing model with dynamically formed context.

REFERENCES

[1] Stempkovskij A.L., Levchenko N.N., Okunev A.S., Cvetkov V.V. Parallel Dataflow Computing System: Further Development of Architecture and Structural Organization of the Computing System with Automatic Distribution of Resources. Zhurnal «INFORMACIONNYE TEHNOLOGII». 2008, No. 10, pp. 2-7 (In Russian).

[2] Stempkovskij A.L., Klimov A.V., Levchenko N.N., Okunev A.S. Methods of Parallel Dataflow Computing System Adaptation for Problems of Individual Classes. zhurnal «Informacionnye tehnologii i vychislitel'nye sistemy», 2009. No. 3, pp. 12-21 (In Russian).

[3] Zmeev D.N., Levchenko N.N., Okunev A.S., Hodosh L.S. Visualization Tools for the Process of Task Running in the Programming Model of PDCS. Materialy Mezhdunarodnoj nauchno-tehnicheskoj konferencii «Mnogoprocessornye vychislitel'nye i upravljajushhie sistemy» (MVUS-2009), Taganrog, TTI JuFU. 2009, vol. 1, pp. 49-52 (In Russian).

[4] Zmeev D.N., Klimov A.V., Levchenko N.N., Okunev A.S., Stempkovskij A.L. Emulation on Hardware and Software of the Parallel Dataflow Computing System "Buran". «Informacionnye tehnologii», 2015, vol. 21, No. 10, pp. 757-762 (In Russian).

[5] Marcio Dutra, Paulo Rodrigues, Gilson Giraldi, Bruno Schulze, Distributed visualization using VTK in Grid Environments, pros. conf. Seventh IEEE International Symposium on Cluster Computing and Grid (CCGrid'07), 2007.

[6] Visualization system AVS/Express Parallel Edition. Available at: http://www.avs.com/solutions/express/ (accessed 30.03.2016).

[7] Dzhosan O.V., Popova N.N. Parallel visualization for high data processing systems on the supercomputer Blue Gene/P. Trudy Tret'ej Vserossijskoj nauchnoj konferencii «Metody i sredstva obrabotki informacii», Moscow, 2009, pp. 317-323 (In Russian).