

донного рельефа, включающая контур береговой линии и полученная после обработки значений изобат и промерных глубин, заданных картой (рис. 4,б).

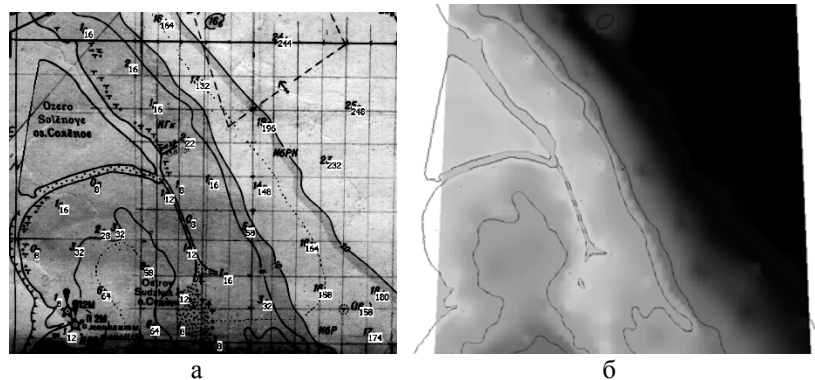


Рис. 4. Фрагмент навигационной карты и ее модельное представление, полученное в результате просчета изобат и промерных глубин

В заключении необходимо отметить, что предложенный метод восстановления рельефа донной поверхности характеризуется естественным параллелизмом, векторизуемостью и масштабируемостью основных операций: алгоритм вычислений в каждом узле растровой сетки одинаков как для каждой четверти итерации, так и для всех итераций в целом. Это позволяет выполнять решение задачи восстановления донной поверхности на параллельных процессорах с SIMD архитектурой, например, на мультипроцессорах с конфигурацией связи типа n-куб, систолическая решетка и т.п. Как показало макетирование, благодаря наличию современных ПЛИС-технологий становится возможным создание отечественных мультипроцессорных систем, обеспечивающих эффективное решение задачи восстановления донного рельефа в реальном масштабе времени.

Автор выражает благодарность ведущему инженеру-программисту кафедры ВТ Переверзеву В.А. за любезно предоставленные им программные средства собственной разработки, позволившие не только протестировать метод, но и наглядно проиллюстрировать результаты его работы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Яне Б. Цифровая обработка изображений. – М.: Техносфера, 2007.–584 с.

УДК 681.5

В.Ф. Гузик, Д.А. Беспалов, М.А. Аль-Ханани

РАЗРАБОТКА ВЫСОКОПРОИЗВОДИТЕЛЬНОЙ СИСТЕМЫ МОДЕЛИРОВАНИЯ И РЕШЕНИЯ ЗАДАЧ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ НА БАЗЕ МНОГОСЛОЙНЫХ НЕОДНОРОДНЫХ КЛАСТЕРОВ

В современной науке моделирование используется как метод познания сути физических объектов, процессов и явлений. Сложность объектов исследования неуклонно растет. Вместе с этим увеличивается и сложность самих методов и средств исследования. В связи с этим, в настоящее время продолжается и является актуальной работа по разработке и совершенствованию методов моделирования и проведения вспомогательных расчетов, связанных с обработкой специфических

данных особыми методами, выходящими за рамки обычных математических моделей объектов и систем разного рода.

Наиболее пригодным, и нередко единственным методом решения такого рода задач, является компьютерное моделирование, позволяющее изучать объект интереса без его физической реализации, исследовать его критические режимы работы и состояния, а также проводить съем данных и их последующий анализ в условиях, непригодных и небезопасных для человека-исследователя.

Постоянный рост сложности моделируемых объектов требует разработки и постоянного совершенствования соответствующих вычислительных средств, способных обеспечить расчет моделей. Кроме того, большое количество компонент моделей, их высокая размерность, а также зачастую разная физическая природа отдельных элементов моделируемых систем, делает невозможной ручную обработку данных, приводит к сложности и нецелесообразности использования единого вычислительного и аппаратного базиса для просчета всех узлов.

Все это приводит к необходимости разработки аппаратно-программного комплекса, способного решать задачи моделирования сложных процессов и систем, реализующего концепцию высокопроизводительных проблемно-ориентированных систем.

В данной статье отражена концепция системы, позволяющей проводить процесс моделирования на параллельных структурах с оптимизацией вычислительного процесса на уровне организации процесса вычислений и на уровне архитектурных и операционных особенностей аппаратного базиса с многоуровневым представлением задач, данных и вычислений.

Средства моделирования функционируют в основном в пределах одной физической дисциплины, работая с одной вычислительной и алгоритмической базой и являясь однодисциплинарными. В данном случае такая система является жестко ориентированной и направлена на решение одной, частной задачи в пределах определенного домена. Для решения задач другого плана необходима полная или частичная реконфигурация программной и/или аппаратной составляющей. Мультидисциплинарные средства моделирования в этом отношении являются более универсальными системами, так как имеют широкую область применения и решают более широкий класс задач.

Поддержка многомашинных или многопроцессорных вычислительных систем является востребованным свойством системы моделирования при моделировании сложных технических объектов и систем. Для этого в управляющей системе моделирования необходимо предусмотреть методы и средства распараллеливания задачи и контроля за исполнением вычислительных алгоритмов на отдельных узлах и для всей системы в целом.

Существует достаточно большое число вариантов аппаратных средств моделирующих систем: аналоговые, цифровые и гибридные [1]. Подавляющее большинство систем базируются на ЭВМ цифрового типа и построены по следующим принципам:

- одиночный поток команд и одиночный поток данных – SISD (однопроцессорные последовательные компьютеры с микропроцессорами общего или специализированного назначения, все команды обрабатываются последовательно друг за другом, и каждая команда инициирует одну операцию с одним потоком данных);

- одиночный поток команд и множественный поток данных – SIMD (в таких системах единое управляющее устройство контролирует множество процессорных элементов, каждый процессорный элемент получает от устройства управления в каждый фиксированный момент времени одинаковую команду и выполняет ее над своими локальными данными);

- множественный поток команд и одиночный поток данных – MISD – MIMD (подразумевает наличие в архитектуре многих процессоров, обрабатывающих один и тот же поток данных, к данному классу также относят массивы системных компьютеров и конвейерные машины);

- множественный поток команд и множественный поток данных (этот класс предполагает, что в вычислительной системе есть несколько устройств обработки команд, объединенных в единый комплекс и работающих каждое со своим потоком команд и данных, делиться на три подкласса: симметричные мультипроцессорные, массово параллельные и кластерные системы).

Последний вариант организации аппаратных средств моделирующей системы является наиболее подходящим, так как обеспечивает большую гибкость и более широкие возможности по сравнению с остальными, сохраняя баланс сложности (цены) и предоставляемых исследователю возможностей. Такая организация близка по идеологии к массово параллельным и кластерным системам, в которых вычислительные узлы объединены в некоторую коммуникативную структуру или сеть, определенной или изменяемой топологии с доступом к локальной памяти системы и коммуникационным элементам для связи с внешним миром. По такому принципу можно организовать вычислительную структуру на базе специализированных микропроцессоров, например на основе сигнальных процессоров.

Соединив универсальные средства разработки, мониторинга, управления и проведения вычислений со специализированными аппаратными элементами типа сигнальных процессоров или иных вычислительных узлов, имеющих более узкую направленность, можно получить универсальную систему, обладающую всеми достоинствами обоих подходов.

Так как аппаратный базис в таком случае представляется как совокупность вычислительных узлов разного плана, то можно говорить о гетерогенных системах, позволяющих распределить задачу по системе, выбрав наиболее подходящий аппаратный и вычислительный базис фактически для каждой процедуры обработки данных.

В таком случае можно говорить о локальном и глобальном применении параллельного подхода к решению задач и MIMD структуры на уровне данных в межпроцессорном пространстве, так и внутри каждого отдельного микропроцессора, если это позволительно с точки зрения его типа и структуры.

Объединяя вычислительные средства общего и проблемно-ориентированного назначения, можно прийти к понятию многослойной гетерогенной сети, каждый уровень и сектор которой ориентирован на свой вычислительный базис, способ организации вычислительного процесса, способ коммутации и т.п. [4]. Причем, при необходимости, используя данную концепцию можно изменять конфигурацию гетерогенной сети под конкретную задачу, добавлять или удалять отдельные сектора.

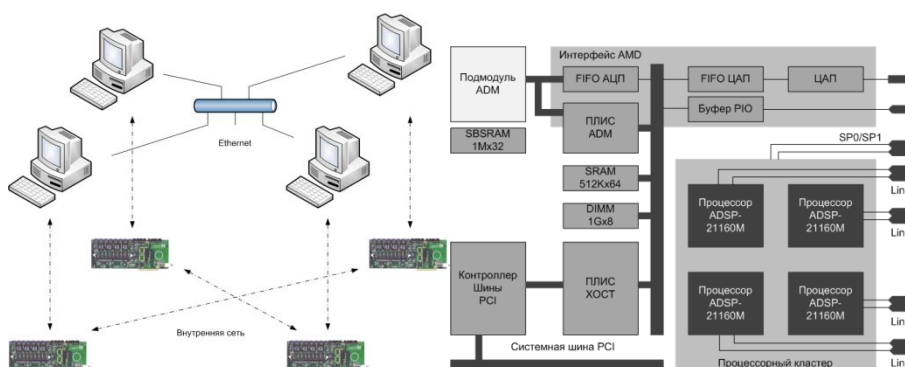


Рис. 1. Пример организации гетерогенного проблемно-ориентированного кластера: а – организация многослойной сети, б – структура отдельного ускорителя

Для апробации разрабатываемой концепции была организована гетерогенная структура, состоящая из двух слоев (подсетей). Верхний слой образовали четыре персональных ЭВМ, построенных на базе двухядерных процессоров Intel, в каждый из которых встроен аппаратный ускоритель на базе процессоров цифровой обработки сигналов. Кроме того, каждый ускоритель представляет собой мини-кластер из четырех идентичных микропроцессоров ADSP 21160 [6], соединенных по полному графу. Структура такого вычислителя представлена на рис. 1, б.

Все четыре кластера также объединены в сеть второго уровня, замыкая, таким образом, свой отдельный слой из четырех секций. Каждый процессор также является двухядерным, работающим по принципу SISD или SIMD в зависимости от поставленной задачи.

Параллельность обработки данных может быть достигнута на всех уровнях кластера: между уровнями (слоями, сетями), между отдельными секциями, внутри микропроцессоров. Сама структура и подбор аппаратного базиса такой системы подразумевает распределение программ и данных на каждом уровне сети, между вычислительными модулями нижнего уровня и в базисе отдельных микропроцессоров.

Распределение и управление загрузкой вычислительных элементов может быть как статическим, так и динамическим. Динамическое распределение предполагает балансировку загрузки вычислительных элементов непосредственно в процессе проведения моделирования. Это позволяет оптимизировать использование вычислительных ресурсов. При статическом распределении оптимизация загрузки узлов вычислителя реализуется до проведения моделирования. Очевидно, что статическое распределение не обладает гибкостью использования вычислительных ресурсов в сравнении с динамическим, однако оно позволяет избежать накладных расходов, вызванных перераспределением задач и анализом загруженности вычислительной системы, поэтому и может быть использовано в начале работы как способ для апробации идей на практике.

Получение распараллеливаемого программного кода может быть выполнено как в ручном, так и в автоматическом режиме при помощи специальных компиляторов [2]. Более удачным методом является распараллеливание на уровне реализации предварительно определенного метода решения [3]. Объектом в этом случае служат ветви или операторы алгоритма. Недостатком является необходимость формирования и предварительного расчета общей модели (математической в об-

шем случае), адаптируемой под метод, без учета сложности исходной модели. Кроме того реализация существует только для однородных систем.

Для данного случая применима идея декомпозиции модели на высоком уровне с распределением вычислительного потока для подчиненных структур нижних слоев. В рамках такой концепции именно инкапсуляция уровней и структурный подход к организации вычислительного процесса обеспечивают эффективное решение задач моделирования, снижают сложность описания на каждом уровне, а также позволяют адаптировать вычислительный процесс к аппаратному базису и структуре связей того сектора гетерогенного кластера, который решает конкретную подзадачу.

Следует также учесть, что распределение необходимо выполнять не на уровне элементарных элементов (например, уравнений), а на уровне небольших структур, соблюдая баланс. При этом следует учитывать силу связей между уровнями модели и между элементами отдельных слоев, а также трудоемкость решаемых локально задач и необходимый для их решения аппаратный базис.

С точки зрения формализации описания, такую модель можно представить в виде неориентированного графа [5] в базисе гетерогенного кластера $S = (N, L, p, c, m)$, где множество процессорных узлов (вершин графа) $N = \{n_j\}$ с соответствующими пиковыми производительностями $p(n_j)$ связано между собой множеством элементов $L = \{l_i\}$ с соответствующей пропускной способностью $c(l_i)$ в коммуникационном пространстве модели S . Управление потоками данных осуществляется частично через функцию маршрутизации $m(L, c, p)$ коммуникационного пространства.

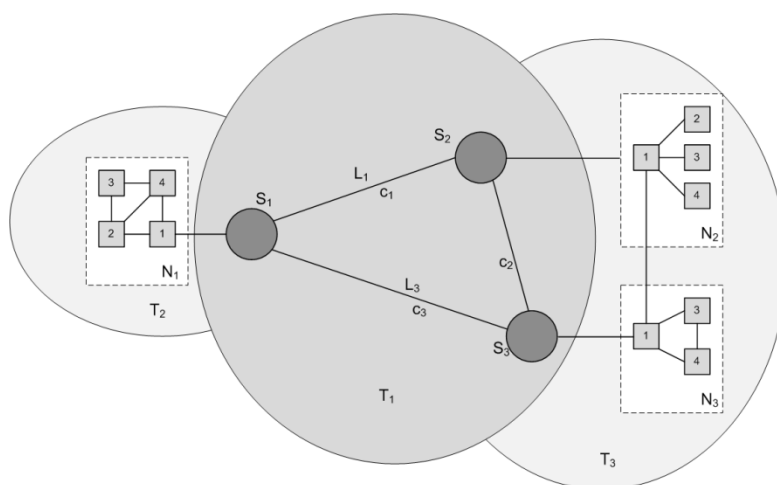


Рис. 2. Пример организации иерархически представленной сети для двух подзадач

На рис. 2 представлен пример конфигурации гетерогенного кластера для решения задачи $T = (T_1, T_2, T_3)$, состоящей из трех подзадач T_1 , T_2 и T_3 , причем выполнение T_1 требует подключения нижнего уровня спецвычислителей с T_2 и T_3 . Управляющие узлы S_1 , S_2 и S_3 объединены в сеть верхнего уровня, внутри каждого сектора второго слоя сконфигурирована своя сеть на базе кластера (для T_2) и между ними (для T_3).

Распределение задач и вычислительной нагрузки проводится в соответствии с таблицами вычислительных мощностей $Tr(p(n_i), q_i)$, где q_i – сводный параметр вычислительных затрат, допустимый к использованию на данном вычислительном базисе и следующий из совокупной трудоемкости элементарных операций, выполняемых в текущем узле n_i . На функциональном уровне задача представима в виде вершинного иерархического гиперграфа $H = (G, h)$ в котором основной граф иерархии G вырождается в ограниченное множество вершин $s_k \in S$, отвечающих за выполнение задачи $T_l \in T$. Структура иерархии h зависит от строения сети и формируется на этапе распределения задачи по множеству вычислительных узлов с учетом их аппаратных особенностей, ориентации на конкретное множество операций и пиковую производительность $p(n_j)$.

Временной граф решения совокупной задачи имеет аналогию с гиперграфом вычислительной нагрузки. Наихудшим вариантом решения задачи, естественно является последовательное решение на цепи элементов или последовательное решение этапов задачи на одном элементе. Соответственно коэффициент ускорения $e_{отн} = t_{решения}^p / t_{решения}^n$ можно определить как отношение времени реального параллельного (распределенного) решения $t_{решения}^n$ к времени последовательного решения $t_{решения}^p$. Так как внутри кластерных ускорителей обмен данными происходит через общую память, то время, затрачиваемое на обмен данными мало и может не учитываться, хотя при расчете необходимо учитывать время передачи данных $t_{обм}$ по внешним каналам данных, зависящее от их пропускной способности, тогда $t_{решения}^p = \max(t_{выч}) + \max(t_{обм})$. Естественно, что при генерации временного графа решений необходимо минимизировать как время вычислений $t_{выч}$, так и время обмена $t_{обм}$. Причем, чем выше эффективность распределения задачи, тем меньше совокупное время выполнения и выше общая эффективность решения задачи.

Для экспериментальной проверки теории, было выполнено моделирование и расчет модели на эвристическом уровне с реализацией вычислительных процедур высшего уровня при помощи технологии параллельного программирования MPI с применением динамической библиотеки BRD Deasy Tools, для реализации функций управления, обмена информацией с акселератором и вспомогательной синхронизации вычислительного процесса, предоставленных разработчиками платы Insys Corp вместе со средой разработки VisualDSP, Analog Devices Inc [7].

В качестве тестовой задачи была выбрана реализация модели работы гидроакустической антенны. Программа расчета модели выполнена для многолучевого эхолота с разными углами распространения волн. Для упрощения тестирования на данном этапе была выбрана двухпроцессорная конфигурация на базе аппаратного кластера под управлением базовой ПЭВМ через соответствующие библиотеки. На каждом задействованном процессоре второго слоя было просчитано по 50 дискретных лучей, за одну итерацию алгоритма, получены данные о поверхности дна. На каждом шаге итерации также происходил обмен данными. Из ведомого процессора к ведущему процессору передавались данные для вывода в файл, а также условие завершения моделирования на данном процессоре. Ведущий процессор так же возвращал флаг завершения процесса, это было необходимо для синхронизации обмена. После окончания моделирования фиксированного участка сектора

донной поверхности, данные возвращались на верхний слой гетерогенного кластера для отображения в графической оболочки, анализа, коррекции и выработки новых командных пакетов. Для последовательного решения время выполнения составило $t_{\text{решения}}^n = 21,65$ с, время распределенного решения $t_{\text{решения}}^p = 14,28$ с.

Без учета операций передачи данных $t_{\text{решения}}^n = 16,18$ с, $t_{\text{решения}}^p = 10,12$ с.

Полученные данные позволяют судить о том, что эффективность многопроцессорной реализации в рамках данной концепции высока. Процесс дальнейшего разбиения задачи на большее число процессоров будет целесообразен при соответствии временных затрат на вычисления и межпроцессорные взаимодействия, а также после окончательной проработке и проверке системы моделирования, позволяющей в автоматическом режиме и при помощи эксперта проводить решение задачи от определения алгоритмической базы вычислительного алгоритма до генерации оптимальных конфигураций гетерогенного кластера с последующим мониторингом и контролем процесса решения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Советов Б.Я., Яковлев С.А. Моделирование систем: Учебник для вузов – 3-е изд., перераб. и доп. – М.: Высш. шк., 2001. – 343 с.
2. Blume W, Eigenmann R. Performance Analysis of Parallelizing Compilers on the Perfect Benchmarks Programs// IEEE Transactions of Parallel and Distributed Systems, 3(6):643-656, November 1992.
3. Aronsson P. Automatic Parallelization of Simulation Code from Equation Based Simulation Languages, 2002.
4. Нурждин В. Н. Концептуальное программирование вычислительных моделей. – Иваново: ИЭИ, 1985.- 32 с.
5. Celler F.E. Hierarchical non-linear bond graphs: a unified methodology for modeling complex physical systems// Simulation Vol. 58, 1992. – №4. – 283 с.
6. ADSP 2106x SHARC User's manual. – Analog Devices Inc., 1996.
7. Базовый модуль ADP160QPCI v2.1. Руководство пользователя. Insys Corp. –М., 2002. – 54 с.

УДК 615.47

Е.А. Бабякина

ЭКОНОМИКО-СТАТИСТИЧЕСКОЕ МОДЕЛИРОВАНИЕ НАЛОГОВОГО БРЕМЕНИ И НАЛОГОВЫХ ДОХОДОВ СТАВРОПОЛЬСКОГО КРАЯ

Одним из наиболее дискуссионных вопросов современных налоговых отношений является вопрос о целесообразности и эффективности методов налогового стимулирования. Помимо традиционно постулируемых еще со времен классической экономической теории выводов о необходимости снижения налогов, получивших дополнительный импульс в рамках концепции «экономики предложения», существует целый ряд противоречивых эмпирических исследований о влиянии налогового бремени на экономическую активность.

В этой связи особую актуальность приобретает исследование количественных параметров взаимосвязи налогового бремени и экономической активности на эмпирическом материале городов и районов Ставропольского края.

Предположим, что налоговое бремя является одним из факторов, детерминирующим динамику экономического роста, тем более, что на формально-логическом уровне данное утверждение традиционно не вызывает сомнений.