



МЕТОДИКА РАЗРАБОТКИ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ

Лучше совсем не помышлять об отыскании
каких бы то ни было истин, чем делать это
без всякого метода.

Р. Декарт

Суперкомпьютеры и их применение

Содержание

2

- Модели параллельных вычислений
- Принципы разработки параллельных алгоритмов

Суперкомпьютеры и их применение © М.Л. Цымблер

Модели параллельных вычислений

3

- Модель "задачи-сообщения"
- Модель "процессы-каналы"

Суперкомпьютеры и их применение © М.Л. Цымблер

Модель "задачи-сообщения"

4

- Агрегированное представление графа "операции-операнды".
 - Вершины графа – задачи (набор операций).
 - Дуги – сообщения (обмен данными).
- Модель позволяет
 - Определить подзадачи одинаковой вычислительной сложности.
 - Обеспечить низкий уровень информационной зависимости между подзадачами.

Суперкомпьютеры и их применение © М.Л. Цымблер

Модель "процессы-каналы"

5

- Процесс* – выполняемая на процессоре *программа*, использует для своей работы часть локальной *памяти* процессора и содержит ряд *операций обмена данными* с другими процессами.
- Канал передачи данных* – *очередь сообщений*, в которую один или несколько процессов могут отправлять пересылаемые данные и из которой процесс-адресат может извлекать сообщения.
 - Каналы возникают динамически в момент выполнения первой операции приема/передачи с каналом.
 - Канал может соответствовать одной или нескольким командам приема/передачи данных различных процессов.
 - Канал имеет неограниченную емкость.
 - Операции приема сообщений по каналу могут приводить к блокировкам (если запрашиваемые процессом-приемником данные еще не были отправлены процессом-источником).
- Модель позволяет
 - Осуществить оптимальное распределение подзадач по процессорам.
 - Выполнить анализ эффективности разработанного параллельного метода.
 - Обеспечить возможность контроля и управления процессом выполнения параллельных вычислений.

Суперкомпьютеры и их применение © М.Л. Цымблер

Разработка параллельного алгоритма

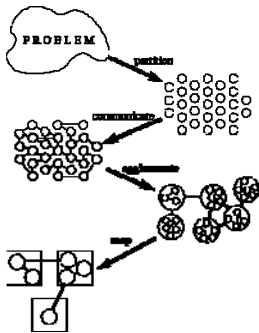
6

- Декомпозиция задачи на подзадачи, которые могут быть выполнены параллельно.
- Выделение информационных взаимодействий между подзадачами.
- Определение необходимой (доступной) вычислительной системы и оценка алгоритма (ускорение, эффективность и др.).
- Масштабирование подзадач в зависимости от количества процессоров вычислительной системы.
- Распределение подзадач между процессорами вычислительной системы.

Суперкомпьютеры и их применение © М.Л. Цымблер

Разработка параллельного алгоритма

7



Суперкомпьютеры и их применение © М.Л. Цымблер

Декомпозиция задачи

9

- Цель – разделение вычислений на независимые части, которые могут быть выполнены параллельно.
- Требования
 - Равный объем вычислений в подзадачах (балансировка загрузки).
 - Минимальные информационные зависимости между подзадачами (лучше 1 раз передать 100 байтов, чем 100 раз передать 1 байт :-).
- Подходы
 - Параллелизм данных.
 - Параллелизм задач.

Суперкомпьютеры и их применение © М.Л. Цымблер

Параллелизм данных и задач

10

- Параллелизм данных имеет место, когда вычисления заключаются в выполнении однотипной обработки каждого элемента набора данных.
 - Выделение подзадач сводится к разделению имеющихся данных.
 - Примеры: матричные вычисления, численные методы решения уравнений в частных производных и др.
- Параллелизм задач имеет место, когда вычисления заключаются в выполнении разных операций над одним и тем же набором данных.
 - Выделение подзадач сводится к функциональной декомпозиции.
 - Примеры: задачи обработки последовательности запросов к базам данных, вычисления с одновременным применением разных алгоритмов расчета и др.

Суперкомпьютеры и их применение © М.Л. Цымблер

Уровень декомпозиции

11

- Большое количество мелких подзадач обеспечивает высокий уровень параллелизма, но затрудняет анализ параллельных вычислений.
- Небольшое количество крупных подзадач дает ясную схему параллельных вычислений, но затрудняет эффективное использование большого количества процессоров.
- Оптимум: использование в качестве конструктивных элементов декомпозиции только тех подзадач, для которых методы параллельных вычислений являются известными.
 - Позволит обеспечить и простоту представления вычислительных схем, и эффективность параллельных расчетов.
 - Выбираемые подзадачи будем именовать *базовыми*.
 - Базовые задачи могут быть *элементарными (неделимыми)*, если не допускают дальнейшего разделения, или *составными* в противном случае.

Суперкомпьютеры и их применение © М.Л. Цымблер

Оценка декомпозиции

12

- Выполненная декомпозиция не увеличивает объем вычислений и необходимый объем памяти?
- Возможна ли при выбранном способе декомпозиции равномерная загрузка всех имеющихся процессоров?
- Достаточно ли выделенных частей процесса вычислений для эффективной загрузки имеющихся процессоров (с учетом возможности увеличения их количества)?

Суперкомпьютеры и их применение © М.Л. Цымблер

Информационные зависимости

13

- Этап, трудно отделимый от предыдущего. Выделение подзадач должно происходить с учетом возникающих информационных связей; после анализа объема и частоты необходимых информационных обменов между подзадачами может потребоваться повторение этапа декомпозиции.
- Схемы взаимодействия подзадач:
 - Локальная vs глобальная
 - Структурная vs произвольная
 - Статическая vs динамическая
 - Синхронная vs асинхронная

Суперкомпьютеры и их применение © М.Л. Цымблер

Схемы взаимодействия

14

- Локальная схема* – в каждый момент времени передачи выполняются только между небольшим числом подзадач (располагаемых, как правило, на соседних процессорах).
- Глобальная схема* – в коммуникации принимают участие все подзадачи.
- Структурная схема* приводит к формированию стандартных схем коммуникации (кольцо, прямоугольная решетка и др.).
- Произвольная схема* выполняемых операций передач данных не носит характер однородности.
- Статическая схема* – моменты и участники взаимодействия фиксируются на этапах проектирования и разработки параллельных программ.
- Динамическая схема* – структура операции передачи данных определяется в ходе выполняемых вычислений.
- Синхронная схема* – операции передачи данных выполняются только при готовности всех участников взаимодействия и завершаются только после полного окончания всех коммуникационных действий.
 - Более проста для использования, но может привести к тупикам.
- Асинхронная схема* – участники взаимодействия могут не дожидаться полного завершения действий по передаче данных.
 - Более сложна для использования, но может повысить эффективность.

Суперкомпьютеры и их применение © М.Л. Цымблер

Оценка схемы взаимодействия

15

- Соответствует ли вычислительная сложность подзадач интенсивности их информационных взаимодействий?
- Является ли одинаковой интенсивность информационных взаимодействий для разных подзадач?
- Является ли схема информационного взаимодействия локальной?
- Не препятствует ли выявленная информационная зависимость параллельному решению подзадач?

Суперкомпьютеры и их применение © М.Л. Цымблер

Масштабирование подзадач

16

- Если количество подзадач больше числа планируемых к использованию процессоров
 - необходимо выполнить *агрегацию* (укрупнение) подзадач с помощью вышензложенных походов.
 - первые кандидаты на объединение – подзадачи с высокой степенью информационной взаимозависимости.
- Если количество подзадач меньше числа планируемых к использованию процессоров
 - необходимо выполнить *детализацию* (разделение) подзадач с помощью вышензложенных походов.
 - детализация несложна, если для базовых задач известны методы параллельных вычислений.
- Масштабирование – разработка правил агрегации и детализации подзадач, в которых количество процессоров является параметром.

Суперкомпьютеры и их применение © М.Л. Цымблер

Оценка масштабирования

17

- Не ухудшится ли локальность вычислений после масштабирования имеющегося набора подзадач?
- Имеют ли подзадачи после масштабирования одинаковую вычислительную и коммуникационную сложность?
- Соответствует ли количество задач числу имеющихся процессоров?
- Зависят ли параметрически правила масштабирования от количества процессоров?

Суперкомпьютеры и их применение © М.Л. Цымблер

Распределение задач по процессорам

18

- Управление распределением нагрузки для процессоров возможно только для мультикомпьютеров. Для мультипроцессоров распределение нагрузки обычно выполняет ОС автоматически.
- Данный этап является избыточным, если количество подзадач совпадает с числом имеющихся процессоров, а топология сети передачи данных вычислительной системы представляет собой полный граф (т.е. все процессоры связаны между собой прямыми линиями связи).
- Показатель эффективности распределения – относительная доля времени, в течение которого процессоры использовались для вычислений, связанных с решением исходной задачи (не обмена и ожидания).

Суперкомпьютеры и их применение © М.Л. Цымблер

Распределение задач по процессорам

19

- Необходимо обеспечить равномерное распределение вычислительной нагрузки между процессорами и минимизировать количество сообщений, передаваемых между процессорами.
- Требование минимизации информационных обменов между процессорами может противоречить условию равномерной загрузки процессоров (все подзадачи можно поместить на один процессор).
- Оптимальное решение проблемы распределения подзадач основывается на анализе информационной связности графа "подзадачи - сообщения".
 - Подзадачи, между которыми имеются информационные взаимодействия, целесообразно размещать на процессорах, между которыми существуют прямые линии передачи данных.

Суперкомпьютеры и их применение © М.Л. Цымблер

Динамическая балансировка

20

- Решение вопросов балансировки вычислительной нагрузки значительно усложняется, если схема вычислений может изменяться в ходе решения задачи.
 - Неоднородные сетки при решении уравнений в частных производных, разреженность матриц и др.
 - Используемые на этапах проектирования оценки вычислительной сложности решения подзадач могут иметь приближенный характер.
 - Количество подзадач может изменяться в ходе вычислений.
- В таких ситуациях может потребоваться перераспределение базовых подзадач между процессорами уже непосредственно в процессе выполнения параллельной программы – выполнить *динамическую балансировку вычислительной нагрузки*.

Суперкомпьютеры и их применение © М.Л. Цымблер

Схема "менеджер-рабочий"

21

- Допущения
 - Подзадачи могут возникать и завершаться в ходе вычислений.
 - При этом информационные взаимодействия между подзадачами либо полностью отсутствуют, либо минимальны.
- Участники
 - Менеджер – выделенный для управления распределением нагрузки процессор системы, которому доступна информация обо всех имеющихся подзадачах.
 - Исполнители – остальные процессоры, которые для получения вычислительной нагрузки обращаются к менеджеру.
- Схема
 - Порождаемые в ходе вычислений новые подзадачи передаются обратно менеджеру и могут быть получены для решения при последующих обращениях исполнителей.
 - Завершение вычислений происходит в момент, когда исполнители завершили решение всех переданных им подзадач, а менеджер не имеет каких-либо вычислительных работ для выполнения.

Суперкомпьютеры и их применение © М.Л. Цымблер

Оценка распределения задач по процессорам

22

- Не приводит ли распределение нескольких задач на один процессор к росту дополнительных вычислительных затрат?
- Существует ли необходимость динамической балансировки вычислений?
- Не является ли процессор-менеджер "узким" местом при использовании схемы "менеджер-исполнитель"?

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

23

- Дано большое количество тел (планет, звезд и др.), для каждого из которых известна масса, начальное положение и скорость.
- Под действием гравитации положение тел меняется.
- Требуемое решение задачи состоит в моделировании динамики изменения системы N тел на протяжении некоторого интервала времени.

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

24

- Для проведения моделирования интервал времени разбивается на временные отрезки небольшой длительности, на каждом шаге моделирования вычисляются силы, действующие на каждое тело, и обновляются скорости и положения тел.
- Очевидный алгоритм решения задачи N тел состоит в рассмотрении на каждом шаге моделирования всех пар объектов физической системы и выполнении для каждой получаемой пары всех необходимых расчетов.
- При таком подходе время выполнения одной итерации моделирования (τ – время перевычисления параметров одной пары тел):

$$T_1 = \tau N(N-1)/2$$

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

25

- Декомпозиция
 - *Базовая подзадача* – весь набор вычислений, связанных с обработкой данных одного какого-либо тела физической системы.
- Выделение информационных зависимостей
 - Выполнение вычислений в подзадаче возможно только в случае, когда имеются данные обо всех телах физической системы.
 - Перед началом каждой итерации моделирования каждая подзадача должна получить все необходимые сведения от всех других подзадач системы.
 - Такая процедура передачи данных именуется *операцией обобщенного сбора данных (multi-node gather или all gather)*.

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

26

□ Операция обобщенного сбора данных

- **Метод 1:** Обмен данными осуществляется в ходе последовательности шагов, на каждом из которых все имеющиеся подзадачи разбиваются попарно и обмен данными осуществляется между подзадачами образовавшихся пар ($N-1$ итерация).
- **Метод 2:** Первый шаг метода выполняется точно также, как в методе 1 - после выполнения этого шага подзадачи будут содержать свои данные и данные подзадач, с которыми они образовывали пары. Как результат, на втором шаге пары подзадач могут быть образованы для обмена данными сразу о двух телах физической системы. После завершения второго шага каждая подзадача будет содержать сведения о четырех телах системы и др. Тем самым, общее количество шагов для выполнения всех требуемых обменов является равным $\log_2 N$.

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

27

□ Масштабирование и распределение подзадач по процессорам

- Если число тел физической системы N значительно превышает количество процессоров p , рассмотренные ранее подзадачи следует укрупнить, объединив в рамках одной подзадачи вычисления для группы (N/p) тел.
- После проведения подобной агрегации число подзадач и количество процессоров будет совпадать.
- При распределении подзадач между процессорами необходимо обеспечить наличие прямых коммуникационных линий между процессорами с подзадачами, между которыми имеются информационные обмены при выполнении операции сбора данных.

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

28

□ Анализ эффективности

- Предложенные варианты отличаются только методами выполнения информационных обменов и для сравнения подходов достаточно определить длительность операции обобщенного сбора данных.
- Используем для оценки времени передачи сообщений модель, предложенную Хокни (*Hockney*), в которой трудоемкость операции коммуникации между узлами вычислительной системы оценивается в соответствии с выражением

$$t_{nd}(m) = \alpha + m / \beta,$$

где α – латентность сети передачи данных, m – размер передаваемого сообщения в байтах, β – пропускная способность сети.

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

29

□ Анализ эффективности

- ▣ Длительность выполнения операции сбора данных для первого метода реализации

$$T_p^1(comm) = (p-1)(\alpha + m(N/p)/\beta)$$

- ▣ При использовании второго метода для итерации с номером i объем сообщений оценивается как $2^{i-1}(Nm/p)$. Т.е. длительность выполнения операции сбора данных

$$T_p^2(comm) = \sum_{i=1}^{\log p} (\alpha + 2^{i-1}m(N/p)/\beta) = \alpha \log p + m(N/p)(p-1)/\beta$$

Суперкомпьютеры и их применение © М.Л. Цымблер

Пример: задача N тел

30

□ Анализ эффективности

- ▣ Сравнение полученных выражений показывает, что второй разработанный способ параллельных вычислений имеет существенно более высокую эффективность, несет меньшие коммуникационные затраты и допускает лучшую масштабируемость при увеличении количества используемых процессоров.

Суперкомпьютеры и их применение © М.Л. Цымблер

Заключение

31

□ Модели параллельных вычислений

- ▣ Модель "задачи-сообщения"
- ▣ Модель "процессы-каналы"

□ Принципы разработки параллельных алгоритмов

- ▣ Декомпозиция задачи на подзадачи, которые могут быть выполнены параллельно.
- ▣ Выделение информационных взаимодействий между подзадачами.
- ▣ Определение необходимой (доступной) вычислительной системы и оценка алгоритма (ускорение, эффективность и др.).
- ▣ Масштабирование подзадач в зависимости от количества процессоров вычислительной системы.
- ▣ Распределение подзадач между процессорами вычислительной системы.

Суперкомпьютеры и их применение © М.Л. Цымблер
