

Лабораторная работа 1

Имеется распределенная вычислительная система (ВС) укомплектованная n элементарными машинами (ЭМ). Задан набор из m параллельных задач. Каждая задача $j \in J = \{1, 2, \dots, m\}$ характеризуется временем t_j решения и количеством r_j элементарных машин необходимых для неё.

Требуется построить расписание S решения параллельных задач на распределенной ВС. Для каждой задачи необходимо определить момент времени τ_j начала решения её ветвей и их распределение по элементарным машинам. Пусть $x_{ji} \in C = \{1, 2, \dots, n\}$ – номер ЭМ, на которую распределена ветвь $i \in \{1, 2, \dots, r_j\}$ задачи $j \in J$.

Обозначим через $J(t) = \{j \in J \mid \tau_j \leq t \leq \tau_j + t_j\}$ множество задач, решаемых на распределенной ВС в момент времени t .

Будем называть расписание $S = (\tau_1, \tau_2, \dots, \tau_m; x_{11}, x_{12}, \dots, x_{1r_1}, \dots, x_{m1}, x_{m2}, \dots, x_{mr_m})$ допустимым, если оно удовлетворяет ограничениям.

1) В любой момент времени на ресурсах распределенной ВС решается не более n ветвей параллельных задач:

$$\sum_{j \in J(t)} r_j \leq n, \quad \forall t \in \mathbf{R}.$$

2) Ветви параллельных задач решаются на разных элементарных машинах:

$$\prod_{j \in J(t)} \prod_{j' \in J(t) \setminus \{j\}} (x_{ji} - x_{j'i'}) \neq 0, \quad \forall t \in \mathbf{R}, \quad i = 1, 2, \dots, r_j, \quad i' = 1, 2, \dots, r_{j'}.$$

Обозначим через Ω множество допустимых расписаний. В качестве показателя оптимальности расписаний будем использовать время $T(S)$ окончания решения последней задачи

$$T(S) = \max_{j \in J} \{\tau_j + t_j\}.$$

Итак, требуется найти допустимое расписание $S \in \Omega$, доставляющее минимум целевой функции $T(S)$. Формально

$$T(S) = \max_{j \in J} \{\tau_j + t_j\} \rightarrow \min_{S \in \Omega} \quad (1)$$

при ограничениях:

$$\sum_{j \in J(t)} r_j \leq n, \quad \forall t \in \mathbf{R}, \quad (2)$$

$$\prod_{j \in J(t)} \prod_{j' \in J(t) \setminus \{j\}} (x_{ji} - x_{j'i'}) \neq 0, \quad \forall t \in \mathbf{R}, \quad i = 1, 2, \dots, r_j, \quad i' = 1, 2, \dots, r_{j'}, \quad (3)$$

$$x_{ji} \in C, \quad j = 1, 2, \dots, m, \quad i = 1, 2, \dots, r_j, \quad (4)$$

$$\tau_j \in \mathbf{R}, \quad j = 1, 2, \dots, m. \quad (5)$$

Задача (1) – (5) относится к дискретной оптимизации и является трудноразрешимой.

Один из подходов к приближенному решению задачи (1) – (5) основан на её сведении к задаче двумерной упаковки прямоугольников в полуограниченную полосу (2D Strip Packing, 2DSP). Параллельная задача $j \in J$ представляется в виде прямоугольника шириной r_j и высотой t_j условных единиц. Ширина полосы – n условных единиц, высота не ограничена. Требуется упаковать прямоугольники без их вращений и пересечений в полосу так, чтобы высота упаковки была минимальной. На рис.1 приведен пример упаковки 5 прямоугольников (задач) в полосу шириной 10 условных единиц (элементарных машин). Задача 1 запускается на решение в нулевой момент времени и использует элементарные машины 1 – 5, задача 4 начинает решаться в момент времени 2 и использует ЭМ 8, 9, 10. Значение целевой функции $T(S) = 8$.



Рис. 1. Пример упаковки прямоугольников в полуограниченную полосу:
 $m = 5$; $n = 10$; $T(S) = 8$

В рамках лабораторной работы требуется выполнить нижеследующие задания.

1. Написать программу, реализующую алгоритмы NFDH и FFDH [1, 2] для приближенного решения задачи (1) – (5). В качестве входных параметров программа получает имя файла с набором задач, количество n ЭМ в системе и название алгоритма (NFDH или FFDH). Результат работы программы: расписание S решения задач, значение $T(S)$ целевой функции, отклонение ε значения целевой функции от её нижней границы T' , время t выполнения алгоритма в секундах.

$$\varepsilon = \frac{T(S) - T'}{T'}, \quad T' = \frac{1}{n} \sum_{j \in J} r_j t_j.$$

Упорядочивание задач в алгоритмах NFDH и FFDH выполнять сортировкой подсчетом (Counting Sort).

Для реализации алгоритма FFDH с вычислительной сложностью $O(n \log n)$ рекомендуется использовать бинарное дерево турнира (tournament tree, max winner tree). Каждый лист дерева – это уровень упаковки, а значение листа – это количества свободных ЭМ на уровне. Каждый внутренний узел дерева содержит максимальное из значений левого и правого дочерних узлов. В таком дереве поиск первого подходящего уровня (First Fit) выполняется за время $O(\log n)$. Подробнее (примеры для алгоритма FF):

- <http://cpct.sibsutis.ru/~apaznikov/teaching/uploads/Site/winner-tree.pdf>
- http://iphotonics.dyndns.org/dwhuang/files/2012/02/DS_10.pdf
- <http://logic.pdmi.ras.ru/midas/sites/default/files/Johnson-Tuesday.pdf>

2. Исследовать время выполнения алгоритмов в зависимости от количества m задач в наборе. Сформировать 10 наборов задач с $m = 500, 1000, \dots, 5000$; параметры задач генерировать как равномерно распределенные псевдослучайные числа $r_j \in \{1, 2, \dots, n\}$, $t_j \in \{1, 2, \dots, 100\}$. Рассмотреть случаи $n = 1024, 4096$.

Ответить на вопросы:

- Какова вычислительная сложность алгоритмов?
- Как зависит время работы алгоритмов от значения параметра m ?
- Как зависит время работы алгоритмов от значения параметра n ?

3. Провести сравнительный анализ значений целевой функции от расписаний, формируемых алгоритмами. Сформировать 10 наборов задач ($m = 500, 1000, \dots, 5000$); параметры задач генерировать как равномерно распределенные псевдослучайные числа $r_j \in \{1, 2, \dots, n\}$, $t_j \in \{1, 2, \dots, 100\}$. Во всех 10 экспериментах $n = 1024$. По результатам экспериментов построить оценки математического ожидания и среднеквадратического отклонения случайной величины ε .

Ответить на вопрос: какой из алгоритмов, на рассмотренных наборах задач, формировал более точные расписания?

4. Провести сравнительный анализ значений целевой функции от расписаний, формируемых алгоритмами. На основе протоколов решения параллельных задач на промышленных распределенных ВС [3] сформировать наборы задач ($m = 500, 1000, 1500$) для любой из систем: LLNL uBGL, LLNL Atlas, LLNL Thunder. По результатам экспериментов построить оценки математического ожидания и среднеквадратического отклонения случайной величины ε .

Ответить на вопрос: какой из алгоритмов, на рассмотренных наборах задач, формировал более точные расписания?

Контрольные вопросы

1. Дать определение мультипрограммного режима функционирования распределённых ВС.
2. Пояснить содержательный смысл целевой функции (1).
3. Пояснить смысл ограничений (2) – (5).
4. Объяснить, почему задача (1) – (5) относится к трудноразрешимым задачам дискретной оптимизации.

Литература

1. Heidi Smith. Level Algorithms and Shelf Algorithms for // <http://users.cs.cf.ac.uk/C.L.Mumford/heidi/Approaches.html>
2. Jan H van Vuuren. 2D Strip Packing Problem // [http://dip.sun.ac.za/~vuuren/repositories/levelpaper/spp\[1\].htm](http://dip.sun.ac.za/~vuuren/repositories/levelpaper/spp[1].htm)
3. Logs of Real Parallel Workloads from Production Systems // Parallel Workloads Archive. – <http://www.cs.huji.ac.il/labs/parallel/workload/logs.html>