

УДК 681.324

С.В. Минухин, К.П. Стативкин

Харьковский национальный экономический университет им. С. Кузнеця

МЕТОД ПЛАНИРОВАНИЯ РЕСУРСОВ В РАСПРЕДЕЛЕННОЙ ГЕТЕРОГЕННОЙ СИСТЕМЕ

В статье рассмотрена процедура планирования в распределенной гетерогенной системе на основе использования метода о наименьшем покрытии. Рассмотрены основные параметры и ограничения, накладываемые на решение задачи, а так же практическая реализация метода планирования ресурсов. Делается вывод о целесообразности использования метода о наименьшем покрытии при планировании заданий распределенных систем.

Ключевые слова: гетерогенная система, планирование, ресурс, наименьшее покрытие.

Вступление

Эксплуатация современных Grid-систем связана с необходимостью повышения их производительности. Она определяется такими характеристиками, как загруженность ресурсов, время выполнения всех заданий на имеющихся ресурсах, время завершения выполнения наиболее поздних заданий и т.д. [1]. В окружении Grid владельцы и потребители ресурсов имеют различные цели, используют различные стратегии и экономические схемы регулирования спроса и предложения. Таким образом, актуальной проблемой является разработка систем управления ресурсами Grid, нацеленных на оптимизацию отношений между владельцами ресурсов и пользователями в соответствии с выбранными ими стратегиями [2]. Поскольку существует тенденция к увеличению количества гетерогенных вычислительных кластеров и заданий с дифференцированной сложностью решения, актуальным является использование более тонкой настройки процедур планирования, которая позволит эффективно «упаковать» задания на ресурсы с одной стороны, а с другой – обеспечить высокую загрузку всех ресурсов. Это может быть достигнуто путем использования эвристических алгоритмов планирования и дополнительных компонент (модулей) в самом механизме планирования ресурсов распределенной системы. Планировщики распределенных систем (Grid) являются ключевыми компонентами в взаимосвязанной информационно-коммуникационной среде. В гетерогенных распределенных системах (Grid) используются различные политики и стратегии работы планировщиков. Как правило, сравнивают эффективность различных алгоритмов планирования в рамках единого механизма планирования распределенной системы. В случае же

сравнения результатов работы различных систем планирования, оценки эффективности планировщиков не всегда являются корректными и сопоставимыми, что требует проведения дополнительных исследований. Известно, что в условиях высокой интенсивности потоков заданий и гетерогенности распределенной системы используемые алгоритмы не являются достаточно эффективными, а более предпочтительными являются методы пакетных режимов обработки заданий [3]. **Цель статьи** – рассмотреть задачу о покрытии как метод планирования ресурсов в распределенной гетерогенной системе.

Основная часть

Для достижения поставленной цели был разработан программный продукт, который позволяет производить планирование ресурсов Grid системы основываясь на решении задачи о наименьшем покрытии. Математическая модель метода планирования предлагается в виде задачи линейного булевого программирования

$$L_t = \sum_{j=1}^n x_j(t_k) \rightarrow \min \quad (1)$$

при ограничениях

$$\sum_{j=1}^n \beta_{ij} x_j(t_k) \geq 1, i = \overline{1, m} \quad (2)$$

$$\beta_{ij} \in \{0,1\}; x_j(t_k) \in \{0,1\}$$

где m – количество заданий, подлежащих планированию; n – количество ресурсов исследуемой системы, доступных и свободных на момент планирования; $t_k \in [T_0, T_N]$.

Планирование осуществляется на интервале времени $[T_0, T_N]$, где T_0 – время начала планирования; T_N – время окончания планирования заданий глобальной очереди.

Задачу (1), (2) можно рассматривать как задачу определения минимального числа столбцов в булевой матрице B , покрывающего единицами все строки данной матрицы, элементы которой в контексте решения

задачи планирования интерпретируются следующим образом: столбцам соответствуют свободные на момент планирования ресурсы системы, строкам – задания, ресурсе будет решаться с учетом, что на одном ресурсе подлежащие планированию, которые должны быть решены на этих ресурсах [3]. Так же при решении задачи необходимо учесть тот факт, что для обеспечения максимальной производительности и при этом не вызывать дополнительного энергопотребления на охлаждение необходимо поддерживать нагрузку на ресурсе не более 75 – 80%. Таким образом появляется еще одно

ограничение вида:

$$K_m = \sum_{j=1}^n x_j \leq P \quad (3)$$

где n – количество заданий, подлежащих планированию; m – количество ресурсов исследуемой системы, на которых возможно решение x_j задачи; P - максимально допустимый процент загрузки ресурса, $P \in [75, 80]$. Распределение задач на ресурс осуществляется таким образом, чтобы ожидаемое время выполнения не превышало указанное в спецификации задания. Расчет времени выполнения осуществляется по формуле:

$$V_m = \frac{t}{f} \quad (4)$$

где m – количество ресурсов исследуемой системы; t – время выполнения задания, указанное в спецификации; f – частота работы процессора. Так же предлагается расчет суммарного можно рассчитать по формуле:

$$T_m = \sum_{j=1}^n t_{proc} - t_{deadline} \quad (5)$$

где n – количество заданий, подлежащих планированию; m – количество ресурсов исследуемой системы; t_{proc} – время выполнения задания на ресурсе; $t_{deadline}$ – крайний срок выполнения задания.

Анализ результатов вычислительного эксперимента. В качестве входных параметров для решения задачи о покрытии является матрица возможного решения задания на ресурсе, столбцы которой представлены ресурсы, а строки – заданиями. Пример входной матрицы представлен в табл.1.

Таблица 1

Задания	Ресурсы							
	R1	R2	R3	R4	R5	R6	R7	R8
task1	1			1				
task2		1						1
task3					1			1
task4	1						1	
task5				1		1		
task6			1				1	
task7	1							1
task8		1	1					
task9					1			1
task10		1	1					
task11	1			1				
task12						1	1	

Таблица 2

Задания	Ресурсы							
	R1	R2	R3	R4	R5	R6	R7	R8
task1	1							
task2		1						
task3					1			
task4							1	
task5				1				
task6			1					
task7	1							
task8			1					
task9					1			
task10		1						
task11	1							
task12						1		

Как видно на рисунке не все ресурсы попали в конечное покрытие, что свидетельствует о решении задачи наименьшего покрытия. При этом следует отметить, что суммарная нагрузка на каждый из ресурсов не превышает налагаемые ограничения.

Практическая реализация метода.

Разработанное программное решение реализовано на языке программирования C# и позволяет произвести следующие операции:

- 1) Генерация входного пакета заданий
- 2) Формирование матрицы возможного решения
- 3) Формирование матрицы соответствия
- 4) Расчет суммарного времени выполнения задач на ресурсе
- 5) Расчет суммарного времени запаздывания заданий на ресурсе

Для практической реализации было выбрано пять моделей процессоров, а именно: Test CPU 12, Test CPU 4, Intel Pentium M, AMD Athlon, Intel Core i5. В программном модуле модели процессоров хранятся в базе данных, что позволяет осуществлять изменение характеристик уже существующих моделей или же добавить новый тип модели. Для начала работы необходимо ввести объем входного пакета заданий, минимальные и максимальные сроки выполнения заданий. После чего на основании введенной информации формируется пакет заданий, причем время выполнения генерируется случайным образом на промежутке, введенном пользователем. После чего формируется матрица возможного решения задания на ресурсе исходя из способности ресурса решить задачу в указанный срок, согласно формуле (4). При этом допускается, что одно задание может выполняться на нескольких ресурсах. Далее на основании матрицы возможного решения задания согласно алгоритму решения задачи о покрытии формируется матрица соответствия, в которой строго указано на каком ресурсе какая задача будет решаться. Реализация метода решения задачи о покрытии на языке программирования C# приведена ниже.

```
public void MakeNewPokritie()
```

```

{
int kol = Create_zn.kol_zad;
pokr = new int[5,kol];
pokr2 = new int[5, kol];
for (int i = 0; i < 5; i++)
for (int j = 0; j < kol; j++)
{
pokr[i, j] = int.Parse(dgvZadachaPokritia[i + 1,
j].Value.ToString());
pokr2[i, j] = int.Parse(dgvZadachaPokritia[i + 1,
j].Value.ToString());
}
for (int i = 0; i < kol; i++)
{
int k = Uniform(1, 5);
if (pokr[k,i]==1)
{
if (k > 1)
{
k--;
}
else k++;
}
pokr[k, i] = 1;
pokr2[k, i] = 1;
}
int[] postupilo = new int[5];
for (int i = 0; i < 5; i++)
{
postupilo[i] = 0;
for (int j = 0; j < kol; j++)
{
if (pokr[i,j]!=0)

```

```

for (int j = 0; j < 5; j++)
{
if (pokr[j, i] == 1 && x == 0)
x = j;
else if (pokr[j, i] == 1)
{
y = j;
}
}
if ((postupilo[x] / kol) * 100 < 80)
{
pokr[y, i] = 0;
postupilo[y]--;
}
else
{
pokr[x, i] = 0;
postupilo[x]--;
}
}
}}

```

На основании сгенерированного пакета заданий с указанной длительностью производится расчет суммарного времени запаздывания заданий на каждом из ресурсов системы. Расчет производится согласно формуле (5).

Результат работы приложения предоставляется пользователю в виде таблиц DataGridView, которые отображают информацию о полученных в результате вычислений матрицы возможного решения заданий и матрицы соответствия.

На рис. 1 предоставлен графический интерфейс разработанного программного решения, на котором

представлены полученные в ходе экспериментов программный модуль рассчитывает общее время работы результаты. каждого из представленных ресурсов и среднее время выполнение задания, что позволяет проанализировать рабочую загрузку каждого ресурса исследуемого кластера.

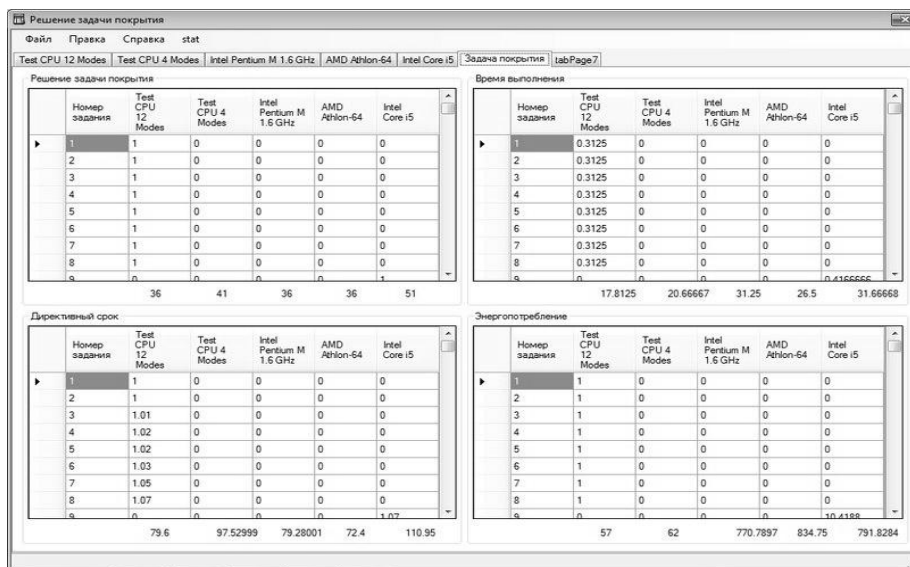


Рис. 1. Интерфейс программного решения

Выводы. Планирование ресурсов играет основную роль в Grid системах. Поскольку правильное распределение заданий на ресурсы позволит оптимальнее использовать систему в целом. На сегодняшний день существует большое количество методов распределения задач системы Grid. В статье был рассмотрен метод, основанный на решении задачи о покрытии и его реализация средствами языка C#. Проведенное исследование и полученные результаты показали эффективность предложенного метода планирования ресурсов в Grid системе.

Список литературы

1. Пономаренко В.С. Методы и модели планирования ресурсов в GRID-системах: монография./ В.С.Пономаренко, С.В. Листровой, С.В. Минухин, С.В. Знахур. – ИД «ИНЖЕК», 2008. – 408 с.
2. Мащенко Е.Н. // Оптимизация процессов планирования заданий в грид-системах с использованием комплекса имитационных моделей //

3. Минухин С.В. Метод планирования ресурсов в распределенной гетерогенной системе и исследование его практической реализации // С.В. Минухин, С.В. Баранник, С.В. Знахур, Р.И. Зубатюк, А.В. Соболев, Міжнародна конференція "Високопродуктивні обчислення" 2012.

Рецензент: д.т.н., проф. каф. ЭВМ ХНУРЭ Н.М. Кораблев.

Авторы:

МИНУХИН Сергей Владимирович, к.т.н., проф. каф. информационных систем, Харьковский национальный экономический университет им. С. Кузнеця, Харьков

E-mail: ms_vl@mail.ru

СТАТИВКИН Кирилл Павлович

Харьковский национальный экономический университет им. С. Кузнеця, Харьков, магистр.

E-mail – kirill.stativkin@hotmail.com

МЕТОД ПЛАНУВАННЯ РЕСУРСІВ У РОЗПОДІЛЕНІЙ ГЕТЕРОГЕННІЙ СИСТЕМІ

С.В. Мінухін, К.П. Статівкін

У статті розглянута процедура планування в розподіленій гетерогенній системі на основі вико-користування методу про найменшому покритті. Розглянуто основні параметри і обмеження накладаються на вирішення завдання, а так само практична реалізація методу планування ресурсів. Робиться висновок про доцільність використання методу про найменшому покритті при плануванні завдань розподілених систем.

Ключевые слова: талон амбулаторного пациента, показатели общей заболеваемости, интенсивные показатели заболеваемости.

METHOD OF RESOURCE SCHEDULING IN DISTRIBUTED HETEROGENEOUS SYSTEM

S. Minukhin, K. Stativkin

The article deals with the planning process in a distributed heterogeneous system is based on the use of the method of the least coverage. There are considered the basic parameters and constraints imposed on the solution, and the practical implementation of the method of resource planning. The conclusion about expediency of use of the method of the least coverage for job scheduling, distributed systems.

Keywords: heterogeneous system, planning, resource, the lowest floor.