

МЕТОДЫ РАСПРЕДЕЛЕНИЯ ЗАДАНИЙ МЕЖДУ ПРОЦЕССОРАМИ ВЫЧИСЛИТЕЛЬНОГО КЛАСТЕРА, ИСПОЛЬЗУЮЩИМИ ТЕХНОЛОГИИ МАСШТАБИРОВАНИЯ ЧАСТОТЫ И НАПРЯЖЕНИЯ

Рассмотрены методы распределения независимых заданий, одновременно поступающих на выполнение на параллельные процессоры вычислительного кластера. Предложены методы распределения заданий для однородных и неоднородных кластерных архитектур, отличающиеся предварительной сортировкой заданий входного потока. Для распределения заданий использованы подходы, базирующиеся на определении текущего значения коэффициента загрузки процессоров кластера и методе Round Robin, а также методы масштабирования частоты и напряжения одиночных процессоров. Приведены результаты моделирования исследуемых методов распределения заданий, подтверждающие их эффективность для обеспечения требуемого уровня балансировки загрузки для однородных и неоднородных вычислительных кластеров.

The methods of scheduling of independent tasks, which are arrival at the same time to perform on the parallel processors of computing cluster are developed. The methods of job distribution for homogeneous and heterogeneous cluster architectures, different previously sorted of the input stream. To distribute the task to use the approach based on determining the current value of the utilization of processor and based on the Round Robin method which utilized embedded technologies of scaling the frequency and voltage of single processors. The results of the simulation study methods of tasks distribution are given which proving their effectiveness to achieve the desired level of load balancing for computing clusters with homogeneous and heterogeneous architecture.

1. Введение

Одной из актуальных задач эксплуатации вычислительных ресурсов (кластеров, ЦОД) является обеспечение эффективности функционирования распределенных вычислительных систем. Для их решения необходимы методы, обеспечивающие требуемый уровень загруженности вычислительных ресурсов, качество обслуживания пользователей (минимизацию времени или стоимости выполнения заданий), минимизацию затрат на их содержание и эксплуатацию, включающую оптимизацию энергопотребления процессорами, полученную на основе технологий масштабирования частоты и напряжения. В общем случае эти задачи связаны между собой: минимизация времени выполнения заданий (времени работы процессора) обеспечивает качество обслуживания, однако снижение при этом уровня энергопотребления процессора не всегда представляется возможным, так как уменьшение времени выполнения задания требует использования более производительных режимов его работы. Важным при этом является поддержка штатного режима эксплуатации процессора (узла) кластера, так как его нарушение приводит к увеличению затрат на использование вспомогательного оборудования (например, системы охлаждения). С другой стороны, простаивание процессора при-

водит к уменьшению его коэффициента загрузки (использования) и увеличению холостого времени его работы. При этом при обеспечении работы процессора в штатном режиме также возникает задача ограничения коэффициента загрузки, который определяется некоторым пороговым значением (в пределах 80 – 90 % от пиковой). В условиях высокой интенсивности потоков заданий приведенные задачи дополняются необходимостью равномерного распределения заданий между параллельно работающими процессорами вычислительного кластера.

Целью данной работы является усовершенствование, исследование и сравнительный анализ методов распределения выполняющихся на параллельных процессорах вычислительного кластера заданий, имеющих произвольные длительности и директивные сроки, которые обеспечивают балансировку загрузки. Данная работа представляет собой дальнейшее развитие и исследование методов распределения заданий на параллельных, использующих технологии масштабирования частоты и напряжения, процессорах кластера, предложенных в работах [1, 3].

2. Постановка задачи

Пусть на вычислительный кластер, состоящем из m процессоров (узлов), одновременно

поступает n независимых заданий t_j (l_j, d_j), описываемых длительностью l_j и директивным сроком выполнения d_j .

Процессоры вычислительного кластера имеют разные штатные частоты и напряжения работы, причем, в зависимости от уровня их загрузки и характеристик поступающих на обработку потока (пакета) заданий, на каждом из них могут быть использованы режимы масштабирования частоты и напряжения, переключения между которыми управляются операционной системой.

Требуется разработать методы распределения заданий между параллельно работающими процессорами вычислительных кластеров с однородной и неоднородной архитектурой, обеспечивающие максимальное значение коэффициента балансировки загрузки процессоров β , определяемого по формуле

$$\beta = 1 - \frac{d}{k_{aver}}, \quad (1)$$

где

$$k_{aver} = \frac{\sum_{i=1}^m k_i}{m}, \quad d = \sqrt{\frac{\sum_{i=1}^m (k_{aver} - k_i)^2}{m-1}},$$

k_{aver} – средний коэффициент загрузки всех процессоров; k_i – коэффициент средней загрузки i -го процессора.

Коэффициент средней загрузки i -го процессора на каждом шаге распределения заданий рассчитывается как отношение времени завершения текущего (последнего) задания к его директивному сроку.

Методы решения поставленной задачи базируются на выполнении следующих этапов.

Этап 1. Задания сортируются по правилам, использующим их длительность, директивный срок выполнения и плотность [1, 3].

Этап 2. Осуществляется распределение заданий на процессоры (узлы) вычислительного кластера с целью обеспечения их равномерной загрузки, включающее построение допустимых расписаний выполнения заданий на всех процессорах, доставляющих максимум (1).

Этап 3. После распределения заданий осуществляется их выполнение с учетом методов масштабирования частоты и напряжения для одиночного процессора [2].

3. Методы распределения заданий между процессорами вычислительного кластера с однородной архитектурой

Метод 1 [1, 3].

Этап 1. Сортируем все одновременно поступившие на выполнение задания в порядке убывания их длительности (Largest Job First, LJF) [1, 3]. Если таких заданий несколько, первым располагается задание, имеющее меньший директивный срок выполнения.

Этап 2. Распределяем множество заданий, равное количеству процессоров, по методу Round Robin. Определяем коэффициент загрузки каждого процессора и время завершения выполнения задания.

Этап 3. Распределяем из оставшихся нераспределёнными количество заданий, равное количеству процессоров, согласно следующей процедуре: самое длинное задание распределяется на процессор, имеющий наименьшее текущее значение коэффициента средней загрузки; если таких процессоров несколько, выбирается тот, на котором время завершения последнего задания наименьшее. Следующее задание распределяется на процессор с большим коэффициентом средней загрузки и т. д. до тех пор, пока все задания текущего множества не будут распределены.

Этапы 2, 3 повторяются до тех пор, пока все задания входного потока не будут распределены. После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабирования частоты одиночного процессора [2].

Метод 2.

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания величины их плотности (Highest Density First, HDF). Если таких заданий несколько, первым располагается задание, имеющее большую длительность.

Этап 2. Множество заданий, равное количеству процессоров, распределяется между процессорами по методу Round Robin. Рассчитывается текущий коэффициент загрузки каждого процессора и время завершения выполнения заданий.

Этап 3. Оставшиеся нераспределёнными задания в количестве, равном количеству процессоров, распределяются согласно следующей процедуре: первым распределяется задание с наибольшей плотностью на процессор, имею-

щий наименьшее текущее значение среднего коэффициента загрузки; если таких процессоров несколько, выбирается тот, на котором время завершения последнего задания наименьшее. Следующее задание распределяется на процессор с большим коэффициентом использования и т. д. до тех пор, пока все задания текущего множества не будут распределены.

Этапы 2, 3 повторяются до тех пор, пока все задания входного потока не будут распределены. После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабирования частоты одиночного процессора [2].

Метод 3.

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания их длительности (LJF). Если таких заданий несколько, первым располагается задание, имеющее меньший директивный срок.

Этап 2. Множество заданий, равное количеству процессоров, распределяется по методу Round Robin.

Этап 2 повторяется до тех пор, пока все задания входного потока не будут распределены на процессоры.

После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабированием частоты одиночного процессора [2].

4. Методы распределения заданий между процессорами вычислительного кластера с неоднородной архитектурой

Метод 1.

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания их длительности (LJF). Если таких заданий несколько, первым располагается задание, имеющее меньший директивный срок выполнения.

Этап 2. Процессоры сортируются в порядке убывания их относительной скорости (частоты) [1, 3]; если процессоров с одинаковой скоростью несколько, то выбирается любой из них.

Этап 3. Множество заданий, равное количеству процессоров, распределяется согласно следующей процедуре: задание с большей длительностью распределяется на процессор с большей относительной скоростью (частотой). Рассчитывается коэффициент текущей средней

загрузки каждого процессора и время завершения выполнения на нем заданий.

Этап 4. Оставшихся нераспределёнными задания в количестве, равном количеству процессоров, распределяются согласно следующей процедуре: первое, то есть самое длинное задание из текущего множества, распределяется на процессор, имеющий наименьшее текущее значение среднего коэффициента загрузки; если таких процессоров несколько, выбирается тот, на котором время завершения последнего задания наименьшее. Следующее задание из множества распределяется на процессор с большим коэффициентом средней загрузки и т. д. до тех пор, пока все задания текущего множества не будут распределены.

Этапы 3, 4 повторяются до тех пор, пока все задания входного потока не будут распределены. После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабирования частоты одиночного процессора [2].

Метод 2 [1].

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания величины их плотности (HDF). Если таких заданий несколько, первым располагается задание, имеющее большую длительность.

Этап 2. Множество заданий, равное количеству процессоров, распределяется по методу Round Robin. В этом методе использован подход [4, 5], реализующий идею классификации заданий по их плотностям, отличающийся правилами определения и сортировки заданий по плотностям.

Этап 2 повторяется до тех пор, пока все задания входного потока не будут распределены на процессоры.

После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабированием частоты одиночного процессора [2].

Метод 3.

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания их длительности (LJF). Если таких заданий несколько, первым располагается задание, имеющее меньший директивный срок выполнения.

Этап 2. Процессоры сортируются в порядке убывания их относительной скорости (частоты).

Этап 3. Задания в количестве, равном количеству процессоров, распределяются в соответствии с алгоритмом Round Robin.

Этап 3 повторяется до тех пор, пока все задания входного потока не будут распределены на процессоры. После этого для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабирования частоты одиночного процессора [2].

Метод 4 [1].

Этап 1. Все задания, одновременно поступившие на выполнение, сортируются в порядке убывания их плотности (HDF). Если таких заданий несколько, первым располагается задание, имеющее большую длительность.

Этап 2. Процессоры сортируются в порядке убывания их относительной скорости (частоты).

Этап 3. Множество заданий, равное количеству процессоров, распределяется на них согласно следующей процедуре: задание с большей плотностью распределяется на процессор с большей относительной скоростью (частотой). Далее рассчитывается текущий коэффициент средней загрузки каждого процессора и время завершения выполнения на нем заданий.

Этап 4. Оставшиеся нераспределёнными задания, количество которых равно количеству процессоров, распределяется, следуя процедуре: первое из оставшихся заданий, то есть задание с наибольшей плотностью из текущего множества заданий, распределяется на процессор, имеющий наименьшее текущее значение среднего коэффициента загрузки; если таких процессоров несколько, выбирается тот, на котором время завершения последнего задания наименьшее. Следующее задание распределяется на процессор с большим коэффициентом загрузки и т. д. до тех пор, пока все задания текущего множества не будут распределены.

Этапы 3, 4 повторяются до тех пор, пока все задания входного потока не будут распределены на процессоры. Далее для каждого из процессоров строится допустимое расписание выполнения заданий с использованием методов масштабирования частоты одиночного процессора [2].

5. Экспериментальное исследование методов и анализ полученных результатов

Для проведения вычислительных экспериментов, моделирующих распределений заданий между процессорами, использующими технологии масштабирования частоты и напряжения, выбраны следующие параметры заданий: длительности заданий определялись по равномерному закону распределения в интервале [1,10], директивные сроки рассчитывались путем умножения их длительности на множитель D , распределенный по равномерному закону в интервале [1,3], в качестве процессоров для проведения моделирования использованы: для однородного кластера – Intel Core i5, для неоднородного – Intel Core i5 и Intel Xeon E5 2690. Для получения статистически достоверных данных экспериментов на каждую точку результатов расчетов генерировалось 100 наблюдений. В качестве метода масштабирования частоты одиночного процессора выбран метод, предложенный в [2], обеспечивающий наименьший коэффициент загрузки одиночного процессора с масштабированием частоты и напряжения.

Результаты моделирования рассмотренных методов распределения заданий на параллельные процессоры вычислительного кластера с однородной архитектурой приведены в табл. 1, кластера с неоднородной архитектурой – в табл. 2. Они показывают, что для кластера с однородной архитектурой наиболее предпочтительным является метод Round Robin, использующий предварительную сортировку заданий по убыванию их длительности; далее следует метод с балансировкой по текущему значению коэффициента средней загрузки, и потом – метод с предварительной сортировкой по убыванию плотности и балансировкой загрузки на основе текущего значения коэффициента загрузки. Для кластера с неоднородной архитектурой предпочтительным является метод с сортировкой заданий по убыванию плотности и балансировкой на основе текущего значения коэффициента загрузки (метод 4), далее следует метод, использующий сортировку по убыванию плотности заданий и процедуру Round Robin (метод 2), а затем с практически одинаковыми результатами располагаются методы 1 и 3.

Табл. 1. Результати моделювання методів розподілення завдань на процесори однорідного кластера

Метод	Кількість завдань	Кількість процесорів	Діапазон значень d	Діапазон значень β
Метод 1 (LJF)	500	50	0,04 – 0,27	0,90– 0,91
	500	100	0,02 – 0,28	0,90 – 0,91
	1000	100	0,04 – 0,23	0,91–0,92
	5000	100	0,02 – 0,35	0,91–0,92
Метод 2 (HDF)	500	50	0,18 – 0,52	0,88–0,91
	500	100	0,14 – 0,54	0,90–0,92
	1000	100	0,19 – 0,4	0,89–0,92
	5000	100	0,25 – 0,43	0,89–0,91
Метод 3 RR (LJF)	500	50	0,18 – 0,33	0,91–0,92
	500	100	0,16 – 0,34	0,90 –0,93
	1000	100	0,1 – 0,28	0,91–0,93
	5000	100	0,03 – 0,13	0,92–0,94

Табл. 2. Результати моделювання методів розподілення завдань на процесори неоднорідного кластера

Метод	Кількість завдань	Кількість процесорів	Діапазон значень d	Діапазон значень β
Метод 1 (LJF)	500	50	0,6 – 5,0	0,88–0,93
	500	100	0,5 – 4,3	0,90–0,93
	1 000	100	0,4 – 5,6	0,87–0,93
	5 000	100	0,6 – 4,6	0,88–0,93
Метод 2 (HDF RR)	500	50	0,5 – 5,1	0,88–0,92
	500	100	0,4 – 5	0,88–0,90
	1 000	100	0,4 – 4,8	0,89–0,90
	5 000	100	0,4 – 4,6	0,88–0,92
Метод 3 (RR)	500	50	0,6 – 5,6	0,87–0,90
	500	100	0,5 – 5,6	0,87–0,90
	1 000	100	0,4 – 5,8	0,87–0,92
	5 000	100	0,5 – 5,2	0,87–0,92
Метод 4 (HDF)	500	50	0,6 – 5	0,89–0,93
	500	100	0,4 – 4,8	0,90–0,94
	1 000	100	0,4 – 4,8	0,90–0,94
	5 000	100	0,45 – 4,85	0,90–0,95

6. Висновки

Полученные в данном исследовании результаты позволяют сформулировать следующие выводы.

1. Выбор метода распределения заданий между параллельными процессорами вычислительного кластера в значительной степени зависит от особенностей построения его архитектуры. В случае неоднородной архитектуры требуется учет характеристик заданий и процессоров, влияющих на эффективность распределения.

2. Использование величины текущего значения коэффициента загрузки процессора и сортировки по убыванию плотности позволяет получить для неоднородного кластера более высокий уровень балансировки, чем при использовании процедуры Round Robin.

3. После распределения заданий на процессоры кластера необходимо использовать методы масштабирования частоты процессора, обеспечивающие минимальную загрузку и работу в штатном режиме его эксплуатации для оптимизации энергопотребления.

Список литературы

1. Минухин С.В. Алгоритмы, программная архитектура и информационная технология моделирования методов масштабирования скоростей параллельных процессоров вычислительного кластера / С.В. Минухин, М.И. Сухонос // Вісник НТУ «ХПІ». Серія: Системний аналіз, управління та інформаційні технології. – Х.: НТУ «ХПІ». – 2013. – № 62 (1035). – С.78-92.

2. Минухин С.В. Алгоритмы оптимизации энергопотребления и повышения эффективности процессоров с масштабированием частоты и напряжения гетерогенного кластера / С.В. Минухин, М.И. Сухонос // International Conference Parallel and Distributed Computing Systems PDCS 2013 (Ukraine, Kharkiv, March 13–14, 2013). – 2013. – P. 209-217.
3. Минухин С.В. Энергоэффективные алгоритмы масштабирования скорости процессоров вычислительного кластера / С.В. Минухин // Second International Conference "Cluster Computing" CC 2013 (Ukraine, Lviv, June 3–5, 2013). – P. 131-140.
4. Bell P.C. Multiprocessor Speed Scaling for Jobs with Arbitrary Sizes and Deadlines / Paul C. Bell, Prudence W. H. Wong // Theory and Applications of Models of Computation. Lecture Notes in Computer Science. – 2011. – Vol. 6648. – P. 27-36.
5. Albers S. Speed scaling on parallel processors / S. Albers, F. Müller, S. Schmelzer // Algorithmica. – 2014. – Vol. 68. – Issue 2. – P. 404-425.