

ЭНЕРГОЭФФЕКТИВНЫЙ АЛГОРИТМ ОБРАБОТКИ ЗАДАЧ В РАСПРЕДЕЛЕННЫХ СИСТЕМАХ

Глоба Л.С., Косовненко Р.А., Степурич А.В.

Институт телекоммуникационных систем

КПИ им. Игоря Сикорского, Украина

E-mail: romankosovnenko@gmail.com

Energy efficient task processing algorithm in distributed systems

We consider the results of experiments and simulations on the study of the energy efficiency of task distribution algorithms in distributed systems. A new algorithm for distribution of tasks aimed at optimizing energy consumption is proposed.

В данной работе рассмотрены результаты экспериментов и моделирования по исследованию энергоэффективности алгоритмов распределения задач в распределенных системах. Предложен новый алгоритм распределения задач, направленный на оптимизацию энергопотребления.

Глобальные вычислительные сети Grid [1] были предложены в качестве новой парадигмы для решения крупномасштабных вычислительных задач в науке, технике и бизнесе [2]. Они дают возможность одновременного использования миллионов вычислительных ресурсов [3], принадлежащих различным организациям и расположенных в различных административных областях. Системы Grid объединяют разнородные вычислительные ресурсы (персональные компьютеры, рабочие станции, кластеры, суперкомпьютеры), используя разные стратегии доступа к ним, выполняя различные приложения (научные, инженерные и коммерческие), предъявляющие к системе различные требования. Ресурсы принадлежат различным организациям, имеющим свои правила управления ресурсами, их использования и определения их стоимости для различных пользователей в различное время. Доступность и загруженность ресурсов также может динамически изменяться во времени.

В окружении Grid владельцы и потребители ресурсов имеют различные цели, используют различные стратегии и экономические схемы. С другой стороны, нет стратегии позволяющей достигнуть аналогичного быстродействия при более низком энергопотреблении. Таким образом, актуальной проблемой является разработка алгоритма распределения задач, нацеленного на оптимизацию энергопотребления системы в целом.

Для разработки алгоритма было проведено ряд практических экспериментов, которые моделировались. И результаты моделирования этих экспериментов сравнивались с экспериментальными результатами, а также с результатами моделирования с помощью различных алгоритмов распределения задач (первый свободный, раунд робин). Во время проведения эксперимента были снимались показатели напряжения, потребляемого тока, мощности при различной нагрузке нод.

По результатам эксперимента был спроектирован прототип алгоритма распределения задач с учетом энергоэффективности используемых ресурсов в системе Grid, а также несколько модификаций. Результаты моделирования представлены в табл. 1.

Algorithm	Time of processing (same tasks) (s)	Power consumed (W)	Gain (in comparison with "First available") (%)		Gain (in comparison with Round Robin) (%)	
			in time	in power consumed	in time	in power consumed
"First available"	16,79	380,69	0%	0%	1%	5%
Round Robin	17,02	401,81	-1%	-6%	0%	0%
Proposed	16,17	369,65	4%	3%	5%	8%
Proposed + tasks typification	15,55	366,97	7%	4%	9%	9%
Proposed + sleeping mode	16,17	319,64	4%	16%	5%	20%

Табл.1. Результаты моделирования алгоритмов распределения задач и их потребляемая мощность при равных задачах и ресурсах системы.

Проанализировав таблицу, можно сделать вывод, что использование предложенного алгоритма может снизить энергопотребление до 20%.

Входными параметрами предложенного алгоритма является набор вычислительных нод, которые подключены к брокеру ресурсов, каждая из нод описывается следующими параметрами: количество доступной оперативной памяти, количество ядер процессора, производительность CPU, функция, описывающая энергопотребление ноды от CPU.

Каждая задача, поступающая в очередь в определенный момент времени может быть описана следующим набором параметров;

- необходимое количество оперативной памяти;
- необходимое количество ядер для выполнения задачи;
- максимальное время для выполнения задачи.

Предложенный алгоритм словесно можно описать в виде последовательности шагов, которые нужно выполнить:

1. Определить состояние системы в момент времени τ_{k-1} .
2. Отфильтровать ноды по количеству оперативной памяти и по количеству доступных ядер, необходимых для выполнения задачи.
3. Рассчитать набор суммарных энергопотреблений при выполнении задачи на каждой ноды отдельно.

4. Отсортировать ноды по увеличению вероятной потребляемой мощности и сформировать последовательность нод в порядке от меньшей потребляемой мощности к большей

5. Отсортировать ноды по увеличению производительности и сформировать последовательность нод в порядке от большей производительности к меньшей.

6. Определить оптимальное сочетание мощность/производительность с помощью пересечения двух последовательностей.

7. Отправить задачу на выполнение.

Работа первого прототипа была экспериментально подтверждена на реальной физической системе.

Итоги работы прототипа были зафиксированы аналогичным экспериментом. Полученные показатели сравнили с результатами моделирования, что показало правдивость предположений.

Результаты экспериментов показывают, что предложенный алгоритм может производить эффективное распределение при возможной незначительной потере производительности, но с большим выигрышем в потребляемой энергии, как следствие это делает предложенный алгоритм экономически более выгодным. Для повышения эффективности работы алгоритма предлагается использовать различные модификации данного алгоритма, которые учитывают характеристики потока задач, их классификацию, а также более детально используют характеристики нодов. Разработка таких алгоритмов является предметом дальнейших исследований.

Література

1. P. Brucker Scheduling Algorithms, Springer Ferlag 1998, pp 217-218.
2. L. Hluchy, V.D. Tran, D. Froehlich, and W. Castaings, Methods and Experiences of Parallelizing Flood Models, The 10th EuroPVM/MPI conference. LNCS 2840. Sept. 2003, Venice. pp. 677-681.
3. B.S. Baker, D.J. Brown and H.P. Katseff, A $5/4$ algorithm for two-dimensional packing, J. of Algorithms, 1981, v. 2, pp. 348-368.
4. K. Jansen, Scheduling malleable parallel tasks: an asymptotic fully polynomial-time approximation scheme, Proc. European Symposium on Algorithms, ESA, 2002.
5. S. Orlando, P. Palmerini, R. Perego, F. Silvestri, Scheduling high performance data mining tasks on a data Grid environment, Euro-Par 2002, LNCS 2400, Springer-Ferlag Berlin Heidelberg 2002, pp. 375-384.
6. I. Foster, C. Kesselman, S. Tuecke, Intl J. Supercomputer Applications, 2001.
7. Dinda, P. and O'Hallaron, D., An Evaluation of Linear Models for Host Load Prediction. In Proc. 8th IEEE Symposium on High-Performance Distributed Computing, 1999, IEEE Press.
8. Berman, F., Wolski, R., Figueira, S., Schopf, J. and Shao, G. Application-Level Scheduling on Distributed Heterogeneous Networks. In Proc. Supercomputing '96, 1996.