Энергоэффективные вычисления для группы кластеров[1]

Д.А. Грушин, Н.Н. Кузюрин
grushin@ispras.ru, nnkuz@ispras.ru

Аннотация. Рассмотрена проблема балансировки нагрузки для множества параллельных задач на группе географически распределенных кластеров уменьшающая количество энергии при вычислениях. Предложены несколько алгоритмов распределения задач и проведена экспериментальная проверка их эффективности.

Ключевые слова: энергоэффективность; балансировка нагрузки; распределенные вычисления

1. Введение

В последние годы во всем мире происходит значительный рост потребности в вычислительных ресурсах. Если раньше суперкомпьютеры были крайне дороги и доступны единицам, то с появлением вычислительных кластеров, собранных из общедоступных компонентов, наука и промышленность получили в своё распоряжение простой и недорогой способ использования высокопроизводительных вычислений.

Типичный вычислительный кластер (Beowulf кластер[2]) состоит из широко распространённого аппаратного обеспечения и работает под управлением операционной системы GNU/Linux или FreeBSD. Если кластер предназначен для использования многими пользователями, то управление кластером осуществляет менеджер ресурсов. Пользователи отправляют свои задания менеджеру, который ставит их в очередь и, по мере высвобождения вычислительных узлов, осуществляет запуск заданий.

[1] Выполнено при финансовой поддержке Минобрнауки РФ, контракт 07.514.11.4001

[2] Одна из типичных конфигураций -- набор компьютеров, собранных из общедоступных компонентов, с установленной на них операционной системой Linux, и связанных сетью Ethernet, Myrinet, InfiniBand или другими относительно недорогими сетями. Такую систему принято называть кластером Beowulf.

От количества поступающих задач зависит сколько узлов кластера будет занято выполнением задач, а сколько простаивать. Согласно статистике большинство кластеров испытывает периодическую нагрузку – когда интенсивность потока задач различается в несколько раз в разное время суток. Это означает, что даже при относительно плотной загрузке заданиями в среднем, существуют периоды, когда большая часть узлов кластера не выполняет заданий и простаивает.

В работе [1] мы показали, что временный перевод простаивающих узлов кластера в спящий режим приводит к существенной экономии электроэнергии. В данной работе мы хотим рассмотреть возможность снижения расхода электроэнергии для группы кластеров, находящихся под управлением одного менеджера ресурсов – брокера, который, получая поток заданий от пользователей, распределяет их между кластерами. В такой системе существует несколько возможностей для экономии электроэнергии (как в количественном смысле, так и в денежном -- снижая стоимость энергии):

- Различная энергоэффективность (отношение производительности к энергопотреблению) кластеров;
- Географическое положение кластеров. Стоимость энергии в разных регионах может существенно различаться. Отправляя задачи на кластер, с более низкой ценой на электроэнергию, можно уменьшить общую стоимость энергии. Стоимость энергии различается также в разное время суток, что даёт дополнительные возможности выбора если кластеры находятся в разных часовых поясах.

В данной статье мы оцениваем с помощью моделирования насколько возможно снизить общее количество энергии и её стоимость в подобной вычислительной системе. При этом задача снижения энергопотребления группы кластеров должна рассматриваться совместно с общей проблемой повышения эффективности использования вычислительных ресурсов. Мы считаем, что её исследование позволит обеспечить в будущем значительную экономию энергопотребления.

2. Оптимизация энергопотребления одного кластера

Следует отметить, что в литературе рассматриваются разные модели и постановки задач энергосберегающих вычислений [2]. Наибольшее количество работ посвящено задаче оптимизации энергопотребления вычислительной системы, состоящей из одного процессора [2,3,4,5,6,7]. При этом основным способом уменьшения затрачиваемой энергии является выключение простаивающих компонент вычислительной системы и их последующее включение по мере надобности. Дополнительными способами могут быть:

- Перераспределение вычислительных заданий по времени при условии наличия многотарифной схемы оплаты электроэнергии (например день-ночь). Тогда за счёт повышения загрузки системы ночью, днем вычислительная нагрузка будет снижена и простаивающие компоненты вычислительной системы отключены. Общее количество потреблённой электроэнергии не снижается, однако уменьшается её стоимость, что также рассматривается как повышение энергоэффективности;
- Программное управление производительностью компонент вычислительной системы. Современные процессоры и оперативная память имеют возможность динамически изменять свою частоту и рабочее напряжение. Такой механизм носит название DVS (dynamic voltage and frequency scaling). Основной принцип данного механизма заключается в том, что при понижении напряжения процессора время вычислений увеличивается, однако общее количество энергии потраченной на вычисления уменьшается.

Важный аспект динамического управления энергопотреблением состоит в том, что смена состояния системы (включение, изменение производительности и т.п.) имеет стоимость, выраженную в дополнительном количестве потреблённой энергии, задержки или потери производительности, что, вообще говоря, не гарантирует снижения энергозатрат при переводе системы в спящий режим в отсутствие работы и обратно по мере надобности. Для того, чтобы компенсировать данные потери энергии, система должна находиться в спящем состоянии не менее определённого промежутка времени. Такой промежуток времени называется ``минимальным временем сна'' (таблица 1).

Так, не более чем в 2 раза худший результат по сравнению с оптимальным по критерию минимизации затраченной энергии гарантируется, если отключать устройство через время $\tau = T_{be}$ [8]. И не более чем в $e/(e-1) = 1.58$ раза, если выключать устройство через время t с вероятностью $p_t = Ke^{t/E_{wu}}$, где $K = 1/(E_{wu}(e-1))$ [8], что является наилучшей возможной оценкой в классе онлайновых вероятностных алгоритмов [2].

Величина	**Значение**
E_{wu}	количество энергии, расходуемое при включении
E_{sd}	количество энергии, расходуемое при выключении
T_{be}	минимальное время сна, компенсирующее потери энергии от включения и выключения (break-even time)
T_{wu}	wakeup delay --- задержка при включении
τ	время простоя

Таблица 1. Основные обозначения

Рассматривая задачу отключения/включения узлов вычислительного кластера следует отметить, что задача является многокритериальной, где основным критерием является сэкономленная энергия, а вспомогательными время ожидания и число включений узлов. При этом важно учитывать следующие особенности:

- Каждая многопроцессорная задача требует одновременного включения нескольких узлов.
- Каждый многопроцессорный узел может выполнять одновременно несколько задач. Выключить узел можно только тогда, когда все его процессоры простаивают.

Из проведённых нами измерений [1] видно, что в состоянии простоя узел кластера потребляет в 12 раз больше энергии чем в выключенном состоянии. Моделирование показывает, что использование простого алгоритма, отключающего узлы существенно снижает расход энергии.

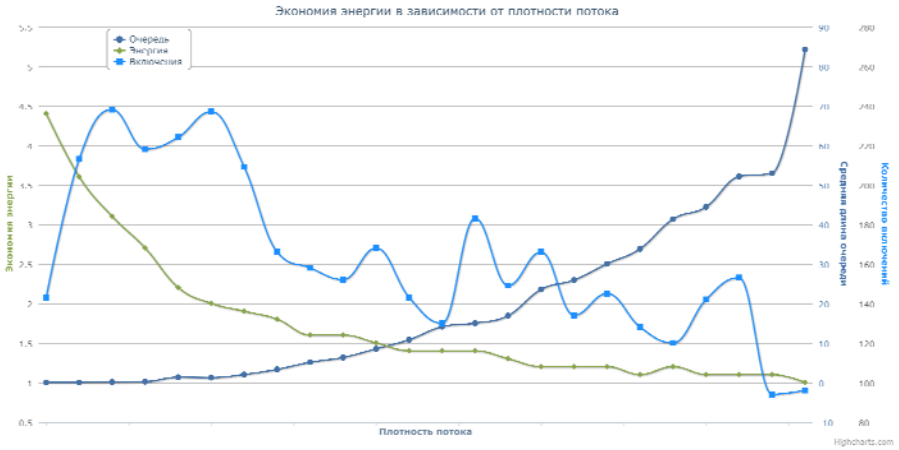

Рисунок 1. Результаты моделирования для одного кластера

На рисунке 1 показаны результаты моделирования для одного кластера. Мы сравнивали количество потраченной энергии, среднюю длину очереди, количество включений узлов для одного кластера в двух случаях: без отключения простаивающих узлов и с отключением. При этом проводилась серия испытаний, в каждом последующем плотность потока задач увеличивалась.

Эксперименты проводились для потоков с различным соотношением 1-процессорных и многопроцессорных задач, но существенного влияния на полученные результаты это не оказало, так как мы не использовали в модели узлы с несколькими процессорами.

Результаты показывают, что при очень слабых потоках (примерно 10 задач в день в нашем эксперименте) отключение узлов снижает потребление энергии в 4-5 раз. При меньшем количестве задач эта величина будет ещё больше. Средние значения находятся в середине графика. Потребление энергии в данном случае сокращается на 20-80% в зависимости от интенсивности потока задач.

3. Оптимизация энергопотребления группы кластеров

Возникающая задача управления ресурсами групп кластеров является многокритериальной и затрагивает много различных факторов.

Рассмотрим типичную двухуровневую схему где центральный планировщик (брокер ресурсов) решает на какой кластер направить поступающую к нему задачу.

В распределенной вычислительной системе кластеры располагаются в различных географических регионах. Стоимость энергии в разных регионах отличается, и в каждом отдельном регионе изменяется в течении суток. Следовательно, в каждый момент времени брокер имеет возможность отправить поступившую к нему задачу на кластер с минимальной стоимостью энергии. Мы провели серию экспериментов для того, чтобы оценить насколько возможно сократить стоимость затраченной на вычисления энергии в такой системе.

Для создания экспериментальной модели были выбраны несколько кластеров из списка TOP500 за 2011 год [9]. Для выбранных кластеров мы взяли из списка значения энергоэффективности и размера, минимальное значение каждого параметра было принято за единицу.

Для упрощения модели мы предположили, что все кластеры имеют одинаковую производительность вычислительных узлов. Базовую величину пикового энергопотребления одного узла кластера мы предположили равной 520 Вт, исходя из данных, полученных нами ранее для узла кластера установленного в ИСП РАН (HP ProLiant DL380 G3) [1]. Значения энергопотребления в состояниях простоя и сна составили 0,43 и 0,03 от пикового соответственно. Величина энергопотребления для каждого узла вычислялась как произведение базового энергопотребления и коэффициента энергоэффективности кластера.

Список кластеров представлен в таблице 2. Стоимость электроэнергии в представленных странах была взята из публично доступных данных [10].

Страна, часовой пояс (GMT)	**Год установки**	**Относи-тельный размер**	**Относительная энерго-эффективность (Mflop/ Watt)**	**Стоимость энергии (Цент США за 1kWh)**
Spain, +2	2011	1.0	1.9	19.69
United Kingdom, 0	2011	1.0	1.7	18.59
United States, -7	2011	1.8	1.4	11.2
Canada, -4	2011	2.1	1.5	6.18
Australia, +8	2011	2.4	1.2	28.88

United States, -5	2011	4.2	8.4	11.2
Russia, +3	2011	8.4	1.0	9.49
Saudi Arabia, +3	2009	16.7	1.6	13.1
Japan, +9	2010	18.7	3.6	12
China, +7	2010	30.8	2.1	16

Таблица 2. Набор кластеров для моделирования

В моделируемой системе задачи поступали в очередь брокера и затем распределялись по кластерам. Для потока задач, поступающих на единственный кластер, характерными особенностями являются переменная интенсивность поступления задач и периодичность. В среднем вероятность появления задачи в ночное время суток в два раза меньше, чем в дневное. Для брокера, распределяющего задачи между большим количеством кластеров, находящихся в разных часовых поясах, мы считаем, характерна равномерная интенсивность потока.

Для моделируемой системы мы выбрали поток задач, характеристики которого представленны в таблице 3.

Минимальная и максимальная ширина одной задачи	1 -- 10
Минимальная и максимальная длительность одной задачи	10 мин -- 2 часа
Доля однопроцессорных задач	0.8

Таблица 3. Характеристики потока задач

Приведем далее результаты моделирования для двух случаев. В первом случае брокер распределяет задачи между кластерами не учитывая стоимость энергии, используя алгоритм S. Во втором случае стоимость энергии учитывается -- алгоритм P. Оценка эффективности распределения проводилась по следующим критериям:

- Минимизация общей стоимости затраченной энергии;
- Минимизация среднего времени ожидания задачи в очереди.

Алгоритм S. Выбирается кластер с минимальным отношением общей площади задач в очереди к ширине кластера -- $h_j = (\sum_{k=1}^{N} S_k)/W_j$, где N -- число задач, стоящих в очереди, S_k -- площадь задачи, W_j -- число узлов кластера. Данную величину можно рассматривать как оценку времени пребывания задачи в очереди -- время, через которое задача, поступившая в очередь, будет запущена.

Алгоритм P. Входным параметром алгоритма является значение H_{diff}, которое определяет допустимую разницу между минимальным и максимальным временем ожидания в очередях кластеров при которой алгоритм будет экономить энергию -- алгоритм будет выбирать кластер, который выполнит задачу с минимальным расходом энергии.

Для каждого кластера C_j брокер определяет стоимость электроэнергии в данный момент времени и значение $h_j = (\sum_{k=1}^{N} S_k)/W_j$ -- отношение общей площади задач в очереди к ширине кластера (оценка времени пребывания задачи в очереди). Если $max(h_j) - min(h_j) > H_{diff}$, то задача отправляется на кластер с минимальным значением h_j. В противном случае задача отправляется на кластер с минимальной стоимостью энергии.

На рисунке 2 показана зависимость между плотностью потока, долей однопроцессорных задач в потоке и отношением значений реального времени ожидания к величине $h_j = (\sum_{k=1}^{N} S_k)/W_j$ (оценка времени пребывания задачи в очереди). Значения получены моделированием работы одного кластера, на котором применялся алгоритм распределения задач Backfill **Ошибка! Источник ссылки не найден.**. Плотность потока и доля однопроцессорных задач в потоке постепенно увеличивались. Результаты показали, что оценка h_j отличается от реального времени ожидания задачи в очереди не более чем на 30%. При этом большая часть значений h_j попадает в диапазон $[0.8,1]$. Эти значения соответствуют потокам со средней долей однопроцессорных задач -- $[0.3,0.6]$.

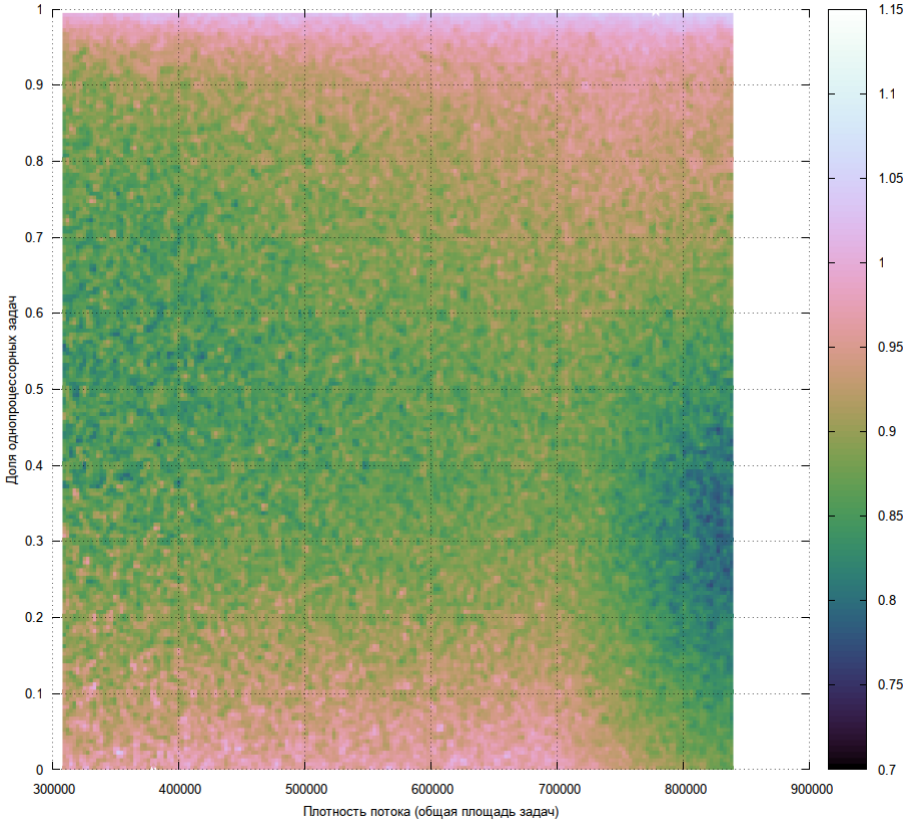

Рисунок 2. Оценка времени ожидания -- отношение значений реального времени ожидания к величине $h_j = (\sum_{k=1}^{N} S_k)/W_j$. *Результаты показали, что оценка* h_j *отличается от реального времени ожидания задачи в очереди не более чем на 30%. При этом большая часть значений* h_j *попадает в диапазон* $[0.8,1]$. *Эти значения соответствуют потокам со средней долей однопроцессорных задач --* $[0.3,0.6]$.

4. Эксперименты

Для каждого из двух вариантов алгоритма распределения задач брокером: S и P мы провели серию экспериментов с описанной выше группой кластеров (таблица 2). В обеих сериях использовались одинаковые потоки задач. В

одной серии использовался поток с постоянными характеристиками (таблица 3), но различной интенсивности -- начиная с небольшой плотности задач (500 задач в сутки) и затем каждый последующий раз увеличивая плотность на равное количество задач. В каждой серии использовался также второй параметр -- H_{diff}, который увеличивался с шагом 5 минут. На каждом кластере простаивающие узлы отключались. В ходе экспериментов измерялось среднее время ожидания задач в очередях кластеров и общая стоимость затраченной энергии. Далее будем обозначать две серии S и P соответственно.

На рисунке 3 показана разница между средним временем ожидания в серии P и S и отношение стоимости затраченной энергии в серии S к P.

Эксперименты показали, что время ожидания в серии P всегда увеличивается. Обратим внимание на отмеченные на рисунке области: 1, 2, 3. Наибольшее увеличение времени ожидания наблюдается в области 1 при плотности потока от 2000 до 5000 задач в сутки и значениях H_{diff} 7000-9000 секунд. Максимальное увеличение времени ожидания составило 2659 секунд при значениях плотности и H_{diff} 3140 и 9000 соответственно.

Минимальное увеличение времени ожидания составляет не более 300 секунд и наблюдается в области 3 при наибольшей плотности потока и наименьших значениях H_{diff}.

Область средних значений отмечена номером 2. Максимальное увеличение времени ожидания в данном случае составляет около 800 секунд при значении H_{diff} 4000.

Изменение стоимости затраченной энергии показано на второй части рисунка 3. Наибольшая экономия достигается на потоках с самой маленькой плотностью в рамках данного эксперимента (500 задач в сутки) и значениях H_{diff} 1000 и выше. При этом, по мере увеличения значения H_{diff}, уже после 2000 стоимость затраченной энергии не изменяется. Максимальное значение составило 1.87 при значениях плотности и H_{diff} 620 и 8700 соответственно.

Таким образом, в данном эксперименте значения входного параметра H_{diff} в интервале от 2000 до 4000 дают наилучшее соотношение между стоимостью затраченной энергии и временем ожидания.

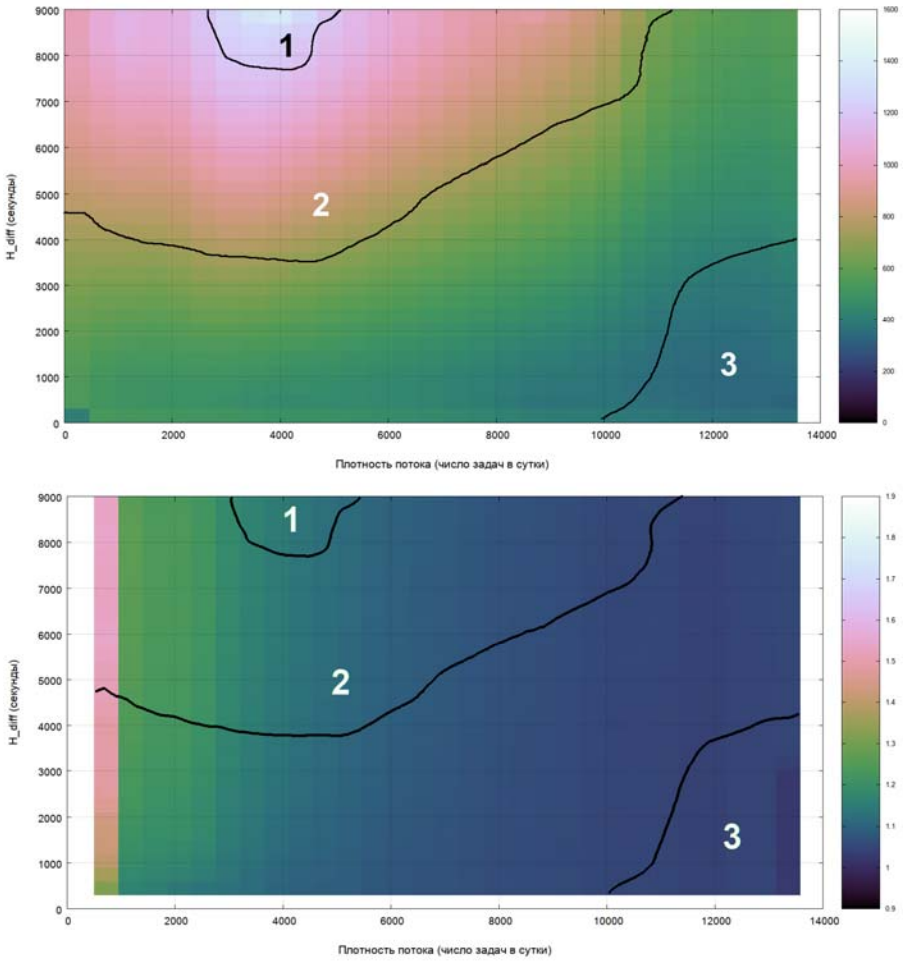

Рисунок 3. Увеличение среднего времени ожидания в серии P по отношению к серии S и отношение стоимости затраченной энергии в серии S к P

5. Заключение

В статье была рассмотрена задача снижения стоимости энергии вычислительной системы, состоящей из нескольких географически распределенных кластеров. В такой системе существует несколько возможностей для снижения стоимости энергии. Это различная

энергоэффективность кластеров и различная стоимость энергии в зависимости от географического положения кластера и времени суток.

Мы провели моделирование вычислительной системы, состоящей из 10 кластеров (данные энергоэффективности и размера были взяты из списка TOP500 за 2011 год), и сравнили результат работы двух алгоритмов --- S и P. Алгоритм S распределял задачи не учитывая расход энергии, а алгоритм P, в зависимости от значений входного параметра H_{diff}, в определенных случаях направлял задачу на кластер с минимальной стоимостью энергии.

Результаты проведенного эксперимета показали, что стоимость энергии возможно снизить, однако при этом увеличивается среднее время ожидания в очереди. Конкретные значения зависят от плотности потока задач --- чем больше плотность, тем меньше возможностей для выбора кластера, и тем меньше величина экономии. В нашем эксперименте наибольшая величина экономии составила 50%. Это соответствует потокам с минимальной плотностью --- около 100 задач в сутки. Для потоков со средней плотнотью (около 2-4 тыс задач в сутки) величина экономии составила 20-15%. Для потоков с большой плотностью (от 10 тыс задач в сутки) экономия составляет не более 3%.

Таким образом, мы считаем, что использование информации о стоимости энергии каждого кластера при распределении задач способно существенно снизить расходы на электроэнергию для владельца распределенной вычислительной системы. Однако, стоит отметить, что в реальной жизни такое перераспределение задач не всегда возможно. Обычно дата-центры строятся для обслуживания пользователей в определенном регионе, чтобы уменьшить ``время отклика". Также задачи могут работать с большими объемами локальных данных, которые не так просто переместить с одного кластера на другой. Несмотря на это многие компании уже используют как альтернативные источники энергии для питания своих дата-центров, так и особенности географического расположения для снижения расходов на электроэнергию [12,13]. Так, известная интернет компания Facebook планирует запустить в 2012 году дата-центр на севере Швеции в городе Лулео (Luleå), удалённом почти на тысячу километров к северу от Стокгольма и находящегося в 100 километрах от Полярного круга. В 2009 году компания Google приобрела здание бумажного комбината в Финляндии в городе Хамина и переоборудовала его в дата-центр, где для охлаждения используется вода из Балтийского моря, что также снижает расходы на электроэнергию. Перечисленные факты говорят о важности энергоэффективности в современных распределенных высокопроизводительных вычислительных системах и тенденции учета географического расположения для снижения стоимости вычислений.

Список литературы

[1] Иванников В.П., Грушин Д.А., Кузюрин Н.Н. и др. Программная система увеличения энергоэффективности вычислительного кластера // Программирование. — 2010. — Т. 6. — С. 28–40.

[2] Albers S. Algorithms for Energy Saving // Efficient Algorithms: Essays Dedicated to Kurt Mehlhorn on the Occasion of His 60th Birthday. — 2009. — P. 173–186.

[3] S. Albers, H. Fujiwara. Energy-Efficient Algorithms for Flow Time Minimization // Lecture Notes in Computer Science. — 2006. — Vol. 3884. — P. 621–633.

[4] Augustine J, Irani S, Swamy C. Optimal power-down strategies // SIAM Journal on Computing. — 2008. — Vol. 37. — P. 1499–1516.

[5] Irani S, Shukla S K, Gupta R. Algorithms for power savings // ACM Transactions on Algorithms. — 2007. — Vol. 3.

[6] Irani, Pruhs. Algorithmic problems in power management // SIGACT News. — 2005. — Vol. 36, no. 2. — P. 63–76.

[7] Zhang, Chatha. Approximation algorithm for the temperature-aware scheduling problem // ICCAD ’07: Proceedings of the 2007 IEEE/ACM international conference on Computer-aided design. — Piscataway, NJ, USA: IEEE Press, 2007. — P. 281–288.

[8] A Karlin, M Manasse, L McGeoch, S Owicki. Randomized competitive algorithms for nonuniform problems // ACM-SIAM Symposium on Discrete Algorithms. — 1990. — P. 301–309.

[9] Top500 supercomputer sites. — 2011. — November. — www.top500.org.

[10] Energy price statistics. — 2011. — November. — http://epp.eurostat.ec.europa.eu.

[11] David Jackson, Quinn Snell, Mark Clement. Core Algorithms of the Maui Scheduler // Job Scheduling Strategies for Parallel Processing / Ed. by D. Feitelson, L. Rudolph. — Springer Berlin / Heidelberg, 2001. — Vol. 2221 of Lecture Notes in Computer Science. — P. 87–102.

[12] Yevgeniy Sverdlik. Microsoft gets wind power for Dublin data center // http://www.datacenterdynamics.com. — 2011.

[13] Ward Van Heddeghem, Willem Vereeckena, Didier Collea et al. Distributed computing for carbon footprint reduction by exploiting low-footprint energy availability // Future Generation Computer Systems. — 2012. — Vol. 28. — P. 405– 414.