

Р. М. Ганопольский, Д. Б. Кепещук

**Результаты исследования
эффективности grid-системы
Тюменского Государственного Университета**

Рассматриваются проблемы создания grid-системы на базе распределенной компьютерной сети Тюменского государственного университета. Описываются конфигурация и результаты эксперимента по созданию тестовой grid-системы.

1. Состояние проблемы

Исследования, проводимые студентами, аспирантами и научными сотрудниками вузов, часто требуют больших вычислительных мощностей, выходящих за пределы возможностей имеющихся персональных компьютеров и серверов. Для решения вычислительноемких задач за ограниченное время необходима производительность, которую могут обеспечить только мощные суперкомпьютеры. В Тюменском государственном университете это следующие задачи:

- моделирование физических процессов с большим количеством первоначальных условий и набор статистики по ним, например моделирование молекулярных или решеточных структур;
 - гидрогазодинамические расчеты нефтяного и газового пластов;
 - расчеты химических свойств органических веществ.

В настоящее время существует несколько видов суперкомпьютеров: массивно-параллельные, векторно-конвейерные, симметричные мультипроцессорные, кластерные и др. Приобретение суперкомпьютера для решения перечисленных задач оказалось бы слишком затратным (суперкомпьютеры первых трех видов выпускаются в очень малых количествах и имеют высокую цену, сборка кластерного несколько дешевле, но требует большого числа дополнительных компьютеров (узлов) и отдельного помещения). Авторами статьи был предложен выход из сложившейся ситуации — более полное использование вычислительных ресурсов существующего парка компьютеров, т. е. создание grid-системы.

«Grid — географически распределенная вычислительная платформа, состоящая из множества машин разных типов, доступ к которым пользователи могут получать через единый интерфейс. Иными словами, grid предлагает технологию доступа к общим ресурсам и службам в рамках распределенных виртуальных организаций, состоящих из предприятий и отдельных лиц, которые могут совместно использовать ресурсы» [7]. Приложения, подходящие для grid-вычислений, должны удовлетворять следующим критериям [12]:

1. Вычислительная емкость. Приложение предназначено для вычислений, а не для передачи данных. Исключения составляют лишь программы, анализирующие данные в пределах одного узла сети, т. е. заранее подготовленные.
2. Возможность распараллеливания вычислений. Не все задачи подходят для распараллеливания. Например, обычно не подходят задачи, использующие на каждом шаге вычислений значения предыдущего шага.

Grid-система строится по принципу «клиент-сервер» и состоит из одного или нескольких компьютеров-серверов и множества компьютеров-клиентов свободной конфигурации. Компьютеры-клиенты занимаются вычислениями. Функции компьютера-сервера заключаются в выделении каждому из клиентов части вычислений, приеме и агрегировании результатов. Причем серверу неизвестно, какое количество клиентов будет доступно в определенный момент времени: задания выделяются по запросам компьютеров-клиентов.

В отличие от вычислительных кластеров, вычислительные узлы grid могут быть недоступны серверу по сети. Основное требование — доступность сервера всем вычислительным узлам. Они могут связываться с сервером с помощью сетевых протоколов среднего уровня (TCP/IP) и высокоуровневых протоколов (HTTP, SOAP и даже e-mail). Это дает возможность использования вычислительных узлов без постоянного подключения к сети (например, посредством dialup), а также «закрытых» с помощью браузера офисных компьютеров.

Создано уже достаточно большое количество grid-систем, объединяющих до нескольких сотен тысяч компьютеров, как обыкновенных офисных, так и специализированных серверов. Такие системы используются для решения научных задач и в любительских целях. Мощных вычислительных ресурсов требуют grand challenges [6] — фундаментальные научные или инженерные задачи, имеющие широкое применение, а также задачи прикладного характера, например: выяснение природы и поведения черных дыр, исследование гравитационных волн, моделирование поведения далеких планет университетской сетью рабочих станций, изучение процессов формирования облаков газа во вселенной, первых звезд и галактик, исследования в области генетики, медицины и разработки лекарственных препаратов.

Решать подобные задачи возможно с привлечением не только дорогостоящих суперкомпьютеров, но и grid-систем, которые можно реализовать без больших расходов, «на месте». В связи с этим количество grid-проектов каждый год удваивается [2, 4]. Самые известные из них [4]:

- EGEE (Enabling Grids for E-science in Europe) — создание общеевропейской инфраструктуры, основанной на grid-технологиях;

- РДИГ — региональная федерация в рамках проекта EGEE;
- Grid2003 — крупнейший в мире, объединяющий ряд мощных вычислительных центров США и функционирующий в непрерывном режиме с ноября 2003 г.

Grid-сообщества:

- Global Grid Forum — общественная организация, включающая более 5 тыс. специалистов, работающих в области распределенного компьютеринга и технологий grid.
- Globus Alliance — организация, разрабатывающая фундаментальные технологии, необходимые для построения вычислительного grid. Принадлежащий ей проект Globus Toolkit играет ведущую роль в формировании концепции grid и создании базовых программных средств.

Кроме того, в ряде проектов фундаментальные задачи решаются с использованием вычислительных ресурсов энтузиастов, чьи компьютеры подключены к grid-системам. Наиболее известны следующие проекты [5]:

- distributed.net — взламывание шифра RC5 и поиск Оптимальных Линеек Голомба;
- ZetaGrid — проверка гипотезы Римана — одной из важнейших задач современной математики;
- SETI@home — поиск в космическом шуме сигналов внеземных цивилизаций;
- Predictor@home — предсказывание структуры белка по последовательности атомов.

В Тюменском государственном университете насчитывается более 1000 компьютеров, объединенных локальной сетью Fast Ethernet (100 Мбит/с). Основная сфера применения парка компьютеров — образование (установлены главным образом в компьютерных классах). Естественно, что при этом большая часть процессорных и дисковых ресурсов не используется.

Использование свободных ресурсов части ЭВМ ТюмГУ для grid-системы позволит создать производительный и в то же время недорогой суперкомпьютер. Преимущество grid-технологий перед классическими кластерами в данном случае очевидно: компьютеры занимаются вычислениями даже во время эксплуатации пользователями, вместе с тем для решения задачи редко требуется круглосуточная работа всех компьютеров — вычисления на каждом узле приостанавливаются, пока компьютер выключен, это не мешает продолжать вычисления другим узлам.

Целью эксперимента было испытание и демонстрация возможностей полученной вычислительной системы.

2. Конфигурация экспериментальной системы

В качестве средств технической реализации было выбрано программное обеспечение класса middleware для построения grid — Fida. Данный продукт бесплатен, поддерживает многие распространенные операционные системы (Windows, Linux, FreeBSD, Unix), написан полностью на языке C и компилируется непосредственно под конечные платформы, что положительно сказывается на производительности и простоте установки. Поставляется в исходных кодах. Еще одним значительным для экспериментов преимуществом является простота освоения — уже через полчаса с начала изучения можно написать и ввести в действие несложное распределенное приложение.

3. Описание эксперимента

Для исследования возможностей масштабирования полученной grid-системы авторами статьи был проведен ряд экспериментов. Для расчетов была выбрана несложная задача: подсчет количества простых чисел в заданном диапазоне целых положительных чисел $[0, M]$. Задача узла-сервера в этом случае сводится к делению всего диапазона чисел на некоторое количество поддиапазонов (называемых далее также вычислительными пакетами) и последовательной передаче их по запросам клиентов, а также сбору ответов (количество простых чисел в указанном диапазоне) и их суммированию (результат вычислений). Задача узла-клиента заключается в получении диапазона для обработки, подсчете количества простых чисел в указанном диапазоне (методом перебора) и передаче ответа серверу. После обработки первого диапазона узел-клиент получает следующий (из диапазонов чисел, которые сервер еще не передавал другим клиентам) и так далее до тех пор, пока все поддиапазоны не будут обработаны.

Ускорение достигается за счет параллельной обработки вычислительных пакетов узлами-клиентами. При разделении диапазона следует учитывать вычислительный объем каждого из поддиапазонов. Мы постарались делить его на как можно более однородные в этом отношении поддиапазоны.

Данная задача последовательно решалась на увеличивающемся с каждым прогоном числе узлов от 1 до 12. Хотя по конфигурации узлы различались незначительно (компьютеры с процессором Intel Celeron/Pentium 4 с частотой 2,2–2,4 ГГц), следует учитывать, что в разные моменты времени они были загружены неодинаково, поэтому неизбежна некоторая погрешность в подсчете времени решения задачи.

При увеличении количества узлов от 1 до 12 число поддиапазонов, на которые делился диапазон чисел, выбиралось равным количеству узлов-клиентов. Далее количество поддиапазонов последовательно

увеличивалось с 13 до 39, т. е. число вычислительных пакетов превышало количество вычислительных узлов. Это было сделано для проверки авторской гипотезы о том, что значительную часть времени решения задачи система находится в состоянии, когда более мощные или свободные узлы уже обработали свои поддиапазоны, а более слабые или занятые продолжают обработку.

4. Интерпретация результатов

В результате экспериментов была получена следующая зависимость (рис.).

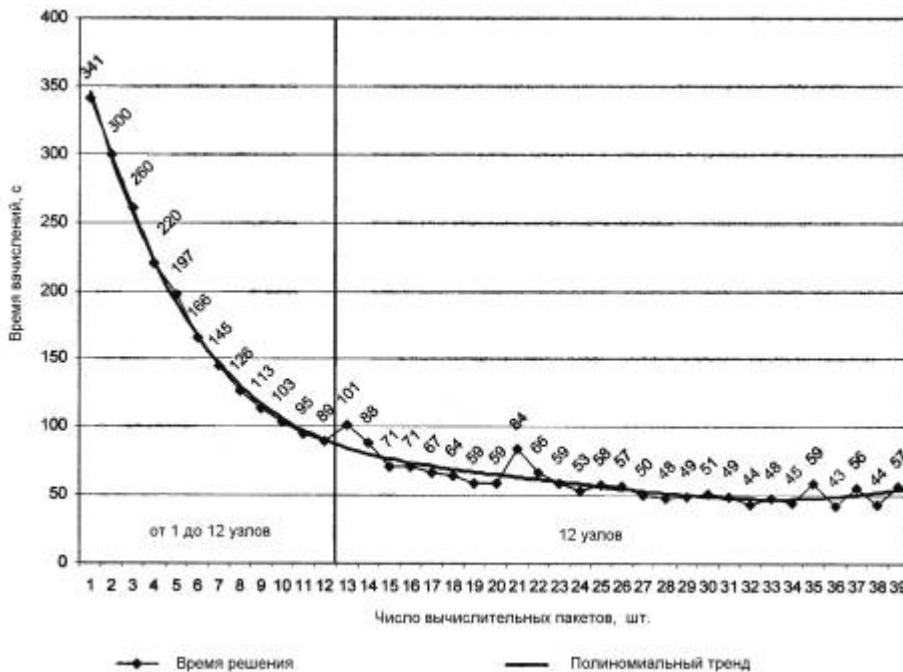


Рис. Зависимость времени решения задачи от количества вычислительных пакетов

График разделен вертикальной линией на две части: левая — число вычислительных пакетов не превышает число узлов, правая — число вычислительных пакетов превышает число узлов. Проанализируем его:

1. Если нет потерь на передачу данных между сервером и узлами, все компьютеры имеют одинаковую мощность, то распределение вычислений между компьютерами уменьшает время расчета пропорционально их числу. Из левой части графика видно, что это не так.
2. Для проверки того, что оказывает большее влияние, различная мощность компьютеров или потери на сеть, необходимо проанализировать правую часть графика. Легко увидеть, что уменьшение вычислительного пакета сглаживает разницу в мощности узлов. Когда число пакетов достигает 34, время выполнения расчета имеет наименьшее значение. Дальнейшее дробление задачи не уменьшает время расчета, а в итоге даже увеличивает его. Сказываются потери на передачу пакетов по сети и на организаторскую работу с ними сервера.

Таким образом, подтверждается гипотеза о большом времени ожидания более мощных или более свободных машин при обработке своего поддиапазона. При увеличении числа подзадач время простоя снижается до минимума.

5. Выводы

Проанализировав данные эксперимента, можно сделать вывод о том, что возможен оптимальный выбор количества пакетов для заданного числа узлов и примерного времени расчета решаемой задачи: величина одного пакета ограничена снизу потерями на обмен данными, ограничение сверху можно оценить, зная количество подключаемых компьютеров, уровень устойчивости к ошибкам и вычислительную требовательность задачи. Кроме того, необходимо исследование зависимости общей вычислительной мощности grid-системы от количества, мощности и распределения составляющих ее компьютеров, а также от величины вычислительных пакетов. Необходимо повысить уровень программного обеспечения и организации обмена данными и результатами.

Перед созданием grid-системы также полезен мониторинг загрузки компьютеров сети с тем, чтобы отобрать класс эффективно решаемых задач.

ЛИТЕРАТУРА

1. Анни П. Этот Grid — неспроста... // Открытые системы. 2003. № 1.
2. Грид-проекты // <http://www.gridclub.ru/projects/>.
3. Коваленко В., Корягин Д. Организация grid: есть ли альтернативы? // Открытые системы. 2004. № 12.
4. Национальные и международные проекты // <http://parallel.ru/meta/projects.html>.
5. Распределенные вычисления в Интернете // <http://parallel.ru/meta/internet.html>.
6. Сверхсложные вычислительные задачи, решаемые на суперкомпьютерах // <http://www.parallel.ru/research/challenges.html>.
7. Телиа Д. OGSA: Где Grid встречается с Web // Открытые системы. 2003. № 1.
8. Фостер Я., Кессельман К., Ник Д., Тьюке С. Grid-службы для интеграции распределенных систем // Открытые системы. 2003. № 1.
9. Черняк Л. Grid как будущее компьютеринга // Там же. 2003.
10. Черняк Л. Web-сервисы, grid-сервисы и другие // Открытые системы. 2004. № 12.
11. Fida: a Toolkit for Developing Distributed Applications for Scientific Computing // <http://fida.chem.washington.edu/docs>.
12. Grid Computing Info Centre: Frequently Asked Questions (FAQ) // <http://www.gridcomputing.com/gridfaq.html>.

R. M. Ganopolsky, D. B. Kepetschuk

*INVESTIGATION RESULTS ON GRID-SYSTEM EFFICIENCY
IN TYUMEN STATE UNIVERSITY*

The article considers problems of creating a GRID-system basing on distributed computer network in Tyumen State University, describing configuration and results of the carried experiment to create a test GRID-system.