

NumGRID, система для решения больших задач численного моделирования

Малышкин Н.В.
МО ВВС ИВМиМГ
Новосибирск
e-mail: nikmal@ssd.ssc.ru
получена 2 октября 2007

Аннотация

Статья посвящена программной системе NumGRID, позволяющей объединять различные высокопроизводительные вычислительные ресурсы в Grid для решения больших задач численного моделирования. Рассказывается о системе NumGRID в целом, приводятся основные идеи и концепции, анализируется класс задач численного моделирования, анализируются результаты тестирования системы NumGRID на реальных задачах и делаются выводы о применимости такого рода Grid систем в рассматриваемой предметной области.

1. Введение

В последние годы все больший интерес вызывает идея распределенной вычислительной обработки в слабосвязанных сетях, получивших название Grid. Такие сети, объединяющие различные компьютеры и вычислительные центры, уже давно используются для решения ряда научных задач, требующих большой вычислительной мощности, например, для поиска простых чисел, расшифровки генома человека, создания новых лекарств, задач астрофизики, прогноза погоды и т.д. [1-5].

Важнейшим классом параллельных задач являются задачи численного моделирования, и именно о них в дальнейшем и пойдет речь. Решение больших и сверхбольших задач численного моделирования на традиционных суперкомпьютерах представляет собой достаточно сложную и нетривиальную задачу. Важно не только правильно и эффективно распараллелить уже существующую последовательную программу, но и учесть различные особенности выполнения параллельной программы на разных суперкомпьютерах. Большинство суперкомпьютеров до сих пор в некоторой мере уникальны и отличаются друг от друга как архитектурой, так и программным обеспечением. При переносе прикладной программы с одного суперкомпьютера на другой зачастую приходится производить переделки не только программного кода, но и алгоритмов, использовать разные системы управления задачами. В дальнейшем с ростом числа требуемых вычислительных ресурсов встает уже задача объединения разнородных суперкомпьютеров в единый вычислитель для решения сверхбольших задач численного моделирования. И именно здесь на помощь приходят Grid технологии.

Несмотря на большое разнообразие различных программных и аппаратных средств для реализации Grid, до настоящего момента не было специализированных средств, позволяющих эффективно решать задачи численного моделирования на Grid. Поэтому в ИВМиМГ СО РАН за последние годы была создана система NumGRID [6-8], позволяющая успешно решать крупные численные задачи на объединении суперЭВМ. В данной работе рассказывается о системе NumGRID в целом, приводятся основные идеи и концепции, анализируется класс задач численного моделирования, анализируются результаты тестирования системы NumGRID на реальных задачах и делаются выводы о применимости такого рода Grid систем в рассматриваемой предметной области.

2. Класс задач численного моделирования

Чтобы понять, какие требования предъявляются к вычислительной Grid сети, вначале необходимо рассмотреть отличительные черты класса задач численного моделирования. Для анализа был взят ряд типичных больших реальных задач численного моделирования из различных областей. Все задачи характеризуются достаточной завершенностью, методы и алгоритмы их решения совершенствовались в течение ряда лет, разработан и отлажен параллельный код, проведены многократные численные эксперименты.

1. Cuevas. Задача моделирования поведения жидкости в условиях “мелководья” (лаборатория L3M, CNRS, Марсель, Франция) [9]

В задаче изучаются отложения осадков в устьях и прибрежных водах, проводится моделирование скорости течения жидкости и колебания поверхности. Программный пакет, реализующий эту задачу, написан на языке Fortran-90 и использует библиотеку DooM для распараллеливания задачи. Библиотека DooM представляет собой средства для распараллеливания (используя библиотеку MPI) по методу декомпозиции расчетной области: планирование распределения, выделение памяти, синхронизация процессов, удовлетворение зависимостей по данным на границах разрезания области.

Соотношение времени вычислений и времени коммуникаций в задаче.

Задача итерационная, на каждом временном шаге проводится решение ряда гидродинамических уравнений на сетке. В самом расчете уравнений параллельных конструкций нет, но сетка, на которой происходят вычисления, представляет собой специальный объект библиотеки DooM. Сетка - это фактически специальная распределенная память, расположенная на разных компьютерах и, таким образом, при записи/чтении данных в нее, пересылка осуществляется автоматически посредством библиотеки DooM. Так как библиотека универсальна и ничего не знает о сути решаемой задачи, то логично ожидать, что страдает производительность. Действительно тестирование показывает, что в процессе расчетов на каждом временном шаге библиотека вызывает множество функций синхронизации (все-со-всеми) и групповых пересылок. Число операций синхронизации и пересылок достигает нескольких сотен на каждый временной шаг. Можно предполагать, что перенос задачи на Grid с медленными коммуникациями не даст ускорения, а возможно, даже и замедлит расчет.

Используемые программные средства: Fortran-90, библиотека Doom, MPI-1.1.

Объем исходного кода программы: 550 Кб, 13700 строк

2. Задача моделирования динамики гравитирующих систем и, в частности, протопланетного диска [10]

В исследованиях нестационарных процессов в гравитационной физике существуют две первоочередные проблемы:

- Найти механизм самоорганизации околозвездного диска, который сгущает вещество, разрушает диск и создает планеты.
- Определить диапазон физических условий: давление, температура и другие параметры среды, которые могли иметь место в околозвездном диске при его самоорганизации.

Для решения этих проблем была разработана численная модель динамики вещества в условиях развития гравитационной неустойчивости, на которой ставятся крупномасштабные вычислительные эксперименты с использованием современных суперЭВМ.

Соотношение времени вычислений и времени коммуникаций в задаче.

При соотношении примерно 100 частиц на 1 ячейку и количестве процессоров меньше 30 время пересылок составляет около 10% в общем времени счета на комплексе МВС1000-128.

Используемые программные средства: Fortran-90, C++, MPI-1.2

Объем исходного кода программы: 154 Кб, 5400 строк.

3. МАЕС. Задача создания типового программно-технического комплекса и технологии испытания цифровых устройств релейной защиты и противоаварийной автоматики для использования на объектах ОАО “ФСК ЕЭС России” [11]

Основное назначение ПТК МАЕС – получение информации о работоспособности устройств релейной защиты/противоаварийной автоматики (РЗ/ПА) при тестировании и функциональной диагностике в условиях, имитирующих их функционирование в реальной сети. В составе комплекса имеется специальный высокопроизводительный программный вычислитель, позволяющий рассчитывать большие и сверхбольшие электрические схемы на мультипроцессорах и мультикомпьютерах. Вычислитель может работать как под ОС Windows (тестовые расчеты на ПК или мультикомпьютере), так и под ОС реального времени QNX (в реальном времени на мультипроцессорах) и UNIX (большие численные эксперименты на мультикомпьютере). При распараллеливании задачи используется разрезание крупной электрической схемы на несколько более мелких и последующий их отдельный расчет, с обменов данными на каждом временном шаге.

Соотношение времени вычислений и времени коммуникаций в задаче.

Рассмотрим возможные варианты расчетов:

- Подсхемы маленькие (единицы, десятки элементов), каждый шаг расчета занимает очень маленькое время (миллисекунды или даже микросекунды). В этом случае получаем вариант: мало счета, много обменов данными. Очевидно, что в этом случае на кластере задача будет распараллеливаться хуже, ввиду того, что необходимо реализовать много времяемких коммуникаций и небольшой счет. На Grid с его медленными коммуникациями время выполнения параллельной программы может даже замедлиться, по сравнению с последовательной версией.
- Подсхемы большие (сотни и тысячи элементов, сложные составные элементы с времяемкими математическими моделями), каждый шаг расчета занимает относительно большое время (секунды). В этом случае получаем вариант: много счета, столько же обменов данными, как и в предыдущем случае. Здесь эффект распараллеливания будет значителен. Так как соотношение счет/коммуникации в пользу счета, на Grid также будет получено значительное ускорение по сравнению с последовательной версией.

Используемые программные средства: Fortran, C++, MPI-1.2, ...

Объем исходного кода программы: 794 Кб, 21000 строк.

4. Другие задачи

Естественно, класс задач численного моделирования не исчерпывается полностью этими тремя задачами. Тем не менее, при анализе разного рода численных параллельных задач, решаемых в суперцентрах, и по публикациям видно, что алгоритмы решения этих задачи довольно типичны. Рассмотрим типовые черты таких задач:

- Большинство из них ныне представляют собой некоторые итерационные алгоритмы по времени, где на каждом временном шаге проводится решение систем уравнений и затем происходит обмен данными между параллельными процессами. При этом, как правило, время, затрачиваемое на расчет каждого шага (решение уравнений и т.д.), достаточно большое и часто превосходит время, требующееся на обмены данными. То есть подзадачи относительно слабо связаны. Это позволяет использовать для решения такие медленные по части коммуникаций суперЭВМ, как кластеры и Grid.
- Задач, которые могут быть разделены на несколько полностью независимых подзадач, не требующих обменов данными друг с другом, очень мало (задачи перебора, комбинаторные и т.д.). Это заставляет использовать различные системы распараллеливания – MPI, OpenMP, HPF и т.д.
- Задачи очень требовательны к мощности процессора и особенно к объему оперативной памяти (используются большие массивы данных). Таким образом, эффективный расчет может быть проведен только на совокупности мощных вычислительных станций, что автоматически приводит к необходимости использовать современные кластеры или мультипроцессоры.
- Подавляющее большинство задач написано на языках C/Фортран и в качестве средства распараллеливания используют библиотеки MPI.
- Задачи требуют большого количества машиночасов для расчетов и могут выполняться по несколько дней. Это приводит к необходимости иметь доступные средства отладки и мониторинга, которые позволяют выявить ошибки в расчетах или программе как можно раньше, экономя тем самым дорогое машинное время и время специалистов.
- Часть задач требует настолько больших вычислительных ресурсов для получения достоверных результатов, что возникает необходимость объединять все имеющиеся вычислительные ресурсы для решения такого рода сверхзадач.

Выводы: Многие задачи компьютерного моделирования для точного решения требуют больших вычислительных ресурсов, и при этом для них зачастую не хватает мощностей даже самых производительных кластеров. С другой стороны, такого рода задачи имеют сложный программный код, часто состоящий из десятков тысяч строк и отлаживаемый годами. Поэтому актуален вопрос о специализированной комплексной Grid системе, которая позволила бы решать задачи численного моделирования на объединении суперЭВМ, при этом желательно сохраняя структуру и текст исходных программ без изменений. При этом количество обменов данными между параллельно исполняющимися фрагментами программы и объемом вычислений во фрагментах существенно влияют на требования к оборудованию, включаемому в состав NumGRID.

3. Требования к GRID системе

На основе выводов, сделанных при анализе задач, сформулируем основные требования к Grid системе:

- Система должна позволять объединять вычислительные кластеры, при этом каждый узел кластера в свою очередь может быть мультипроцессором. Таким образом, суперкомпьютеры объединяются для решения суперзадачи.
- Программировать NumGRID метакомпьютер необходимо как один мультикомпьютер. В коде прикладной программы не должна отражаться распределенная структура метакомпьютера.
- Переделки уже существующих параллельных программ должны быть минимизированы.
- Должна быть возможность исполнения программ, написанных на C, C++ и Фортране 77, 90.
- В качестве средства задания межпроцессных коммуникаций необходимо поддерживать библиотеку MPI.
- “Дружелюбность” прикладного интерфейса. Ввиду использования большого количества разнородного оборудования необходима разработка интегрирующего пользовательского интерфейса для обеспечения максимальной простоты управления задачами. Пользователь должен без особых усилий и дополнительного изучения управлять своими прикладными задачами в NumGRID.
- Требуется обеспечить безопасность и надежность распределенных вычислений в NumGRID.
- Добавление и удаление вычислительных узлов и новых процессов прикладной программы в ходе вычислений не допускается (следует из требования использования библиотек MPI).

4. Система NumGRID

На основе анализа задач и выработанных требований была разработана Grid система для решения больших задач численного моделирования - NumGRID. Система NumGRID представляет собой кроссплатформенное приложение, позволяющее работать как в Windows, так и в UNIX подобных системах. NumGRID состоит из следующих модулей:

- 1) **Библиотека NumGRID_MPI.** Библиотека с поддержкой стандарта передачи сообщений MPI на Grid. Используется прикладной программой, вместо обычных библиотек MPI. NumGRID_MPI позволяет не только использовать уже существующие библиотеки, такие как MPICH, LAM-MPI, но и пересылать MPI сообщения даже при отсутствии выделенных IP адресов у узлов, между локальными подсетями. Для этого в библиотеке реализован механизм маршрутизации через менеджеры задач. Также в библиотеке реализованы различные оптимизации пересылок данных, учитывающие нелинейную топологию сети и разницу во времени пересылок между различными объектами Grid. Например, при операциях групповой рассылки типа MPI_AllGather, пересылка не разбивается на передачу сообщений от каждого узла к каждому, а организуется в виде некоторой древовидной структуры, где по медленным каналам связи (Internet) передается только один экземпляр сообщения, а по быстрым (LAN) сообщение рассылается нужным адресатам.
- 2) **Менеджер задачи (JobManager).** Работает на управляющих узлах кластера и обеспечивает управление прикладной задачей и пересылку данных между кластерами. Для каждой задачи создается свой JobManager.
- 3) **NumGRID сервер.** Работает на управляющих узлах всех кластеров, входящих в NumGRID, и обеспечивает все функции пользовательского интерфейса: авторизацию и аутентификацию пользователей (шифрование SSL, авторизация через PAM и библиотеку shadow), загрузку и выгрузку прикладных программ, их данных, автоматическую компиляцию программ и т.д.
- 4) **NumGRID клиент.** Используется пользователями для запуска, управления своими задачами в NumGRID, пересылки данных в Grid, получения результатов, удаленной компиляции и т.д.
- 5) **Модуль управления кластером (NumGRID TaskControl).** Предназначен для унификации интерфейсов доступа к кластерам, поддерживает как множество стандартных очередей задач, таких как PBS, Torque, SGE, очередь MBC1000, так и позволяет гибко настраиваться на новые системы очередей.

- 6) **Система мониторинга.** Обеспечивает контроль состояния процессов, все коммуникации MPI и позволяет получать и записывать переменные в программе во время выполнения. В качестве системы мониторинга используется пакет GEPARD [12].

Для описания поведения прикладных параллельных программ на Grid применяется специально разработанная спецификация задачи NumGRID. В спецификацию входит множество самых различных параметров, позволяющих гибко настраивать ПО NumGRID для выполнения конкретных задач. К такого рода параметрам относятся списки исходных файлов задачи и способы их компилирования на конкретных ОС, состав требуемого для расчета оборудования (количество требуемых процессоров, дисковой памяти, ...), настройка конфигурации Grid и т.п.

Таким образом, в качестве ключевых отличий системы NumGRID от остальных распространенных Grid систем общего назначения можно отметить следующее:

- Динамическое построение NumGRID из имеющихся ресурсов. Для каждой конкретной задачи есть возможность собрать свой NumGRID, состоящий из необходимых вычислительных ресурсов.
- Унифицированный запуск и управление ходом решения прикладных задач на NumGRID.
- Поддержка библиотеки MPI на NumGRID и возможность исполнять большие численные модели на объединении мультикомпьютеров.
- Простота использования и установки, пользователь работает в привычном ему окружении. При установке системы на кластеры или не требуется вносить вообще никаких изменений в программное обеспечение кластеров, или эти изменения минимальны (требуется только подключить дополнительную библиотеку). Также не требуется вносить никаких изменений в административную политику управления кластерами.
- Вычислительные кластеры включаются в NumGRID не монопольно, и могут быть легко использованы и в обычных расчетах. Например, возможна такая ситуация, что на части узлов считаются задачи под NumGRID, а на остальных узлах считаются обычные параллельные задачи, использующие только этот кластер.
- Реализована возможность использовать MPI в прикладных программах на любых объединениях вычислительных ресурсов.
- В силу узкой специализации системы обеспечивается высокая производительность в прикладных расчетах.
- Исходные прикладные программы могут быть запущены на NumGRID практически без изменений (не требуется вносить в исходный код каких-либо дополнительных инструкций, менять структуру программы и т.д.).
- Можно выполнять специализированную часть вычислений задачи на специализированной установке, например там, где установлен дорогостоящий пакет прикладных программ.

5. Тестирование

Разработанная Grid система NumGRID была подвергнута комплексному тестированию, с целью выявления слабых мест и анализа ее применимости на классе задач численного моделирования. В качестве тестов выступали как специально написанные тестовые MPI приложения, так и реальные прикладные задачи. Тесты, касающиеся поддерживаемого оборудования, операционных систем, компиляторов, синтетических MPI приложений, можно найти в [7]. Далее приведены результаты тестирования реальных прикладных задач.

В рамках исследования было проведено тестирование трех приложений на Grid, созданном из различных кластеров Сибирского суперкомпьютерного центра (ССКЦ, www2.sccc.ru). Бралась реальные входные данные для задач, и затем проводился расчет на различных комбинациях кластеров. Вкратце опишем все проводимые тесты.

1. *Cuevas. Задача моделирования поведения жидкости в условиях мелководья (лаборатория L3M, CNRS, Марсель, Франция)*

- а) Тестирование на 4-процессорном компьютере smp4x64.sccc.ru (Ускорение хорошее).

Таблица 1.

Число параллельных процессов	1	2	3	4
Время расчета (с)	63	35	29	25

б) Тестирование на 4-процессорном компьютере smp4x64.sccc.ru. (Налицо значительное увеличение времени расчета с ростом числа процессоров).

Таблица 2.

Число параллельных процессов	1	2	3
Время расчета (с)	62	125	145

в) Тестирование на Grid из smp4x64.sccc.ru и itanium2.sccc.ru. Время расчетов увеличилось еще в несколько раз по сравнению с пунктом б).

Выводы: В задаче применено средство автоматического распараллеливания DooM, с множеством групповых синхронизаций на каждом шаге. Из-за большого (сотни) числа групповых операций, использованных в прикладной задаче, и маленького времени счета задача хорошо выполняется на системах с общей памятью. Но уже на одном кластере распараллеливание не даёт никакого положительного эффекта. Grid тем более не применим в этом случае.

2. Задача моделирования динамики гравитирующих систем и, в частности, протопланетного диска

а) Тестирование на кластере mvs1000-128.sccc.ru (Ускорение хорошее).

Таблица 3.

Число параллельных процессов	1	2	4	6
Время расчета (с)	227	160	119	101

б) Тестирование на кластере itanium2.sccc.ru (Ускорение хорошее).

Таблица 4.

Число параллельных процессов	1	2	3
Время расчета (с)	22	13	11.5

в) Тестирование на Grid из mvs1000-128 (1) и itanium2 (2) (Ускорение хорошее).

Таблица 5.

Число параллельных процессов	1(1) 1(2)	2(1) 2(2)	3(1) 3(3)
Время расчета (с)	79	60	51

г) Тестирование на Grid из smp4x64 (1) и itanium2 (2) (Ускорение хорошее).

Таблица 6.

Число параллельных процессов	1(1) 1(2)	2(1) 2(2)
Время расчета (с)	14	12

Примечание: Здесь и далее в таблицах обозначение вида x(y) следует читать так: x процессов на кластере y. Например, колонка три вышеприведенной таблицы указывает время, затраченное на расчет в том случае, если на кластере 1 (mvs100-128) было запущено 2 процесса программы и на кластере 2 (itanium2) тоже было запущено 2 процесса программы (Всего 4 процесса участвовали в расчете).

Выводы: Задача хорошо распараллеливается на кластерах. При этом на Grid также во всех случаях показывается хороший рост производительности, несмотря на наличие в программе групповых операций пересылок (все-со-всеми). Связано это с тем, что время расчетов значительно превышает время пересылок, здесь же играют роль алгоритмы оптимизации выполнения групповых пересылок NumGRID_MPI.

3. МАЕС. Задача создания типового программно-технического комплекса и технологии испытания цифровых устройств релейной защиты и противоаварийной автоматики для использования на объектах ОАО ФСК ЕЭС

Тестирование проводилось на объединении кластеров mvsl000-128.sccc.ru и itanium2.sccc.ru

3.1) Для теста была взята небольшая электрическая схема, состоящая из двух десятков элементов. Для тестов она была разрезана на 2 и 4 подсхемы.

а) Тестирование на кластере mvsl000-128.sccc.ru (Ускорение хорошее)

Таблица 7.

Число параллельных процессов	1	2	4
Время расчета (с)	115	70	50

б) Тестирование на кластере itanium2.sccc.ru (Ускорение хорошее)

Таблица 8.

Число параллельных процессов	1	2	4
Время расчета (с)	25	15	9

в) Тестирование Grid из mvsl000-128 (1) и itanium2 (2)

(Явное замедление. Каждый шаг расчета проходит достаточно быстро (микросекунды), затем начинаются коммуникации, которые съедают весь потенциальный выигрыш да еще и существенно замедляют расчет)

Таблица 9.

Число параллельных процессов	1(1) 1(2)	2(1) 2(2)
Время расчета (с)	112	168

3.2) Учитывая предыдущий тест, для нового теста была взята значительно более крупная электрическая схема, состоящая из нескольких сотен элементов, включая математически сложные и нелинейные элементы. Количество коммуникаций осталось такое же, как и в предыдущем случае, но время счета каждой итерации заметно возросло (в сотни раз). Было так же в несколько раз уменьшено общее количество итераций, для того чтобы схема не считалась совсем уж долго (часы). Важно было не конечное время счета, а эффективность распараллеливания. Для тестов схема была разрезана на 2 и 4 подсхемы.

а) Тестирование на кластере mvsl000-128.sccc.ru (Ускорение хорошее)

Таблица 10.

Число параллельных процессов	1	2	4
Время расчета (с)	512	307	181

б) Тестирование на кластере itanium2.sccc.ru (Ускорение хорошее)

Таблица 11.

Число параллельных процессов	1	2	4
Время расчета (с)	117	60	35

в) Тестирование на Grid из mvsl000-128 (1) и itanium2 (2)

Таблица 12.

Число параллельных процессов	1(1) 1(2)	2(1) 2(2)
Время расчета (с)	134	82

Выводы: Задача хорошо распараллеливается на кластерах. На Grid эффект от распараллеливания есть только в том случае, если время счета на каждой итерации достаточно велико, а коммуникаций немного.

Общий вывод

Тестирование на различных задачах и с различными конфигурациями оборудования показало, что разработанное программное обеспечение позволяет достаточно эффективно решать ряд задач численного моделирования на NumGRID. Основным критерием того, подходит или нет задача для решения на сети объединенных суперЭВМ, можно считать соотношение времени счета на каждом временном шаге и времени, затрачиваемого на пересылки данных между процессами задачи. Причем даже если пересылки данных достаточно большого объема и даже в случае массовых рассылок от всех процессов ко всем, на Grid все еще может быть достигнуто ускорение.

Тем не менее, есть большой класс задач с частыми пересылками данных и относительно небольшим временем счета на каждом шаге, где применение Grid систем нецелесообразно, и здесь, по-видимому, стоит ориентироваться только на развитие и наращивание мощности современных кластеров или систем с общей памятью.

6. Заключение

В проекте NumGRID был проведен анализ существующих средств построения Grid систем, рассмотрены различные прикладные задачи численного моделирования. На основе проведенного анализа сформулированы требования к Grid системе для решения задач численного моделирования. Разработана система NumGRID, позволяющая объединять различные разнородные вычислительные ресурсы для решения больших и сверхбольших задач численного моделирования.

Результаты работы были протестированы на различных суперЭВМ, таких как кластеры ИВМиМГ СО РАН, кластере НГУ, в Сибирском суперкомпьютерном центре, на кластерах лаборатории ЛЗМ в Лионе и Марселе, в СибНИИЭ.

На созданной сети NumGRID были решены такие большие задачи, как моделирование протопланетного диска, моделирование поведения жидкости в условиях "мелководья", создание типового программно-технического комплекса испытания цифровых устройств релейной защиты и противоаварийной автоматики.

Разработанная система NumGRID позволяет широко использовать вычислительные ресурсы для численного моделирования различных физических экспериментов.

Список литературы

1. SETI project. <http://setiathome.berkeley.edu/>
2. EuroGrid. <http://www.eurogrid.org>
3. Проект distributed.net. <http://distributed.net/>
4. Проект LCG. <http://lcg.web.cern.ch/LCG/>
5. EGEE project. <http://www.eu-egee.org/>
6. Fougere D. NumGrid Middleware: MPI Support for Computational Grids /Fougere D., Gorodnichev M., Malyshkin N., Malyshkin V., Merkulov A., Roux B. // In Proceedings of the PaCT-2005. Springer Verlag. LNCS 3606. 2005. – P. 313–320.
7. Малышкин Н. Тестирование NumGRID MPI на объединениях мультимикомпьютеров // Труды конференции молодых ученых ИВМиМГ СО РАН. – 2006. – С. 118–124.
8. Городничев М. NumGRID: Объединение кластеров для научных вычислений /Городничев М., Малышкин В., Малышкин Н., Меркулов А., Фужер Д., Ру Б. // Труды второй международной конференции "Распределённые вычисления и Грид-технологии в науке и образовании". – 2006. – С. 59–60.
9. Ramírez León H. Current Trends in High Performance Computing and Its Applications /Ramírez León H., Rodríguez Cuevas C., Herrera Díaz E. //Proceedings of the International Conference on High Performance Computing and Applications. – 2004. – Shanghai, P.R. China.

10. Snytnikov V. Three-Dimensional Numerical Simulation of a Nonstationary Gravitating N-Body System with Gas /Snytnikov V., Vshivkov V., Kuksheva E., Neupokoev E., Nikitin S., Snytnikov A. // Astronomy Letters. 2004. – V. 30, N. 2. – P. 124–138.
11. Наумкин И. Программа анализа переходных электромагнитных процессов в сложных электроэнергетических схемах для многопроцессорных вычислительных систем /Наумкин И., Малышкин Н., Остапкевич М., Корнеев В. // Труды международной научно-технической конференции "Передача энергии переменным током на дальние и сверхдальние расстояния". Новосибирск. – 2003. – Т. 2. – С. 16–22.
12. Малышкин В. Отладчик параллельных программ для мультимпьютера /Малышкин В., Романенко А. // Автометрия, Новосибирск. – 2003. – Т. 39, N 3.– С. 109–114.

NumGRID, the system for large-scale numerical applications solving

Malyshkin N.V.

The paper describes NumGRID metacomputing system which allows to integrate several high performance computational resources into a united Grid. Using this Grid it is possible to solve a lot of large-scale numerical problems. Here one can find key ideas and objectives of NumGRID middleware, analysis of large-scale numerical tasks class, descriptions and results of different tests that were done using NumGRID. There are also some conclusions about Grid applicability to a range of scientific problems solution.