

ОСОБЕННОСТИ ПОСТРОЕНИЯ МАСШТАБИРУЕМЫХ КОМПОЗИТНЫХ ВЫЧИСЛИТЕЛЬНЫХ ПРИЛОЖЕНИЙ ДЛЯ МОДЕЛИРОВАНИЯ СЛОЖНЫХ СИСТЕМ

С.В. Ковальчук, С.В. Иванов, А.В. Бухановский

Класс композитных приложений, использующих для решения поставленной задачи набор готовых программных модулей, в настоящее время достаточно популярен при построении бизнес приложений [1,2]. В то же время данный класс приложений использует подходы, применение которых будет не менее эффективным при разработке программных комплексов, решающих ресурсоемкие научные задачи. К таким задачам, в том числе, относятся задачи моделирования сложных систем. Под сложной системой [3] понимается система, которая (а) состоит из очень большого количества элементов; (б) допускает «дальние» связи между элементами; (в) обладает многомасштабной (пространственно-временной) изменчивостью. Как следствие, поведение системы описывается целым комплексом взаимодействующих моделей (в общем случае основанных на различном математическом аппарате и соответствующих различным областям знаний) и требует существенных вычислительных затрат. Для решения задач моделирования таких систем зачастую приходится организовывать взаимодействие уже существующих вычислительных модулей в составе единого программного комплекса. Использование готовых вычислительных модулей может быть обосновано рядом причин:

1. повторное использование кода позволяет избежать дополнительных затрат на написание соответствующих программных модулей;
2. программные модули, отвечающие за моделирование отдельных элементов сложной системы, могут быть слишком трудоемки для реализации в рамках текущего проекта;
3. использование существующих программных решений может быть продиктовано требованиями технического задания, предписывающего разработчикам использовать конкретные программные решения (например, в силу их сертифицированности).

Тем не менее, ввиду высокой вычислительной сложности решаемых задач, разрабатываемый программный комплекс, использующий сторонние вычислительные модули, должен отвечать требованиям эффективности по использованию параллельных вычислительных ресурсов платформы, на которой производится исполнение. Достижение поставленной цели возможно при условии обеспечения масштабируемости разрабатываемого программного комплекса как цельного программного решения. Для этого необходимо (а) обеспечить наиболее эффективное использование возможностей распараллеливания, предоставляемых каждым из используемых программных модулей, (б) добиться наиболее эффективного функционирования совокупности программных модулей как единого программного комплекса, решающего конкретную задачу предметной области. Данная работа посвящена исследованию особенностей проектирования наукоемкого программного обеспечения для моделирования сложных систем с учетом его модульной структуры и возможности параллельного исполнения отдельных модулей.

Анализ особенностей программных модулей

Использование сторонних вычислительных модулей приводит к необходимости учитывать особенности каждого из них в процессе интеграции в состав программного комплекса. К таким особенностям следует отнести:

- Ограничения предметной области, очерчивающие круг задач, для решения которых подходит данное программное средство.
- Особенности используемых технических решений: язык программирования; аппаратная платформа и операционная система, на которой изначально предполагалось функционирование модуля; множество используемых системных и прикладных библиотек; используемые технологии распараллеливания; способ вызова и использования.
- Алгоритмические особенности, характерные, как в общем для используемого алгоритма, так и для конкретной его реализации.
- Возможность модификации исходных текстов программного средства может быть ограничена рядом причин: наличием лицензионных ограничений, сертифицированностью данной реализации, сложностью для понимания структуры исходного текста, отсутствием в команде разработчиков специалистов по данному языку или технологии и т.п.
- Юридические, политические и другие внешние ограничения, накладываемые на данное программное средство.

Для многих программных модулей указанные особенности проистекают из того, что первоначально их разработка преследовала цели решения конкретной задачи, поставленной перед специалистами-разработчиками предметной области. Как следствие, зачастую специалистам, занимающимся организацией повторного

использования различных программных средств, приходится сталкиваться с большим разнообразием языков программирования, стилей написания исходных текстов (если предполагается возможность их модификации), платформ исполнения и т.п.

Дополнительную сложность может представлять необходимость использования в составе программного комплекса вычислительных модулей, изначально разработанных только для внутреннего использования специалистами предметной области. В этом случае может потребоваться реализации внешних по отношению к программному модулю механизмов обработки ошибок, контроля стабильности работы программного средства и т.п.

Параллельная реализация вычислительных модулей

Каждый из используемых вычислительных модулей, как указывалось выше, может использовать свою технологию распараллеливания и свой вариант ее применения к конкретному алгоритму. При использовании же вычислительного модуля в составе программного комплекса возникает задача обеспечения максимальной эффективности распараллеливания программного модуля с учетом особенностей программно-аппаратной платформы, используемой для ее запуска. Одним из способов решения данной задачи является использование моделей, позволяющих оценить параллельную производительность вычислительных модулей до их непосредственного запуска. Простейшим примером аналитической модели, позволяющей оценить эффективность параллельной реализации алгоритма, является закон Амдала. Но для более эффективного анализа параллельной производительности алгоритма необходимо использовать систему моделей, учитывающих:

- актуальные особенности программно-аппаратной платформы, на которой производится запуск приложения;
- параллельную структуру исследуемого алгоритма;
- характеристики входных данных для запуска программы.

Учитывая указанные параметры, можно оценить время выполнения приложения в заданных условиях в зависимости от параметров параллельного запуска. Более того, по результатам моделирования можно определить оптимальные параметры распараллеливания, позволяющие достичь наилучшего времени работы и загрузки вычислительных ресурсов.

Интеграция вычислительных модулей

Для успешной работы вычислительного модуля в составе программного комплекса необходимо обеспечить унифицированный интерфейс доступа к нему со стороны управляющей подсистемы. Использование концепции сервисно-ориентированной архитектуры [4] позволяет добиться организации унифицированного доступа к вычислительным модулям, что, в свою очередь, существенно упрощает процесс их интеграции. Таким образом, доступ к каждому вычислительному модулю осуществляется посредством сервиса, осуществляющего запуск и контроль работы вычислительного модуля, организацию потоков входных и выходных данных, вторичную обработку ошибок, произошедших в процессе работы вычислительного модуля.

Другой важной задачей, возникающей в процессе интеграции вычислительных модулей, является необходимость организации преобразования входных и выходных данных (а) к виду, требуемому используемым программным модулем, (б) к виду, удобному для представления результатов работы программного комплекса в целом. Одним из вариантов решения данной задачи является реализация хранилища данных, используемого вычислительными модулями программного комплекса в процессе работы. Использование концепции REST [5] позволяет организовать доступ к данным с прозрачным (с точки зрения стороны, формирующей запрос) конвертированием в нужный формат.

Обеспечение производительности целого комплекса

Приведенные выше подходы позволяют обеспечить эффективную работу с отдельными вычислительными модулями. Тем не менее, для обеспечения эффективного функционирования комплекса в целом необходим совместный анализ характеристик вычислительных модулей, целью чего является определение оптимальных параметров запуска. При этом важно учитывать ряд факторов, влияние которых возрастает с переходом к задаче комплексного анализа системы вычислительных модулей:

- Возможность взаимозаменяемости модулей аналогичной функциональности. Данная возможность может быть актуальной, когда один из пакетов позволяет проводить расчеты, которые невозможно выполнить альтернативным пакетом, а так же в случае обеспечения одним из пакетов лучшего времени расчетов или точности для данных условий запуска.
- Необходимость обработки данных, передаваемых управляющей системой от одного пакета другому, с учетом их преобразования. Этот фактор важен, поскольку в условиях оценки производительности программного комплекса при решении сложной задачи, необходимо оценивать полные цепочки исполнения, включающие как этапы непосредственной работы вычислительных модулей, так и промежуточные этапы, включающие и обработку данных.

Использование интеллектуальных технологий

При управлении работой вычислительных модулей производится динамическое построение схемы их параллельного запуска. Принятие решения о выборе способа распараллеливания в конечном итоге осуществляется на основании знаний экспертов, которые условно разделяются на две категории:

- Эксперты предметной области, располагающие информацией о способах использования вычислительных модулей, требованиях к входным и выходным данным, принципиальных особенностях реализованного в них алгоритма и т.п.
- Эксперты в области параллельных вычислений, обладающие комплексом знаний для моделирования параллельной производительности, как отдельных вычислительных модулей, так и комплекса в целом. Кроме того, на основании знаний экспертов данной области производится анализ результатов моделирования с последующим принятием решения о параллельном запуске вычислительных модулей.

Примеры реализации предложенного подхода

Предложенный подход был использован при реализации прототипа программного комплекса в проекте «Высокопроизводительный программный комплекс моделирования и прогноза экстремальных гидрометеорологических явлений и расчета воздействий на морские объекты и сооружения» [6], что обеспечило эффективное решение комплексом поставленной перед ним задачи с достижением достаточно высоких показателей масштабируемости (было получено общее ускорение около 84 раз на 128 вычислителях). На рисунках 1А и 1Б представлены общая сервисная архитектура и принцип работы интеллектуальной подсистемы для данного программного комплекса. Рисунок 1В иллюстрирует эффективность параллельной реализации модуля расчета морского волнения, работающего в составе программного комплекса.

В настоящее время подход применяется в процессе работы над проектом «Разработка высокопроизводительного программного комплекса для квантово-механических расчетов и моделирования наноразмерных атомно-молекулярных систем и комплексов» [7]. На рисунке 1Г приведен сравнительный анализ теоретических моделей времени параллельной работы пакетов квантово-механических расчетов, используемых в рамках комплекса. Результаты этого анализа используются при принятии решения о конфигурации параллельного запуска пакета, обеспечивающего наилучшую производительность.

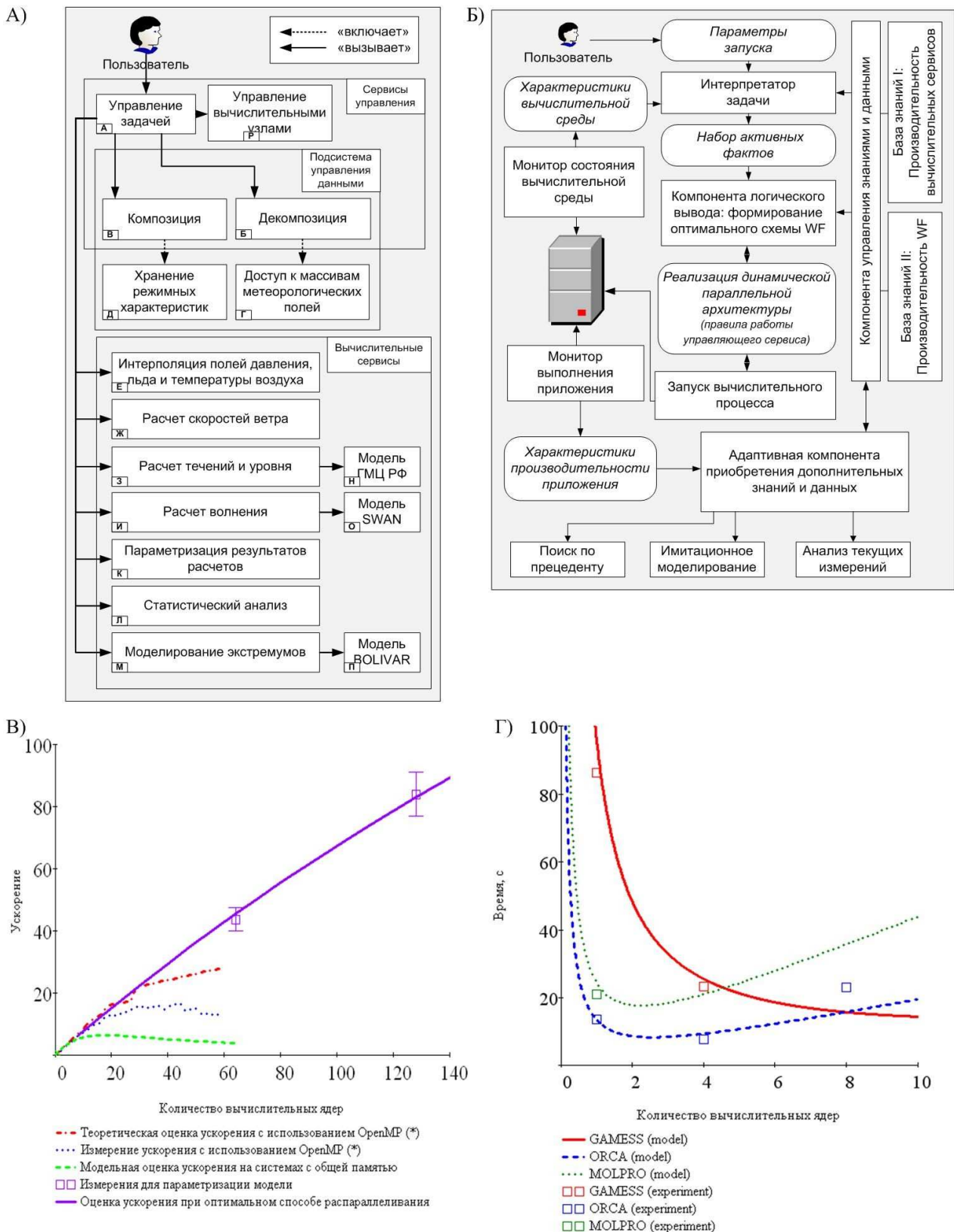


Рис. 1. Архитектура высокопроизводительных программных комплексов и анализ их производительности.

Рисунок 1: Архитектура высокопроизводительных программных комплексов и анализ их производительности. А) Сервисная структура Высокопроизводительного программного комплекса моделирования и прогноза экстремальных гидрометеорологических явлений и расчета воздействий на морские объекты и сооружения, Б) Общая схема интеллектуального управления работой параллельных программных модулей, В) Графики параллельного ускорения теоретической модели и измерений модуля расчета параметров

морского волнения SWAN, Г) Теоретические модели времени параллельной работы квантово-механических пакетов.

Выводы

Рассматриваемый в работе подход позволяет добиться высокой масштабируемости высокопроизводительных программных комплексов, что достигается путем динамической конфигурации параллельного исполнения отдельных вычислительных модулей. Для построения оптимальной схемы параллельного исполнения используется интеллектуальный механизм на основе экспертных знаний. Важной составляющей решения при этом является организация унифицированного доступа к отдельным модулям комплекса, что позволяет упростить организацию их параллельного автоматического запуска. Использование сервисного подхода, позволяет упростить разработку композитных программных комплексов, облегчая управление существующими и интеграцию новых модулей. Интегрируемые модули могут использоваться в качестве альтернативы уже существующим прикладным пакетам, а так же обеспечивать решение новых задач.

ЛИТЕРАТУРА:

1. Сипл Р. Десять правил создания композитных приложений // PC Week/RE (556) 46, 2006.
2. Граничин О.Н., Кияев В.И. Информационные технологии в управлении. Бином. Лаборатория знаний Интернет-Университет Информационных Технологий, 2008. — 336 с.
3. Vossara N. Modeling Complex Systems. — Springer New York, 2004. — 397 p.
4. Lublinsky V. Defining SOA as an architectural style. 9 January 2007. <http://www.ibm.com/developerworks/architecture/library/ar-soastyle/>
5. Fielding R.T. Architectural Styles and the Design of Network-based Software Architectures. Doctoral thesis, University of California, Irvine, 2000. — 162 p.
6. Высокопроизводительный программный комплекс моделирования экстремальных гидрометеорологических явлений. Часть I: Постановка задачи, модели, методы и параллельные алгоритмы / А.В. Бухановский, С.В. Ковальчук [и др.] // Научно-технический вестник СПбГУ ИТМО. Выпуск 54. Технологии высокопроизводительных вычислений и компьютерного моделирования. – СПб.: СПбГУ ИТМО, 2008. – С. 56-63.
7. Высокопроизводительный программный комплекс моделирования наноразмерных атомно-молекулярных систем / В.Н.Васильев [и др.] // Научно-технический вестник СПбГУ ИТМО. Выпуск 54. Технологии высокопроизводительных вычислений и компьютерного моделирования. – СПб.: СПбГУ ИТМО, 2008. – С. 3-12.