

Интегративные возможности курса параллельного программирования

И.Г. Захарова, А.А. Захаров

ФГБОУ ВПО «Тюменский государственный университет»

В статье рассмотрены особенности содержания курса параллельного программирования, ориентированного на студентов направлений «Математическое обеспечение и администрирование информационных систем» и «Информационная безопасность». Обобщен опыт преподавания алгоритмов и технологий параллельного программирования в рамках общих курсов и в качестве самостоятельной дисциплины. Показаны интегративные возможности отдельного курса для обобщения, осмысления и применения знаний программирования, компьютерной графики и прикладной математики в практикоориентированных проектах.

1. Введение

Современные требования к подготовке программистов, сформулированные в NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing [1] предполагают, что образовательные программы уровня бакалавриата в области информационных технологий, точнее, Computer Science (CS) и Software Engineering(SE), обязательно должны включать курсы по параллельным и распределенным вычислениям (PDC, Parallel and Distributed Computing). При этом, для конкретных образовательных программ открытыми остаются вопросы о том, должна быть в учебном плане одна дисциплина или все-таки несколько, обязательна ли она или факультативна, читать ли ее на младших или на старших курсах. С другой стороны, содержание фундаментальных дисциплин, касающихся языков, методов и технологий программирования, структур и алгоритмов компьютерной обработки данных, компьютерной геометрии и графики, дискретной и вычислительной математики, будет неполным без освещения в них роли и возможностей алгоритмов и технологий параллельного программирования в программной реализации решений различных прикладных задач. И на этот счет можно найти достаточно полные и четкие предложения по содержанию соответствующих модулей [2]. Но здесь надо понимать, что трудно требовать от всех преподавателей владения технологиями параллельного программирования, и поэтому вопросы PDC в таких «включениях» будут, скорее всего, представлены на теоретическом уровне и проиллюстрированы (в лучшем случае) готовым программным кодом. Представляется, что курс параллельного программирования, читающийся на старших курсах и ориентированный на выполнение практических проектов, может обеспечить как освоение алгоритмов и технологий PDC, так и углубление компетенций, полученных в ходе изучения базовых математических и компьютерных дисциплин.

2. Изучение PDC для программирования

В Тюменском университете накоплен определенный опыт обучения технологиям параллельных и распределенных вычислений. Изначально, начиная с 2001 г., подготовка проводилась на специальности (с 2011 г. направлении бакалавриата) «Математическое обеспечение и администрирование информационных систем» в рамках обязательной дисциплины «Параллельное программирование» (7 семестр). Курс был ориентирован на практическое освоение технологии MPI на базе C/C++ в процессе программной реализации базовых параллельных алгоритмов умножения матриц и решения систем линейных алгебраических уравнений. Затем курс был расширен и включил также изучение технологии OpenMP (C/C++). В дальнейшем, отдельные разделы, связанные с PDC, были включены в обязательные курсы «Объектно-ориентированное программирование» (3-4 семестры, элементы многопоточного программирования на примере класса Thread библиотеки Microsoft TPL), «Структуры и алгоритмы компьютерной обработки данных» (3 семестр, параллельные алгоритмы для задач поиска и сортиров-

ки), «Операционные системы» (4 семестр, вопросы синхронизации, мьютексы и семафоры), а также дисциплину специализации «Сетевые технологии» (5-6 семестры, сетевые топологии, модели и протоколы передачи сообщений). В определенной мере все это позволило если не организовать непрерывную подготовку по PDC, то, по крайней мере, органично включить ознакомительные модули в базовые дисциплины. Этот подход имеет свои преимущества, поскольку не требует от преподавателя принципиальной перестройки уже отработанного курса и позволяет студентам познакомиться с различными аспектами PDC и освоить базовые понятия.

Аналогичная структура подготовки по параллельному программированию сложилась на направлении бакалавриата «Информационная безопасность» и специальности «Компьютерная безопасность», относящихся к одной укрупненной группе. Однако для этих образовательных программ изначально в учебном плане не предусматривалось самостоятельного курса параллельного программирования. Изучение элементов параллельного программирования служило цели сравнительного анализа различных моделей программирования. Для иллюстрации остановимся на кратком обзоре содержания включенного раздела «Технологии параллельного программирования» дисциплины «Языки программирования», которая читается во 2-4 семестрах. В качестве основы для изложения идей параллельного программирования используются модели реализации подпрограмм, а также управления памятью. Именно для их программной реализации в данном разделе (4 семестр) рассматриваются технологии OpenMP и MPI с акцентом на особенности реализации соответствующих управляющих конструкций и функций. Включение такого раздела не только не разрушает общей концепции курса – изучение семантики языков программирования, но и позволяет органично рассмотреть и сопоставить сущность нескольких моделей и технологий параллельного программирования.

В то же время, необходимость освоения алгоритмов и технологий параллельного программирования определяется тем, что к содержанию компетенций для образовательных программ направления «Информационная безопасность» стандартами (ФГОС ВПО, ФГОС ВО) предъявляются требования, соответствующие области CS. Принципиально важно и то, что выпускники, в основном, ориентируются на работу программистов. Поэтому особых отличий в сквозной подготовке в области PDC от направления «Математическое обеспечение и администрирование информационных систем» в настоящее время уже нет. Исключение составляет углубленное содержание дополнительных разделов по архитектуре распределенных вычислительных систем и сетевым технологиям, определяемое спецификой образовательных программ направления «Информационная безопасность».

Необходимо отметить, что для всех упомянутых образовательных программ наибольший интерес студентов вызывало выполнение заданий практического типа, не предполагавших самостоятельной разработки параллельных алгоритмов и программ. В первую очередь, это вычислительные эксперименты по тестированию параллельных алгоритмов на самых доступных вычислительных системах (от нетбука до сконфигурированной под MPI системы 4-ядерных компьютеров в обычной учебной лаборатории) и сравнению эффективности многопоточных приложений для различных языков программирования (Java, C/C++, C#, Python, Erlang) и технологий (OpenMP, MPI, CUDA, OpenACC). Именно этот факт привел к выводу о возможности разработки обобщающего курса PDC, целью которого будет не только развитие квалификации программистов-профессионалов, но и углубление знаний фундаментального характера, которые часто и даются, и воспринимаются на теоретическом уровне.

3. Изучение PDC для интеграции знаний

Одно лишь соблюдение требований отечественных и международных стандартов и рекомендаций при составлении учебного плана образовательной программы не может дать гарантированного результата в плане качества подготовки выпускников. Учебные планы, как правило, перегружены, поскольку требование следовать примерным образовательным программам (в частности, для упомянутых специальностей и направлений) оставляет весьма ограниченные возможности для включения курсов PDC. Кроме того, традиционно, в образовательных программах подготовки программистов соответствующие дисциплины трудно обеспечить квалифицированными преподавателями, и, как следствие, они воспринимаются студентами как сугубо теоретические. Поэтому их изучение не оказывает никакого влияния на качество подготовки

в целом. При схеме изучения PDC по принципу «модули-разделы плюс итоговый курс параллельного программирования» на уровне бакалавриата и специалитета реально можно говорить только о возможности освоения базовых понятий, основных алгоритмов и технологий параллельного программирования с их практическим применением для программной реализации решения простейших задач дискретной и вычислительной математики.

Очевидно, что один и тот же курс можно читать совершенно по-разному – все зависит от поставленной цели. В этом смысле параллельное программирование не является исключением. Более 10 лет для специальности «Математическое обеспечение и администрирование информационных систем» он читался именно с упором на программирование. Но с переходом на бакалавриат возникла проблема – как определить дисциплину, в ходе изучения которой будущие программисты смогут повторить, отрефлексировать и осмыслить материал множества дисциплин, которые они изучали за предшествующий период. Наиболее подходящее время – это 6 семестр, перед выбором темы дипломной работы, которая должна отражать реальную квалификацию выпускника и перспективы его профессионального развития, в том числе в сфере PDC. Это имеет существенное значение, поскольку в Тюменской области очень востребованы специалисты в области проектирования и разработки высокопроизводительных вычислительных систем с использованием суперкомпьютерных технологий и параллельного программирования для решения задач разведки, добычи и транспортировки углеводородного сырья на базе многопараметрических математических моделей нефтегазоносного пласта. Это, в частности, разработка программных комплексов для изучения геологических объектов, проведения вычислительных экспериментов по исследованию процессов фильтрации в скважинах и окружающих породах, визуализации гидродинамических и геологических моделей, обработки и интерпретации данных сейсморазведки и др.

Именно поэтому представляется оптимальным чтение в 6 семестре (для бакалавриата) и 8 семестре (для специалитета) интегрированного курса, включающего, пусть и в упрощенной постановке, решение практических задач. Лекции (18 – 36 часов, в зависимости от образовательной программы) включают обзор основных сведений об архитектуре многопроцессорных систем, моделях и технологиях параллельного программирования OpenMP, MPI, CUDA, а также многопоточном программировании на базе библиотеки Microsoft TPL. В рамках лабораторных занятий (36 часов) выполняются проекты двух типов: 1) вычислительный эксперимент для определения времени выполнения, ускорения и эффективности для параллельных алгоритмов или для сравнения параллельно выполняющихся различных алгоритмов решения одной и той же задачи (численное интегрирование, поиск и сортировка, алгоритмы на графах, численное решение алгебраических уравнений и систем); 2) реализация многопоточных приложений для расчетов по заданным математическим моделям и визуализации решений с использованием моделей компьютерной графики (градиентные методы для задачи поиска экстремума, численные методы решения краевых задач для 2D уравнения теплопроводности и уравнения Лапласа).

Для компактного интегрированного курса очень важна хорошая информационная поддержка. Помимо авторских материалов (кратких конспектов, презентаций лекций и заданий для проектов) студентами использовались лекции и тесты Интернет-университета информационных технологий (www.intuit.ru), лабораторные работы Сибирского суперкомпьютерного центра для студентов НГУ, методические материалы и лабораторные работы для студентов МГУ, ННГУ, СПбГУ, а также аутентичные технические руководства и описания стандартов технологий PDC от фирм-вендоров и консорциумов (CISCO Systems, Intel, Microsoft, NVIDIA, MPI, OpenMP Architecture Review Board и др.)

Опыт чтения такого курса показал многоплановость его интегративных возможностей. С одной стороны, он позволяет оценить качество содержания и преподавания многих дисциплин. В процессе обучения с очевидностью выясняется, на каком уровне изучались и как были усвоены студентами базовые понятия архитектуры компьютеров, операционных систем, сетевых технологий, семантика языков и моделей программирования, теория компиляции, алгоритмы компьютерной обработки данных, модели и методы дискретной и вычислительной математики. С другой стороны, нацеленность на получение конкретных результатов при реализации проектов, проведение вычислительных экспериментов, работа над пользовательским интерфейсом помогает студентам восстановить и по-новому осмыслить материал фундаментальных математических дисциплин, теоретических основ информатики и информационных технологий в це-

лом. Не менее важен и следующий момент: многие студенты к этому времени уже профессионально работают, как правило, в области Web-программирования, 1С-программирования, сетевого администрирования. Поскольку изучение PDC не является для них принципиально важной задачей, именно по результатам изучения данного курса очень четко проявляются стремление и способности студентов к саморазвитию.

Литература

1. NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing. URL: <http://www.cs.gsu.edu/~tcpp/curriculum/index.php> (дата обращения: 14.06.2015).
2. John D. J., Thomas S. J. Parallel and Distributed Computing across the Computer Science Curriculum //Parallel & Distributed Processing Symposium Workshops (IPDPSW), 2014 IEEE International. – IEEE, 2014. P. 1085-1090.