

МОДЕЛИРОВАНИЕ НЕЙТРОННО-ФИЗИЧЕСКИХ ПРОЦЕССОВ В ЯДЕРНЫХ РЕАКТОРАХ НА ПАРАЛЛЕЛЬНЫХ КОМПЬЮТЕРАХ С РАСПРЕДЕЛЕННОЙ ПАМЯТЬЮ

А.В. Воронков, А.С. Голубев, Н.А. Коновалов, В.А. Крюков, Е.П. Сычугова
Институт прикладной математики Российской Академии Наук, Россия, Москва

Начиная с 1985 года в ИПМ им. М.В. Келдыша все нейтронно-физические программы работают в рамках унифицированного пакета прикладных программ РЕАКТОР [1]. Первоначально пакет РЕАКТОР использовался только на однопроцессорных ЭВМ, что ограничивало область его применения. Когда новые параллельные системы стали доступными, возникла проблема создания соответствующего прикладного и системного программного обеспечения, эффективно использующего потенциальную возможность параллельных систем. В настоящее время пакет РЕАКТОР пополнился новыми параллельными вычислительными модулями. Эта новая версия пакета (РЕАКТОР-Р) предназначена для решения наиболее сложных проблем, требующих высокой эффективности и увеличенный размер памяти. Пакет РЕАКТОР-Р выполняется на многопроцессорной ЭВМ с распределенной памятью МВС-100 [2] с процессорами I860-XP.

Для решения сложных научных проблем на ЭВМ с распределенной памятью используется известное приближение, основанное на описании алгоритма, как набора последовательных процессов, взаимодействующих через передачу сообщений. При таком приближении для построения параллельных алгоритмов могут использоваться два основных подхода. Один из них заключается в использовании специальных библиотек, расширяющих стандартные языки (например ФОРТРАН или С) средствами передачи сообщений. Другой подход использует новые языки параллельного программирования, которые являются стандартными языками, расширенными новыми языковыми конструкциями.

Возникновение библиотеки MPI (Message Passing Interface) является существенным шагом в направлении стандартизации средств, развивающих первый подход. Возможность использования этой библиотеки на различных ЭВМ с параллельной архитектурой позволяет создавать параллельные программы, переносимые с одной ЭВМ на другую. Сравнительно низкий уровень библиотечных средств (близкий к структуре систем с распределенной памятью) позволяет создавать эффективные параллельные программы для таких систем. В настоящее время подмножество библиотеки MPI внедрено в ИПМ в виде программной системы Router. Функциями Router являются команды вызова библиотечных подпрограмм.

Некоторые результаты эффективности таких подходов получены на примере решения в пакете РЕАКТОР-Р тестовой задачи. Задача основана на экспериментах, выполненных в Великобритании на критической сборке ЗЕБРА [3]. Были проведены расчеты этой задачи на примере решения уравнения переноса в многогрупповом диффузионном приближении в трехмерной X-Y-Z геометрии.

Тестовая задача решена, используя ФОРТРАН-77 совместно с библиотекой MPI, погруженной в систему Router. Распараллеливание решения уравнения диффузии в каждой энергетической группе выполняется посредством декомпозиции расчетной области на подобласти по вертикальной оси. Каждая подобласть состоит из нескольких плоскостей и обрабатывается одним процессором. Обмен информацией заключается в обмене значениями нейтронных потоков, расположенных на границах подобластей.

При решении этой задачи в каждой энергетической группе наименее распараллеливаемой частью программы является та ее часть, которая управляет сходимостью внутренних итераций. Алгоритм заключается в том, что один из процессоров (главный) получает информацию относительно локальной сходимости в каждой из подобластей, анализирует ее и решает, продолжить ли внутренние итерации или нет. Имеются две возможности организовать этот процесс на наборе процессоров. В первом случае один из локальных процессоров действует как

главный процессор. Другой метод распараллеливания состоит в том, что дополнительный процессор выполняет функции главного процессора. Основная идея в этом случае заключается в том, чтобы совместить по времени расчет внутренних итераций с анализом сходимости предыдущей итерации. Такой подход допускает одну дополнительную внутреннюю итерацию в каждой группе. Это приближение понижает эффективность распараллеливания в случае малого числа процессоров. Но это приближение является заманчивым в случае систем с массовым параллелизмом.

Сравнительный анализ показывает, что для 17 и более процессоров эффективность распараллеливания во втором случае выше, чем в первом. Однако эффективность снижается в обоих случаях, когда число процессоров растет.

Опыт развития пакета РЕАКТОР-Р на языке ФОРТРАН-77 с MPI библиотекой подтверждает известное мнение: приближение передачи сообщений имеет существенные недостатки. Во-первых, уровень языковых средств, поддерживающих параллелизм программ слишком низок. Во-вторых, эффективное выполнение программы на компьютерах с массовым параллелизмом требует балансировки загрузки, что чрезвычайно трудно обеспечить в модели передачи сообщений. Необходимо разрабатывать новые методы и инструментальные средства для создания переносимых и эффективных параллельных программ для компьютеров с распределенной памятью.

Новая модель параллельного выполнения программы была предложена в 1994 году [4]. Эта модель (DVM-модель) объединяет главные преимущества моделей PCF Fortran и HPF и позволяет выразить функциональный параллелизм и распараллеливание данных в научных и технических приложениях на компьютерах с массовым параллелизмом.

Язык C-DVM был разработан на основе DVM-модели. Главный принцип развития языка: программист должен иметь возможность определить точно параллельное выполнение его программы, но компилятор высокого уровня не требует, чтобы пользователь делал это. Язык C-DVM разрабатывался одновременно и совместно с пакетом программ РЕАКТОР-Р. В результате DVM-модель была улучшена значительно. Две экспериментальные версии компилятора были выполнены, и спецификация языка была пересмотрена дважды. В результате была создана система программирования C-DVM.

В докладе приводится краткое описание этой системы программирования. Приводятся результаты сравнения эффективности расчета эксперимента ZEBRA с использованием языка C-DVM.

Эта работа выполнена при финансовой поддержке проекта МНТЦ # 115-95 и РФФИ N 98-01-00089.

Литература

1. A.V. Voronkov, V.I. Arzhanov, "REACTOR – Program System for Neutron-Physical Calculations", *Proceedings of the International Topical Meeting "Advances in Mathematics, Calculation and Reactor Physics"*, 5, 30.5 1-1 - 30.5 1-4, ANS, Pittsburgh, USA (1991).
2. A.V. Zabrodin, V.K. Levin, V.V. Korneev, "The Massively Parallel Computer System MBC-100", *Proceedings of the Third International Conference on Parallel Computing Technologies*, 341-355, Springer, St. Petersburg, Russia (1995).
3. A.N. Chebeskov, I.V. Krivitskij, G.V. Matveev, Y.N. Miranovich, A.V. Voronkov, E.P. Sychugova, A.D. Knipe "Low reactivity sodium-void benchmark study in an annular heterogeneous assembly", *Proceedings of the International Topical Meeting Sodium cooled fast reactor Safety*, 1, 60-70, IPPE, Obninsk, Russia (1994).
4. N.A. Konovalov, V.A. Krukov, S.N. Mihailov and A.A. Pogrebtsov, "Fortran DVM - a Language for Portable Parallel Program Development", *Proceedings of Software For Multiprocessors & Supercomputers: Theory, Practice, Experience*, 124-133, Institute for System Programming, RAS, Moscow (1994).