О РЕАЛИЗАЦИИМЕЗОМАШТАБНОЙ АТМОСФЕРНОЙ МОДЕЛИ НА АРХИТЕКТУРЕ CELL

Д.Н. Микушин

Региональные совместные модели атмосферы и деятельного слоя суши являются важным средством прогноза погоды и анализа климатических изменений. Дальнейшее увеличение пространственного разрешения и усложнение физических параметризаций в таких моделях возможно лишь в сочетании с использованием мощных многопроцессорных вычислительных комплексов. Однако, некоторые часто используемые в этих моделях численные алгоритмы (например, для решения уравнения Пуассона) не обеспечивают достаточной масштабируемости на традиционных кластерных системах. По этой причине представляется целесообразным изучить возможности использования перспективных архитектур, таких как Cell Broadband Engine (CBEA) и GPGPU, в области численного моделирования атмосферных процессов [1,2]. В данной работе представлен подход к реализации региональной модели для архитектуры CBEA, ориентированный на использование в гибридных MPI-Cell системах, таких как кластер Roadrunner. Подход применим к циклам с независимыми итерациями, соответствующим явным численным схемам, используемым для решения основных уравнений модели. Увеличение производительности в 11-13 раз достигается за счет распределения вычислительной нагрузки между синергетическими процессорными элементами (SPU).

В работе рассматривается мезомасштабная модель, основанная на оригинальном последовательном коде модели NH3D [3] и развиваемая в НИВЦ МГУ. Наряду с системой трехмерных негидростатических уравнений в *sigma*-системе координат используются также параметризация турбулентных процессов, коротковолновой и длинноволновой радиации [4,5], блок деятельного слоя суши [6,7] и внутренних водоемов [8]. Модель включает уравнение для переноса пассивной примеси, приближенное решение которого вычисляется по монотонной схеме Смоларкевича [9].

Причина низкой эффективности MPI-реализаций атмосферных моделей состоит главным образом в недостаточной скорости обмена данными. Производительность ограничена так называемой «стеной памяти» – пропускной способностью RAM, составляющей около 7-10 Гб/сек, и на несколько порядков меньшей скоростью обменов между узлами — 1.5-4 Мб/сек. Архитектура Cell как правило обеспечивает доступ к локальной XDR-памяти со скоростью не менее 20 Гб/сек и 4-6 Гб/сек — к нелокальной (через BIF-шину) [10], так что производительность отдельного устройства ограничена в большой степени частотой работы процессоров. С другой стороны, распределяемые данные должны помещаться в локальную память SPU, размеры которой составляют лишь 256 Кб. По этой причине MPI-подобное полное распределение данных невозможно для реальных приложений с большими сетками. Расчетная область должна быть разделена на число подобластей, превышающее количество SPU, и каждый SPU должен обработать несколько подобластей, последовательно загружая исходные данные и выгружая результаты в общую память для каждой подобласти.

Описанный подход к программированию CBEA практически неприменим в случае уже существующих больших численных моделей, изначально построенных для последовательных расчётов. Если вычислительная нагрузка в приложении распределена равномерно, то ручное распараллеливание в случае нескольких сотен циклов становится слишком накладным. Для вычислительных циклов последовательного кода на языке Фортран возможно предложить средство автоматического преобразования в параллельное представление для Cell, основанное на поиске фрагментов кода, удовлетворяющих определенному характерному шаблону циклов, применяемых в численных моделях. Вместе с фрагментом кода необходимо выделить зависимости по входным и выходным данным, классифицируя переменные как массивы или скаляры. Скалярные данные загружаются на SPU единовременно, единственный раз для каждого цикла, а массивы пересылаются небольшими фрагментами, соответствующими нескольким итерациям цикла. При обмене данными в массивах используется двойная буферизация: во время расчета с уже загруженной порцией данных осуществляется пересылка следующей.

Вызов *spu_program_load*, доступный в Cell SDK, позволяет загружать в память SPU отдельно скомпилированную программу. На этой возможности основана фрагментарность параллельной реализации: для каждого цикла компилируется отдельный исполняемый образ, который загружается в SPU по мере необходимости, например, по результатам динамического анализа программы.

Программа для SPU для заданного цикла представляет собой подпрограмму с оригинальным исходным кодом и подпрограмму-интерфейс, управляющую взаимодействием с PPU и обменом данными. В соответствии с типичной схемой организации SPU-программы, указатель на массив адресов используемых данных поступает в точку входа, и по нему сразу же загружается весь массив, а затем управление передается подпрограмме-интерфейсу.

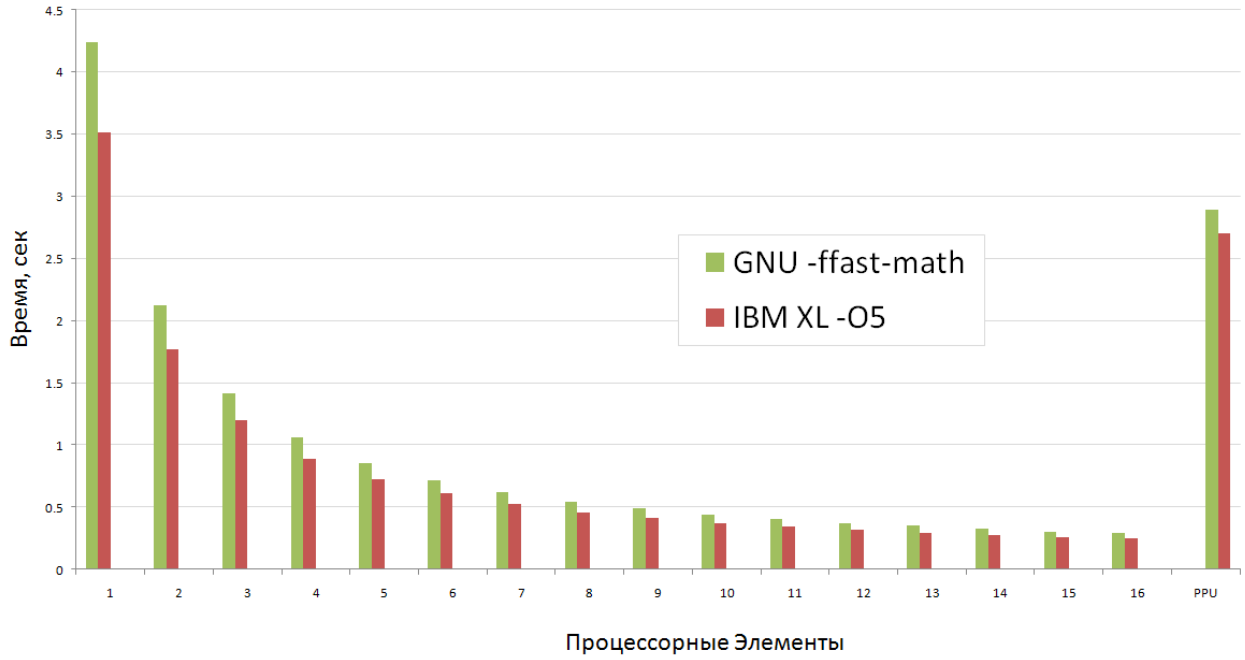

Рис. 1. Производительность компонента ElliptFR.

Для того, чтобы протестировать производительность Cell-реализации, основанной на изложенном решении, были выбраны две компоненты атмосферной модели. Первая (ElliptFR) вычисляет правую часть эллиптического уравнения для геопотенциала, вторая (Leap-frog) представляет собой численную схему «чехарда» для уравнения адвекции. Подпрограмма ElliptFR использует 28 входных массивов, Leapfrog – 10. Размер пространственной сетки составляет 128x128x21 узлов. Время, затраченное на расчет значений прогностических переменных в узлах сетки на одном шаге по времени при различном числе доступных процессорных элементов, показаны на рис. 1 и 2. Были использованы сервер IBM QS22 с 16 SPU, Cell SDK 3.0, компиляторы GNU 4.1 и XL 10.1. Все вычисления выполнены с двойной точностью.

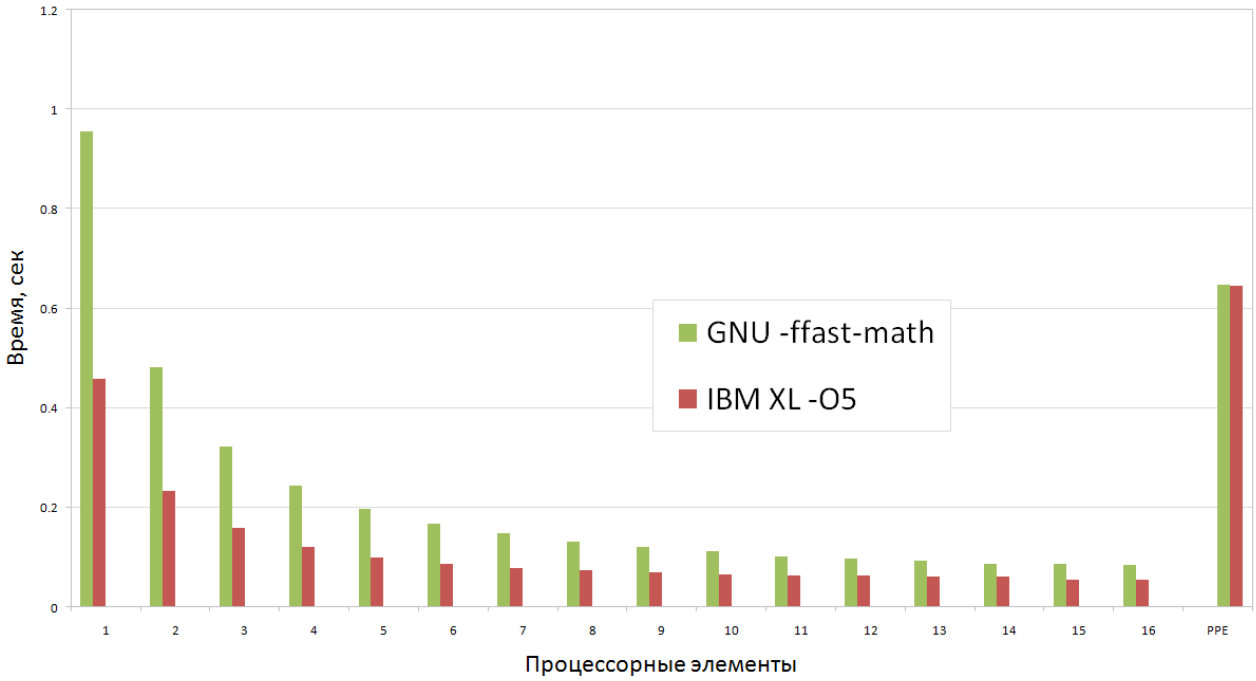

Рис. 2. Производительность компонента Leapfrog.

Полученные результаты демонстрируют достаточно высокую эффективность архитектуры CBEA при решении интенсивных по данным тестовых задач. Одна Cell-плата с 8 SPU обеспечивает близкое к линейному ускорение по отношению к последовательной реализации. Та же самая задача на обычном процессоре не получает ускорения при использовании нескольких ядер по причине наличия «стены памяти». В ходе

дальнейшей работы планируется рассмотреть, насколько применение описанного подхода может увеличить производительность слабо масштабируемых MPI-приложений при переходе на гибридные MPI-Cell системы.

Автор выражает благодарность компании «Т-Платформы» за предоставленный доступ к Cell-серверам.

ЛИТЕРАТУРА:

1. Zhou, S., Duffy, D., Clune, T., Williams, S., Suarez, M., Halem, M. Accelerate Climate Models with the IBM Cell Processor, American Geophysical Union, Fall Meeting 2008, abstract #IN21C-02
2. Michalakes, J., Vachharajani, M. GPU acceleration of numerical weather prediction, Parallel and Distributed Processing, IEEE International Symposium, 14-18 April 2008, pp. 1-7
3. Miranda P.M.A., James I.N. Non-linear three-dimensional effects on gravity wave drag: splitting flow and breaking waves, Quart. J.R. Met. Soc. 1992, Vol. 118, pp. 1057-1082
4. Chou M.-D., Suarez M.J., Liang X.Z., Yan M.M.-H. A thermal infrared radiation parameterization for atmospheric studies: Technical Report Series on Global Modeling and Data Assimilation, NASA/TM-2001-104606, 2003, Vol. 19, 55 p.
5. Chou M.-D., Suarez M.J. A solar radiation parameterization for atmospheric studies: Technical Report Series on Global Modeling and Data Assimilation, NASA/TM-1999-10460, 2002, Vol. 15, 42 p.
6. Mahfouf, A. O. Manzi, J. Noilhan, H. Giordani, and M. Deque. The land surface scheme ISBA within the Meteo-France Climate Model ARPEGE. P.1: Implementation and preliminary results. J. of Climate, Vol. 8, 1995, pp. 2039-2057
7. Е. М. Володин, В. Н. Лыкосов В.Н. Параметризация процессов тепло- и влагообмена в системе растительность - почва для моделирования общей циркуляции атмосферы. 1. Описание и расчеты с использованием локальных данных наблюдений, Известия РАН. Физика атмосферы и океана, 1998, т. 34, 453-465;
8. В. М. Степаненко, В. Н. Лыкосов. Численное моделирование процессов тепловлагопереноса в системе водоем - грунт. - Метеорология и гидрология, 2005, №3, с. 95-104.
9. В.М. Степаненко, Д. Н. Микушин. Численное моделирование мезомасштабной динамики атмосферы и переноса примеси над гидрологически неоднородной поверхностью. - «Вычислительные технологии», 2008, т. 13, ч. 3, стр. 103-110.
10. Altevogt P., Boettiger H., Kiss T., Krnjajic Z. Evaluating IBM BladeCenter QS21 hardware performance. IBM Multicore Acceleration Technical Library, 2008, http://www.ibm.com/developerworks/library/pa-qs21perf/index.html