

МОДЕЛИРОВАНИЕ ПРОЦЕССОВ ВОЗДУШНОГО ОХЛАЖДЕНИЯ ПРОЦЕССОРА

С.В. Поляков, М.В. Яковлевский

В вычислительных модулях современных многопроцессорных систем достаточно часто используется воздушное охлаждение процессоров. Охлаждающий элемент представляет собой пассивный радиатор, продольно обдуваемый постоянным потоком воздуха. Интенсивность потока воздуха определяется энергетическими балансными соображениями, но максимальная температура в точке контакта радиатора и процессора существенно зависит от геометрии самого радиатора. Широко используются пластинчатые радиаторы (Рис. 1), основными параметрами которых, при заданных внешних размерах, являются материал (как правило, медь), число пластин, их толщина и высота.

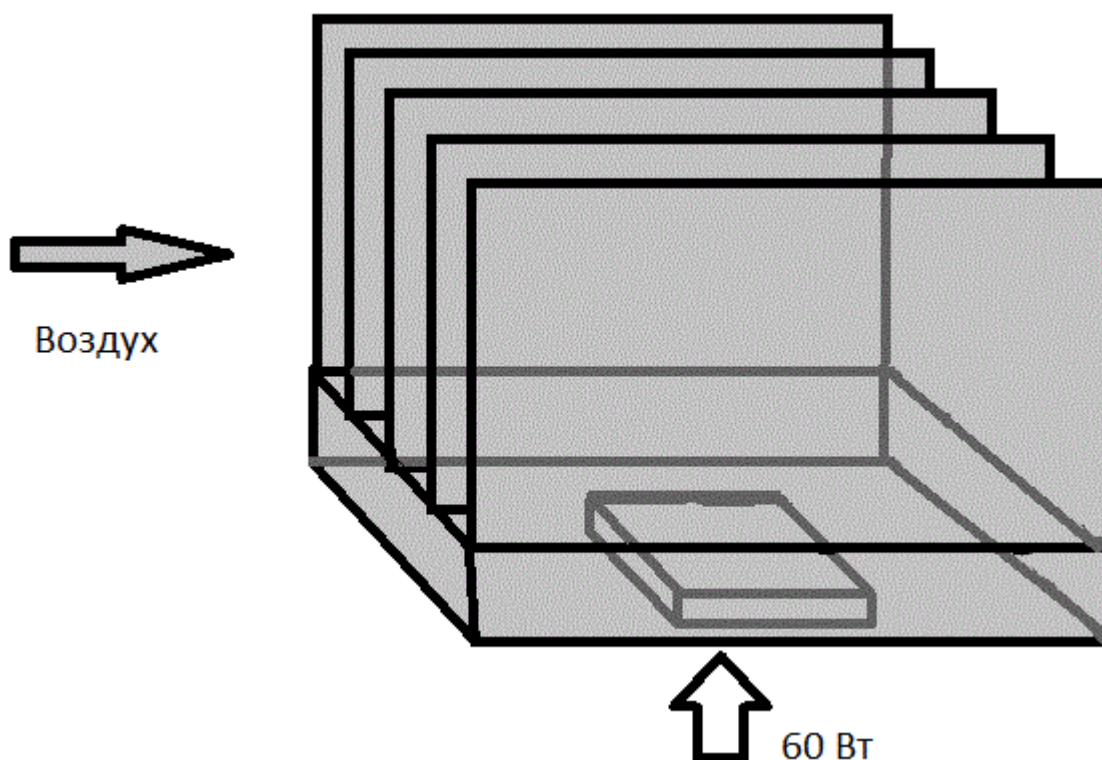


Рис. 1. Геометрия моделируемой области

В рамках представляемой работы на примере решения задачи определения параметров радиатора, обеспечивающих заданный тепловой режим (максимальную температуру на процессоре), изучались возможности современных многоядерных суперкомпьютеров. Структура используемого вычислительного алгоритма вполне соответствует широко используемым для численного моделирования физических процессов явным разностным схемам. Аналогичную структуру имеют алгоритмы, реализующие многие итерационные методы. Разработанные программы численного моделирования тепловых процессов системы охлаждения явно учитывают специфику современных многопроцессорных вычислительных систем, имеющих в своём составе множество узлов, оснащенных многоядерными процессорами. Взаимодействие процессов, запущенных на разных вычислительных узлах обеспечивается с помощью технологии MPI, а взаимодействие процессов в пределах одного вычислительного узла – с помощью базовых средств UNIX – библиотеки pthreads. В качестве математической модели тепловых процессов использовано приближение конвекции-диффузии, содержащее трехмерные уравнения для температуры в стационарном воздушном потоке и в радиаторе.

Численное моделирование показало, что для того, что бы при заданной рассеиваемой мощности 60 Вт, температура на процессоре не поднималась выше заданного уровня в 70 град С, при заданных размерах радиатора, необходимо не менее 25 пластин. Рекомендуемой является конфигурация с количеством пластин 75 и более. Соответствующая задача рассматривалась совместно со специалистами Российского научного центра «Курчатовский институт» и компании HP. Реализованная в технологическом цикле конфигурация с числом ребер 78 попадает в данный диапазон. Для описания подобных конфигураций использовались неравномерные прямоугольные решетки, содержащие $800 \times 700 \times 120$, $1000 \times 3500 \times 150$ и $2000 \times 3500 \times 150$ узлов. Последняя из упомянутых сеток является избыточной с точки зрения физики изучаемого явления, и использована для демонстрации возможности получения практически 100% эффективности использования вычислительной

мощности большого числа вычислительных ядер (вплоть до 7728). Отметим, что при таком числе ядер на каждое из них приходится порядка 135000 узлов. Ввод-вывод, хранение и визуализация соответствующих объемов данных представляет определенную проблему. В рамках настоящей работы была использована библиотека ввода-вывода с контролируемой точностью вещественных сеточных функций и система удаленной интерактивной визуализации RemoteViewer [1]. На рис. 2 представлен результат визуализации с её помощью изоповерхности, соответствующей температуре 20 град. С.

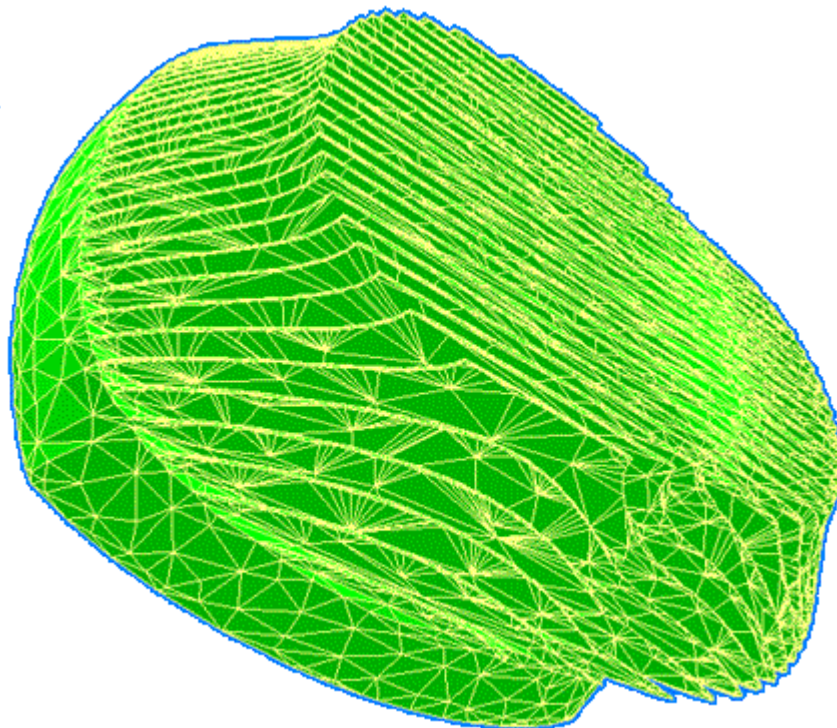


Рис. 2. Огрублённая изоповерхность температуры 20°C. Конфигурация - 50 ребер на радиаторе, сетка $800 \times 700 \times 120 = 67,2$ млн.

Время чтения больших объемов сеточных данных составляет существенную часть общего времени их обработки в ходе визуализации (равно как и время записи в ходе вычислений). В проведенных экспериментах оно составляло десятки минут. Для его сокращения можно прибегнуть к сжатию данных перед их записью на диск, однако, использование стандартных методов сжатия без потерь, применительно к вещественным числам, неэффективно. Для увеличения степени сжатия данных, предназначенных для их анализа методом визуализации, допустимо игнорировать часть младших разрядов мантиссы каждого из значений сеточной функции, что значительно увеличивает коэффициент компрессии. Относительная ошибка огрубления данных, полученная в результате отбрасывания k двоичных разрядов мантиссы float (REAL4) числа, приближенно может быть оценена как $2^{-(k-24)}$. Использование иерархической двухуровневой схемы хранения [2] позволяет задавать параметр огрубления независимо для каждого из записываемых блоков, что позволяет записывать значения сеточной функции, соответствующие разным фрагментам сетки, и независимо с разных процессоров, и с разной точностью.

На рис. 3 представлены достигнутые коэффициенты сжатия с огрублением сеточных данных, полученных при численном моделировании системы охлаждения на сетке $1000 \times 3500 \times 150 = 525$ млн узлов с помощью библиотеки [2]. Результаты, представленные на рис. 2-3 показывают, что объем хранимых с целью дальнейшей визуализации данных может быть в ряде случаев сокращен в сотни и более раз без существенной потери точности их представления. Данный подход обеспечивает возможность сокращения на несколько порядков времени ввода-вывода данных и требований к объемам дисковой или оперативной памяти (в приведенном примере размер файла уменьшен в 1438 раз – с двух гигабайт до 1.5 мегабайт), время чтения и записи компрессированного файла является несущественным.

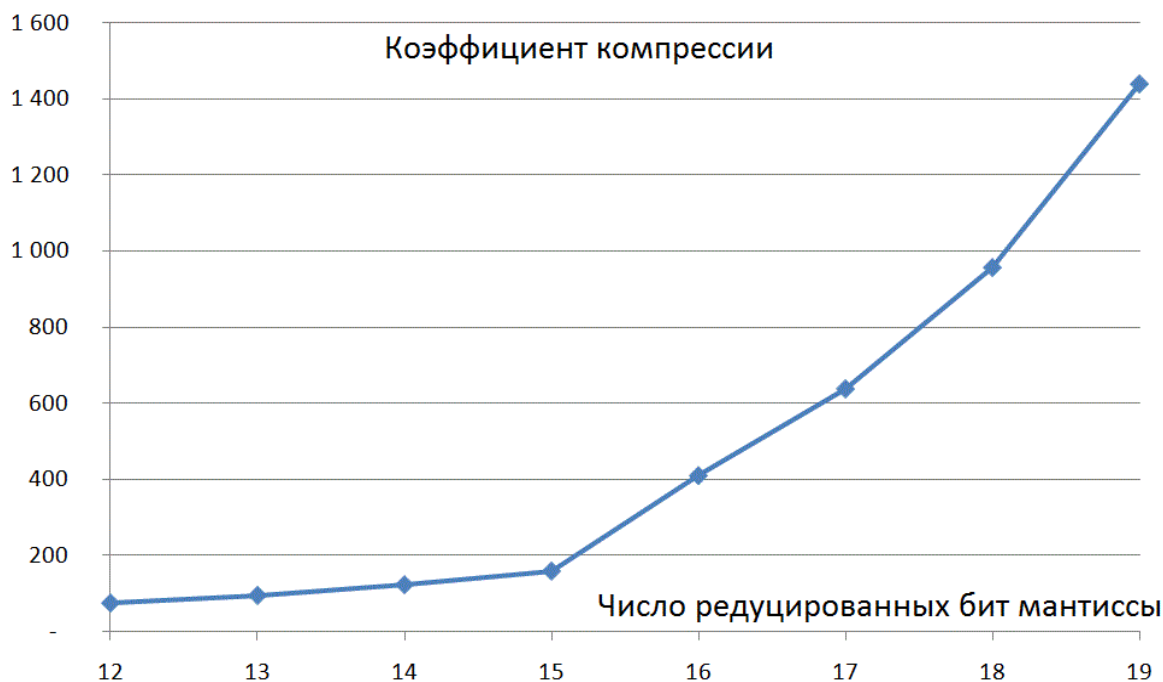


Рис. 3. Зависимость коэффициента сокращения размера файла результатов моделирования теплового режима устройства охлаждения от числа отброшенных бит мантиисы 4х-байтовых float чисел

На рисунке 4 приведены результаты замера эффективности вычислений на суперкомпьютере «Ломоносов», полученные при запуске программы на разном числе процессорных узлов и разном числе ядер. Участок сверхлинейного ускорения соответствует запуску на каждом из вычислительных узлов только одного процесса. Формально при этом семь из восьми ядер каждого узла простаивают и совершенно не очевидно, сколько из них и как принимают участие в выполнении одного MPI процесса. При запуске большего числа процессов на каждом из вычислительных узлов эффективность снижается до уровня существенно ниже 100%. Максимальное ускорение и, практически, 100% эффективность были получены после дополнения программы средствами привязки каждого из процессов к конкретным ядрам вычислительного узла (Таблица 1).

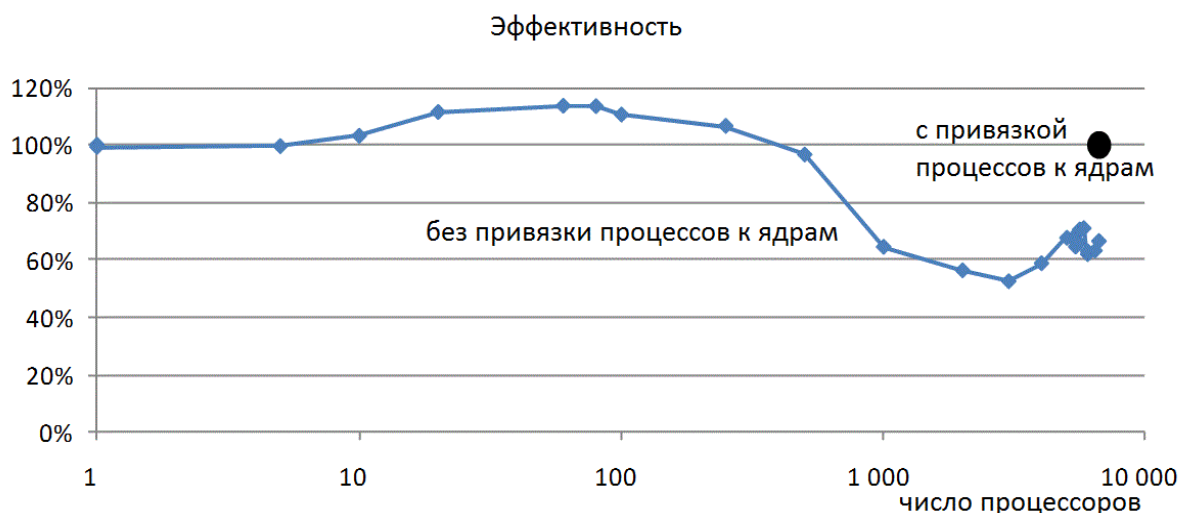


Рис. 4. Зависимость эффективности вычислений на суперкомпьютере «Ломоносов» от числа ядер.

Конфигурация - 78 тонких ребер (толщина ребра 0.3 мм). Сетка: 1000 x 3500 x 150 = 525 млн.

Параллельная реализация: MPI + нити (8 нитей на узле).

Соответствующие процедуры доступны и для привязки MPI-процессов, и для привязки нитей. Отметим, что сам процесс привязки процессов к ядрам носит случайный характер, в том смысле, что однократная привязка, для ряда процессов, оканчивается неудачно. В проделанных экспериментах для достижения положительного результата требовалось не более 100 выполненных подряд попыток привязки.

Таблица 1. Ускорение и эффективность параллельной программы при выполненной привязке процессов к ядрам.

Сетка, узлов	Ядер	Время шага, сек	Ускорение	Эффективность
--------------	------	-----------------	-----------	---------------

525000000	1	01.11.67	1	100.00 %
525000000	7728	0.00161	7274.7	94.13 %
1050000000	1	24.285	1	100.00 %
1050000000	7728	0.003161	7683.72	99.43 %

Выполненные расчеты подтвердили высокую эффективность разработанных программных средств и высокие характеристики предоставленной для экспериментов части вычислительной системы «Ломоносов».

Работа выполнена при поддержке проектов РФФИ № 08-07-00458-а, 09-01-12022-офи_м.

ЛИТЕРАТУРА:

1. Iakobovsky M.V. Karasev D.E. Krinov P.S. Polyakov S.V. Visualisation of grand challenge data on distributed systems. // Mathematical Modeling. Problems, Methods, Applications 2001 Kluwer Academic/Plenum Publishers. New York, Boston, Dordrecht, London, Moscow.
2. Якововский М.В. Ввод-вывод сеток и сеточных функций <http://lira.imamod.ru/IOlibs.html>