

Лекционно-практический курс
«Суперкомпьютерные вычислительные технологии»

сентябрь- декабрь 2012 г.

Лекторы: Н.Н.Попова, В.А.Бахтин

Задание 3.

Численное решение 3-мерной задачи Дирихле.

Методы Якоби, SOR.

Разработка параллельной MPI-программы и исследование ее эффективности.

Постановка задачи.

Разработать параллельную программу с использованием технологии MPI, реализующую решение 3-хмерной Дирихле задачи методом Якоби (или SOR). Провести исследование разработанной программы эффективности на системах Regatta, Blue Gene/P и «Ломоносов».

Провести визуализацию полученного решения. Для этого требуется организовать в параллельной программе вывод решения в файл, формат которого будет соответствовать используемой системе визуализации.

Описание методов Якоби и SOR и параллельная программа для решения 2-ухмерной задачи представлены в материалах лекций. За основу реализации можно взять разработанную во втором задании параллельную программу для решения 2-ухмерной задачи.

Параметры, передаваемые в командной строке

Первый параметр: m – число точек по одному измерению для задания двумерной сетки. По умолчанию – 512.

Второй параметр – точность. По умолчанию – 0.01.

Цель.

Получить навыки разработки и исследования параллельных программ с использованием технологии MPI.

Распараллеливание осуществляется на основе имеющегося примера реализации 2-ухмерной задачи.

Постановка задачи.

Формальная модель и численный метод решения задачи.

Необходимо реализовать численное решение задачи Дирихле для уравнения Лапласа (1):

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = 0, (x, y, z) \in D, \\ u(x, y, z) = g(x, y, z), (x, y, z) \in D^0, \end{cases} \quad (1)$$

где $u(x, y, z)$ - функция, удовлетворяющая в области D уравнению Лапласа и принимающая на границе D^0 области D значения $g(x, y, z)$.

Область D задается условиями:

$$0 \leq x \leq 1, 0 \leq y \leq 1, 0 \leq z \leq 1$$

Граничные условия:

$$u(x, y = 0, z) = 0$$

$$u(x, y = 1, z) = 0$$

$$u(0, y, z) = (a + bz) \sin \pi y$$

$$u(1, y, z) = (a + bz) e^{\pi} \sin y$$

$$u(x, y, z = 0) = a e^{\pi} \sin \pi y$$

$$u(x, y, z = 1) = (a + b) e^{\pi} \sin \pi y$$

Сравнить точное решение $u = (a + bz) e^{\pi} \sin \pi y$ с полученным разработанным методом решением.

Варианты выбора значений а и b:

$$1) \ a=1, b=1$$

$$2) \ a=1, b=0$$

$$3) \ a=1, b=1$$

Численный подход к решению задачи (1) основан на замене производных соответствующими конечными разностями (2).

$$\frac{u_{i+1,j,k} - 2u_{i,j,k} + u_{i-1,j,k}}{h^2} + \frac{u_{i,j+1,k} - 2u_{i,j,k} + u_{i,j-1,k}}{h^2} + \frac{u_{i,j,k+1} - 2u_{i,j,k} + u_{i,j,k-1}}{h^2} = f_{i,j,k}, \quad (2)$$

где $h > 0$ - шаг сетки, $u_{i,j,k}$ - значение функции $u(x, y, z)$ в точке $x = x_i = ih, i = 0, M+1, y = y_j = jh, j = 0, N+1, z = z_k = kh, k = 0, L+1$, где M, N, L - количество внутренних узлов по каждой координате в области D .

Одним из простейших методов решения полученной системы (2) является итерационный метод Якоби (3).

$$u_{i,j,k}^n = (u_{i-1,j,k}^{n-1} + u_{i+1,j,k}^{n-1} + u_{i,j+1,k}^{n-1} + u_{i,j-1,k}^{n-1} + u_{i,j,k+1}^{n-1} + u_{i,j,k-1}^{n-1}) / 6 \quad (3)$$

$$u_{i,j,k}^n = g_{i,j,k}, (x, y, z) \in D^0, n = 1, 2, \dots \text{ где } n - \text{ номер итерации.}$$

Подробное описание трехмерного метода SOR можно найти в [5]. Статья прилагается к материалам лекций.

Требуется.

1. Разработать параллельную версию программы с использованием технологии MPI.
2. Исследовать время выполнения разработанной программы в зависимости от задаваемой точности, размера сетки и количества используемых процессов на вычислительных системах IBM Regatta, Blue Gene/P и «Ломоносов».
3. Для каждой из платформ для заданных значений точности метода (0.01, 0.001) Построить таблицу:

Для вычислительной системы IBM Regatta:

Размер сетки	Точность	Параллельный алгоритм											
		1 процессор			2 процессора			4 процессора			8 процессоров		
		Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций
512x512x512													
1024x1024x1024													

Для вычислительная система Blue Gene/P :

Размер сетки	Точность	Параллельный алгоритм											
		128 процессора			256 процессоров			512 процессоров станд. мэппинг			512 процессоров Произв. мэппинг		
		Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций
512x512x512													
1024x1024x1024													

Для вычислительная система «Ломоносов»:

Размер сетки	Точность	Параллельный алгоритм											
		8 процессов			32 процесса			64 процесса			128 процесса		
		Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций	Время	Ускорение	Число итераций
512x512x512													
1024x1024x1024													

В случае исследования эффективности параллельной программы на Blue Gene/P для 512 процессоров рассмотреть два варианта мэппинга – стандартный, принятый по умолчанию и произвольный. Для произвольного мэппинга предусмотреть генерацию строк файла для задания случайного значения XYZT (см. материалы лекций).

Графическую иллюстрацию полученного решения - линии уровня функции $u(i,j)$ провести, используя систему визуализации по собственному выбору.

Ускорение (*speedup*), получаемое при использовании параллельного алгоритма для p процессоров, определяется величиной:

$$\text{Speedup}(n) = T_1(n)/T_p(n),$$

где $T_1(n)$ - время выполнения задачи на одном процессоре.

$T_p(n)$ - время параллельного выполнения задачи при использовании p процессоров.

4. Построить графики – для каждого из заданных значений точности (0.01, 0.001) зависимость ускорения от количества процессоров для разных размеров сетки (512x512x512, 1024x1024x1024).
5. Подготовить отчет о выполнении задания, включающий таблицы с временами, графики, визуализацию полученного решения (линии уровня решения), текст программы. Сделать выводы по полученным результатам (объяснить убывание или возрастание производительности параллельной программы при увеличении числа используемых процессоров, сравнить поведение параллельной программы в зависимости от размера сетки).

1. Литература.

1. Краткая инструкция по выполнению заданий на системе Regatta Blue и Gene/P (<http://angel.cs.msu.su/~popova/SuperComp2012/Instructions.pdf>)
2. Материалы сайта <http://hpc.cs.msu.su>
3. Инструкция по использованию вычислительного комплекса IBM Regatta <http://www.regatta.cmc.msu.ru/instr.htm>
4. Презентации лекций «Суперкомпьютерные вычислительные технологии. Параллельные алгоритмы численного решения задачи Дирихле». Лекции 4-7. <http://angel.cs.msu.su/~popova/SuperComp2012/>
5. Mathias A. Onabid “Solving three-dimensional (3D) Laplace equations by successive over-relaxation method “ (прилагается)