

Задание по курсу «Суперкомпьютерное моделирование и технологии»

Численное интегрирование многомерных функций методом Монте-Карло

2022

1. Введение

В качестве модельной задачи предлагается задача вычисления многомерного интеграла методом Монте-Карло.

Программная реализация должна быть выполнена на языке C или C++ с использованием библиотеки параллельного программирования MPI.

Требуется исследовать масштабируемость параллельной MPI-программы на следующих параллельных вычислительных системах ВМК МГУ:

1. IBM Polus [2].

2. Математическая постановка задачи

Функция $f(x, y, z)$ — непрерывна в ограниченной замкнутой области $G \subset \mathbb{R}^3$. Требуется вычислить определённый интеграл:

$$I = \iiint_G f(x, y, z) \, dx dy dz$$

3. Численный метод решения задачи

Метод Монте-Карло для численного интегрирования представлен в [1].

Пусть область G ограничена параллелепипедом: $\Pi : \begin{cases} a_1 \leq x \leq b_1 \\ a_2 \leq y \leq b_2 \\ a_3 \leq z \leq b_3 \end{cases}$

Рассмотрим функцию: $F(x, y, z) = \begin{cases} f(x, y, z), & (x, y, z) \in G \\ 0, & (x, y, z) \notin G \end{cases}$

Преобразуем искомый интеграл:

$$I = \iiint_G f(x, y, z) \, dx dy dz = \iiint_{\Pi} F(x, y, z) \, dx dy dz$$

Пусть $p_1(x_1, y_1, z_1), p_2(x_2, y_2, z_2), \dots$ — случайные точки, равномерно распределённые в Π . Возьмём n таких случайных точек. В качестве приближённого значения интеграла предлагается использовать выражение:

$$I \approx |\Pi| \cdot \frac{1}{n} \sum_{i=1}^n F(p_i) \quad (1)$$

где $|\Pi|$ — объём параллелепипеда Π . $|\Pi| = (b_1 - a_1)(b_2 - a_2)(b_3 - a_3)$

4. Программная реализация

Все варианты интегралов, предлагаемые в рамках данного модельного задания, вычисляются аналитически. Требуется реализовать параллельную MPI-программу, которая принимает на вход требуемую точность и генерирует случайные точки до тех пор, пока требуемая точность не будет достигнута. Программа вычисляет точность как модуль разности между приближённым значением, полученным методом Монте-Карло, и точным значением, вычисленным аналитически.

Программа считывает в качестве аргумента командной строки требуемую точность ε и выводит четыре числа:

- Посчитанное приближённое значение интеграла.
- Ошибка посчитанного значения: модуль разности между приближённым и точным значениями интеграла.
- Количество сгенерированных случайных точек.
- Время работы программы в секундах.

Время работы программы измеряется следующим образом. Каждый MPI-процесс измеряет своё время выполнения, затем среди полученных значений берётся максимум.

Предлагается два возможных варианта распараллеливания метода Монте-Карло:

1. Параллельные процессы генерируют случайные точки независимо друг от друга.
2. Парадигма «мастер–рабочие»: один из процессов («мастер») генерирует случайные точки и передаёт каждому из остальных процессов («рабочих») отдельный, предназначенный для него, набор сгенерированных случайных точек.

Все процессы в первом случае и все процессы–рабочие во втором случае вычисляют свою часть суммы в формуле (1). Затем вычисляется общая сумма с помощью операции редукции.

После чего вычисляется ошибка (разность между посчитанным значением и точным значением, вычисленным аналитически). В случае если ошибка выше требуемой точности, которую подали на вход программе, то генерируются дополнительные точки и расчёт продолжается.

В первом варианте параллельной реализации для обеспечения генерации разных последовательностей точек в разных MPI-процессах необходимо инициализировать генератор псевдослучайных чисел (в случае использования стандартного генератора — функцией `srand()`) разными числами.

5. Варианты интегралов

1.

$$I = \iiint_G xy^2z^3 \, dx dy dz,$$

где область G ограничена поверхностями $z = xy$, $y = x$, $x = 1$, $z = 0$.

2.

$$I = \iiint_G \frac{dx dy dz}{(1 + x + y + z)^3}$$

где область G ограничена поверхностями $x + y + z = 1$, $x = 0$, $y = 0$, $z = 0$

3.

$$I = \iiint_G \sqrt{x^2 + y^2} \, dx dy dz$$

где область G ограничена поверхностями $x^2 + y^2 = z^2$, $z = 1$.

4.

$$I = \iiint_G e^{x^2+y^2} z \, dx dy dz$$

где область $G = \{(x, y, z) : z \geq 0, x^2 + y^2 + z^2 \leq 1\}$

5.

$$I = \iiint_G x^3 y^2 z \, dx dy dz$$

где область $G = \{(x, y, z) : -1 \leq x \leq 0, -1 \leq y \leq 0, -1 \leq z \leq 0\}$

6.

$$I = \iiint_G \sin(x^2 + z^2) \cdot y \, dx dy dz$$

где область $G = \{(x, y, z) : x^2 + y^2 + z^2 \leq 1, x \geq 0, y \geq 0, z \geq 0\}$

7.

$$I = \iiint_G \sqrt{y^2 + z^2} \, dx dy dz$$

где область $G = \{(x, y, z) : 0 \leq x \leq 2, y^2 + z^2 \leq 1\}$

8.

$$I = \iiint_G x^2 y^2 z^2 \, dx dy dz$$

где область $G = \{(x, y, z) : |x| + |y| \leq 1, -2 \leq z \leq 2\}$

6. Требования к отчёту

Отчёт должен содержать:

- математическую постановку задачи и численный метод её решения
- нахождение точного значения интеграла аналитически
- краткое описание программной реализации
- исследование масштабируемости программы на системе Polus [2].

Необходимо провести запуски программы на системе Polus для различного числа MPI-процессов и различных значений входного параметра ε в соответствии с таблицей 1, которую нужно заполнить и включить в отчёт. Требуется построить графики зависимости ускорения программы от числа используемых MPI-процессов для каждого значения ε .

Под ускорением программы, запущенной на p MPI-процессах, понимается величина:

$$S_p = \frac{T_1}{T_p}$$

где T_1 — время работы программы на 1 MPI-процессе, T_p — время работы программы на p MPI-процессах.

Таблица 1. Таблица с результатами расчётов для системы Polus

Точность ε	Число MPI-процессов	Время работы программы (с)	Ускорение	Ошибка
$3.0 \cdot 10^{-5}$	1 (для парадигмы «масте-рабочие»: 2)		1	
	4			
	16			
$5.0 \cdot 10^{-6}$	1 (для парадигмы «масте-рабочие»: 2)		1	
	4			
	16			
$1.5 \cdot 10^{-6}$	1 (для парадигмы «масте-рабочие»: 2)		1	
	4			
	16			

7. Варианты заданий

Таблица 2. Варианты заданий

Номер варианта	Вариант интеграла	Метод распараллеливания
1	1	независимая генерация точек MPI-процессами
2	2	независимая генерация точек MPI-процессами
3	3	независимая генерация точек MPI-процессами
4	4	независимая генерация точек MPI-процессами
5	5	независимая генерация точек MPI-процессами
6	6	независимая генерация точек MPI-процессами
7	7	независимая генерация точек MPI-процессами
8	8	независимая генерация точек MPI-процессами
9	1	Парадигма «мастер–рабочие»
10	2	Парадигма «мастер–рабочие»
11	3	Парадигма «мастер–рабочие»
12	4	Парадигма «мастер–рабочие»
13	5	Парадигма «мастер–рабочие»
14	6	Парадигма «мастер–рабочие»
15	7	Парадигма «мастер–рабочие»
16	8	Парадигма «мастер–рабочие»

Список литературы

- [1] Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. ЧИСЛЕННЫЕ МЕТОДЫ. — М.: Наука, 1987.
- [2] IBM Polus. — <http://hpc.cs.msu.su/polus>