计算物理作业八

于浩然　　PB19020634　　2021.10.20

1　作业题目

用 Monte Carlo 方法计算如下定积分，并讨论有效数字位数.

$$\int_0^5 \mathrm{d}x\sqrt{x+2\sqrt{x}}; \tag{1}$$

$$\int_0^{7/10}\mathrm{d}x\int_0^{4/7}\mathrm{d}y\int_0^{9/10}\mathrm{d}z\int_0^{2}\mathrm{d}u\int_0^{13/11}\mathrm{d}v(5-x^2+y^2-z^2+u^3-v^3) \tag{2}$$

2　算法简介

2.1　平均值法计算一维积分

根据积分的平均值定理:

$$\int_a^b f(x)\mathrm{d}x=(b-a)\langle f\rangle \tag{3}$$

而平均值又可从下式得到:

$$\langle f\rangle \approxeq \frac{1}{N}\sum_{i=1}^{N} f(x_i) \tag{4}$$

故有

$$\int_a^b f(x)\mathrm{d}x \approxeq \frac{(b-a)}{N}\sum_{i=1}^{N} f(x_i) \tag{5}$$

此即 Monte Carlo 方法计算一维定积分.

2.2　多重定积分

Monte Carlo 方法真正的威力在于应用于多重积分. 将 (5) 式推广为:

$$\int_{a_1}^{b_1}\mathrm{d}x_1\int_{a_2}^{b_2}\mathrm{d}x_2\cdots\int_{a_n}^{b_n}\mathrm{d}x_n f(x_1,x_2,\cdots,x_n)=\frac{1}{N}\left[\prod_{j=1}^{n}(b_j-a_j)\right]\sum_{i=1}^{N} f(x_1,x_2,\cdots,x_n) \tag{6}$$

其中对每个坐标的抽样值在相应区间范围内均匀抽取.

3 编程实现

用 FORTRAN90 进行编程，将积分程序写在一个模块 `integrate` 中. 其中包含:

- `f(x)` 、`g(x, y, z, u, v)`

  分别用于表示两个被积函数.

- `SUBROUTIHNE integrate_1(n)`

  此子程序为 1 维 Monte Carlo 积分计算程序. 在这里我们调用 16807 生成器程序 `Schrage(P, z0, filename)`，生成 10^n 个均匀分布随机数. `ave` 为计算前 `i` 个数的均值，每当增加一个数时更新均值如下：

$$\bar{x}_N = \bar{x}_{N-1} + \frac{x_N - \bar{x}_{N-1}}{N} \tag{7}$$

  其中 $\bar{x}_N = \frac{1}{N}\sum_{i=1}^{N} x_i$. 在程序中即 23 行所示，我们采用这种方法以避免求和时发生数据溢出.

- `SUBROUTIHNE integrate_1(n)` 此子程序思路与前面基本相同，但使用了 $10^n \times 5$ 维的数组，5 列分别用于存放 `x,y,z,u,v`，具体实现见 38-42 行的循环结构. 数组的 5 个维度出自同一个随机数序列，我们曾在第一道作业题中论证过这些子序列可认为是互不相关的，故这样做与分别产生 5 个随机数序列无异.

```
1  MODULE integrate !聚合两个积分程序的模块
2      IMPLICIT NONE
3      CONTAINS
4          FUNCTION f(x) !定义被积函数
5              REAL(KIND=8) :: f, x
6              f = SQRT(x + 2 * SQRT(x))
7          END FUNCTION f
8          FUNCTION g(x, y, z, u, v)
9              REAL(KIND=8) :: g, x, y, z, u, v
10             g = 5 - x**2 + y**2 - z**2 + u**3 - v**3
11         END FUNCTION g
12
13         SUBROUTINE integrate_1(n) !一维Monte Carlo积分子程序
14             REAL(KIND=8) ,DIMENSION(10**n):: x
15             REAL(KIND=8) :: ave, res
16             INTEGER(KIND=4) :: i, n
17             CALL Schrage(n, 54123654, 'rand.dat') !用16807产
                   生器产生一定数目的均匀随机数
```

```fortran
        OPEN (1, file='rand.dat')
        READ (1, *) x
        CLOSE (1)
        ave = f(5 * x(1)) !将0到1间随机数x转换为0到5间均匀分布随机数
        DO i = 2, 10**n
            ave = ave + (f(5 * x(i)) - ave) / i !为防止溢出调整了平均值求法
        END DO
        res = 5 * ave !积分计算结果
        print *, '1d', res
    END SUBROUTINE integrate_1

    SUBROUTINE integrate_2(n) !五维Monte Carlo积分子程序
        REAL(KIND=8) ,DIMENSION(10**(n + 1)) :: dat
        REAL(KIND=8) ,DIMENSION(10**n, 5) :: x
        REAL(KIND=8) :: ave, res
        INTEGER(KIND=4) :: i, j, n
        CALL Schrage(n + 1, 8455214, 'rand.dat') !用16807产生器产生一定数目的均匀随机数
        OPEN (1, file='rand.dat')
        READ (1, *) dat
        CLOSE (1)
        DO i = 1, 10**n
            DO j = 1, 5
                x(i, j) = dat(5 * i + j) !每隔5个数从前面产生的随机数序列中取一个值，对应5个维度
            END DO
        END DO
        ave = g((7.0/10) * x(1, 1), (4.0/7) * x(1, 2),&
              (9.0/10) * x(1, 3), 2.0 * x(1, 4),&
              (13/11) * x(1, 5))
        !将0到1间均匀随机数变换为0到任意值之间均匀随机数
        DO i = 2, 10**n
            ave = ave + (g((7.0/10) * x(i, 1),&
                  (4.0/7) * x(i, 2),(9.0/10) * x(i, 3),&
                  2.0 * x(i,4),(13/11) * x(i,5))-ave)/i
        END DO
```

```
            res = (7.0/10) * (4.0/7) * (9.0/10) * 2 * (13.0/11) * ave
            print *, '5d', res
        END SUBROUTINE integrate_2
END MODULE integrate
```

- SUBROUTINE Schrage(P, z0, filename)

  16807 生成器子程序.(这是一个外部过程，调用了之前所写的子程序)

```
SUBROUTINE Schrage(P, z0, filename) !Schrage随机数生成器子程序
    IMPLICIT NONE
    INTEGER :: N = 1, P
    INTEGER :: m = 2147483647, a = 16807, q = 127773, r = 2836, In(10**P), z0
    REAL(KIND=8) :: z(10**P)
    CHARACTER(LEN=8) :: filename
    In(1) = z0 !将传入值z0作为种子
    z(1) = REAL(In(1))/m
    DO N = 1, 10**P - 1
        In(N + 1) = a*MOD(In(N), q) - r*INT(In(N)/q)
        IF (In(N + 1) < 0) THEN !若值小于零，按Schrage方法加m
            In(N + 1) = In(N + 1) + m
        END IF
        z(N + 1) = REAL(In(N + 1))/m !得到第N+1个随机数
    END DO
    OPEN (1, file=trim(filename)) !每次运行子程序按照传入参数filename生成数据文件
    DO N = 1, 10**P !将随机数按行存入文件
        WRITE (1, *) z(N)
    END DO
    CLOSE (1)
END SUBROUTINE Schrage
```

在主程序中，使用两个 DO 循环结构，实现对不同抽样点数的两种积分.

```
PROGRAM MAIN
    USE integrate
    INTEGER(KIND=4) :: i
    DO i = 2, 7
        CALL integrate_1(i)
    END DO
    DO i = 2, 7
        CALL integrate_2(i)
    END DO
END PROGRAM MAIN
```

4 计算结果

4.1 一重定积分

将一重定积分程序计算结果列表展示如下：

抽样点数 N	10^2	10^3	10^4	10^5	10^6	10^7
积分值	10.95739	11.35194	11.32440	11.30942	11.32128	11.31761

表 1: 一重定积分计算结果

在 Wolfram Alpha 中计算精确值为 11.31796149(保留小数点后 8 位)，于是我们可讨论有效数字位数.

- $N = 10^2$ 时，只有 1 位有效数字；
- $N = 10^4$ 时，有 2 位有效数字；
- $N = 10^6$ 时，有 3 位有效数字；
- $N = 10^7$ 时，有 4 位有效数字.

从上面规律可以发现，要想提高 1 位数字精度，大概需要抽样数变为原来的 100 倍.

4.2 多重定积分

将五重定积分程序计算结果列表展示如下：

根据所得数据以及前一节中一重定积分有效数字位数的讨论，我们可以得到

抽样点数	10^2	10^3	10^4	10^5	10^6	10^7
积分值	5.29003	5.51398	5.49307	5.46980	5.46718	5.46795

表 2: 五重定积分计算结果

- $N = 10^2$ 时，有 1 位有效数字；
- $N = 10^4$ 时，有 2 位有效数字；
- $N = 10^6$ 时，有 3 位有效数字；
- ($N = 10^8$ 时，有 4 位有效数字.)

这与 Monte Carlo 积分理论上的误差相吻合.

5 结论

本作业中我们用 Monte Carlo 方法计算了积分，显见要想使用 MC 方法达到较高的积分精度需要付出巨大的计算代价. 但是对于高维积分，其他数值积分方法变得难以操作，MC 方法便体现出了其优越性.