数值分析第六次上机练习报告

——数值积分与数值微分

周懿

力 1-2021013053

一、 问题的描述

试用不同数值积分方法计算 $I(f)=\int_1^3 f(x)dx$ 的近似值，其中 $f(x)=\frac{1}{x^2}sin\frac{2\pi}{x}$.
注：$I(f)=-0.238732414637843....$

1. 把 [1,3] 分成 4 个子区间，用五点 Gauss-Legendre 求积公式的复合求积公式计算；
2. 用 Romberg 求积算法计算积分，取 $\varepsilon=10^{-7}$，并与第一种办法比较。

二、 方法描述

(a) **Gauss-Legendre 五点复合积分公式**

我们先给出 Gauss-Legendre 五点积分的算法。记 n 阶 Legendre 多项式为 $P_n(x)$，那么对于 n=5，我们取 $P_n(x)=P_5(x)$ 的零点。Legendre 多项式表如下：

Order	Legendre
0	1
1	x
2	$\frac{3x^2-1}{2}$
3	$\frac{5x^3-3x}{2}$
4	$\frac{35x^4-30x^2+3}{8}$
5	$\frac{63x^5-70x^3+15x}{8}$
6	$\frac{231x^6-315x^4+105x^2-5}{16}$

表 1: Legendre 多项式表

仍然使用求积公式：

$$I(f)=\sum_{k=0}^{n}A_kf(x_k) \tag{1}$$

其中 $\{x_k\}$ 为 $P_5(x)$ 的零点。然后计算 A_k：

$$A_k=\frac{2}{n+1}\frac{1}{P_n(x_k)P'_{n+1}(x_k)} \tag{2}$$

(b)**Romberg 求积算法**

Romberg 求积算法的核心是将梯形公式减半加密网络与 Richardson 外推结合。

我们先给出梯形公式减半加密网络的算法：设积分区间为 [a,b]，记 $h_j = 2^{-j}(b-a)$，也就是减半加密网格的宽度，那么我们有：

$$T_1^{(0)} = \frac{b-a}{2}(f(a)+f(b)) \tag{3}$$

$$T_1^{(1)} = \frac{1}{2}[T_1^{(0)} + h_0 f(\frac{(a+b)}{2})] \tag{4}$$

$$... \tag{5}$$

$$T_1^{(k)} = \frac{1}{2}[T_1^{k-1} + h_{k-1}H_{k-1}] \tag{6}$$

其中

$$H_j = \sum_{l=1}^{2^j} f(a + (l-\frac{1}{2})h_j). \tag{7}$$

然后我们在此基础上进行 Richardson 外推，实际上是对梯形公式进行加速：设 $\varphi(h)$ 在 $h \to 0$ 时可以收敛到 $\varphi(0) = \varphi^*$，那么我们定义新序列：

$$\varphi_1(h) = \varphi(h) \tag{8}$$

$$\varphi_{m+1}(h) = \frac{\varphi_m(qh) - q^{p_m}\varphi_m(h)}{1-q^{p_m}}, m = 1, 2, ... \tag{9}$$

那么 $\varphi_m(h)$ 就可以以更快的速度收敛到 φ^*。在这里我们取 $p_k = 2^k, q = \frac{1}{2}$，于是我们就得到了 Richardson 外推部分的算法：

$$T_{j+1}^{(k-1)} = \frac{4^j T_j^{(k)} - T_j^{(k-1)}}{4^j - 1}, j = 1, 2, ...; k = 1, 2, ... \tag{10}$$

三、 方案设计

我们通过编写 MATLAB 程序来进行线性方程组的求解。

1. main.m：主程序。该程序会生成题目给定的函数和积分区间，并分别调用两个求积公式计算该积分。在本程序中我们在题目要求的基础上增加了复合公式的网格密度和 Romberg 求积算法的精度，进行多次试验，最终会给出题目要求的两个积分结果，并且绘出两个积分公式的误差随着网格密度和外推次数增加的变化图。

2. gaussLegendre5_comp.m：使用 Gauss-Legendre 五点公式进行积分的程序。该程序接受被积函数句柄，积分上限和下限，以及复合公式划分的区间数，最终返回积分值。

3. romberg.m：使用 Romberg 求积算法进行积分的程序。该程序接受被积函数句柄，积分上限和下限，指定最高外推次数和精度，最终返回积分值。

四、 计算结果及其分析

图 1是我们根据程序计算结果得到的数据。此处为了更清楚的显示误差和方便显示有效数字位数的变化，我们记录了 $lg(E)$ 来近似有效位数（位数差别不会超过 1）。

Gauss-Legendre-5			Romberg		
n-section	I(f)	Number of significant digits	n-richardson	I(f)	Number of significant digits
4	-0.238732340343646	7	4	-0.239276089268083	3
5	-0.238732413461231	9	5	-0.238909004534682	4
6	-0.238732415956953	9	6	-0.238734543948288	6
7	-0.238732415253616	9	7	-0.238732414267790	9
8	-0.238732414880270	10	8	-0.238732414621624	11
9	-0.238732414733968	10	9	-0.238732414637833	14
10	-0.238732414677675	10	10	-0.238732414637843	16
11	-0.238732414655229	11	11	-0.238732414637843	16
12	-0.238732414645832	11	12	-0.238732414637843	16
13	-0.238732414641695	11	13	-0.238732414637843	16
14	-0.238732414639784	12	14	-0.238732414637843	16
15	-0.238732414638861	12	15	-0.238732414637843	16
16	-0.238732414638396	12	16	-0.238732414637843	16
17	-0.238732414638154	13	17	-0.238732414637843	16
18	-0.238732414638023	13	18	-0.238732414637843	16
19	-0.238732414637950	13	19	-0.238732414637843	16
20	-0.238732414637908	13	20	-0.238732414637843	16
21	-0.238732414637884	13	21	-0.238732414637843	16
22	-0.238732414637869	14	22	-0.238732414637843	16
23	-0.238732414637860	14	23	-0.238732414637843	16

图 1: 计算结果

(a) **使用 Gauss-Legendre 五点公式的求解结果**

如表中 Gauss-Legendre-5 部分所示。从数据中我们可以看到随着网格的加密，有效数字的位数在提高。其中第一行 (n-section=4) 即为题目要求的求积结果，可以看到有效数字已经达到了 6 位。

(b) **使用 Romberg 求积算法的求解结果**

如表中 Romberg 部分所示。从数据中我们可以看到随着外推次数（也就是外推矩阵的形状）的增加，有效数字的位数在提高，并且提高的速度比 Gauss 求积公式更快，但是最初的计算结果并不好，这也说明梯形公式的精度不如 Gauss 求积公式，起主要作用的应该是外推部分。其中第四行（n-richardson=7）时已经达到题目要求（$\varepsilon < 1^{-7}$），并且有效数字已经达到 9 位。

为了清楚的比较两种求积公式的误差，我们给出了误差随着迭代次数的变化曲线，如图2所示。由图可知，随着网格的加密，Gauss-Legendre 在五点的情况下，即使不把网格取得很密，也可以获得较为精确的解，但是收敛的速度相比 Romberg 要慢一些，但是 Romberg 算法在初期因为梯形公式本身精度的问题，在外推次数较少的情况下无法取得较为精确的解。但是，只要合理设置外推次数，可以更快地获得更加精确的解。

五、 结论

在本次上机实验中，我们分别采用 Gauss-Legendre 五点积分公式和 Romberg 求积算法分别对指定函数和区间进行了积分，从中我们可以看出两种公式各有利弊，Gauss-Legendre

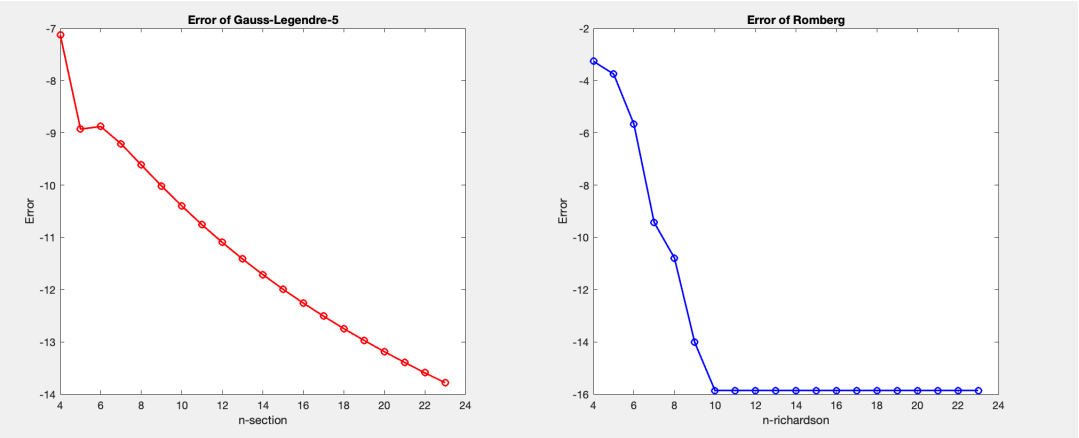

图 2: 误差变化示意图

公式不需要过密的网格就可以得到精确解，而 Romberg 加速效率更到，但是在参数设置得当的情况下，两种公式均可以取得很好的结果。