数值积分——复化梯形求积公式

湘潭大学, 数学与计算科学学院

复化求积法的思想：将区间 $[a, b]$ 进行 n 等分，步长 $h = \frac{b-a}{n}$，等分点 $x_k = a + kh, k = 0, 1, 2, \cdots, n$, 先在每个子区间 $[x_k, x_{k+1}]$ 上采用低阶的数值求积公式求得近似积分值 I_k, 再将它们累加并以和 $\sum_{k=0}^{n-1} I_k$ 作为积分 $\mathrm{I} = \int_{\mathrm{a}}^{\mathrm{b}} \mathrm{f(x)dx}$ 的近似值.

一、复化梯形求积公式

把积分区间 $[a, b]$ 进行 n 等分，步长为 $h = \frac{b-a}{n}$，节点 $x_k = a+kh, (k = 0, 1, 2, \cdots, n)$

$$I(f) = \int_a^b f(x)\mathrm{d}x = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x)\mathrm{d}x$$

在 $[x_i, x_{i+1}]$ 上使用梯形公式

$$\int_{x_i}^{x_{i+1}} f(x)\mathrm{d}x = \frac{h}{2}[f(x_i) + f(x_{i+1})] - f''(\xi_i)\frac{h^3}{12} \approx \frac{h}{2}[f(x_i) + f(x_{i+1})]$$

有

$$I(f) = \sum_{i=0}^{n-1} \left\{ \frac{h}{2}[f(x_i) + f(x_{i+1})] - f''(\xi_i)\frac{h^3}{12} \right\} \approx h \left[\frac{1}{2}f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}f(b) \right]$$

得复化梯形公式

$$T(h) = T_n(f) = h \left[\frac{1}{2}f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}f(b) \right] \approx \int_a^b f(x)\mathrm{d}x$$

二、算法

♡ 复化梯形求积公式：`T = 复化梯形求积_v1(a,b,n,f)`

1. 输入
   - $[a, b]$
   - n : 将 [a,b] n 等分
   - f : 已经定义好的函数，支持向量运算
2. 实现步骤
   - 计算出 $[a, b]$ n 等分后得到的 $n + 1$ 个节点，构成向量 x0
   - y0 = f(x0);
   - 代入
   $$\int_a^b f(x)\mathrm{d}x \approx h\left[\frac{1}{2}f(a) + \sum_{i=1}^{n-1} f(x_i) + \frac{1}{2}f(b)\right] = T$$
3. 输出
   - T：通过复化梯形求积公式得到的积分近似值

三、北太天元源程序

```
function T = 复化梯形求积_v1(a,b,n,f)
% [a,b]
% n : 小区间的个数
% f: 定义好的函数
   h = (b-a)/n;
   k = 0:1:n;
   xi = a + k * h;
   yi = f(xi);
   sumy = sum(yi(2:1:n));
   T = (yi(1)/2 + sumy + yi(n+1)/2)*h;
end
```

将上述代码保存为 `复化梯形求积_v1.m` 文件。

四、数值算例

例 1 用数值积分法近似计算

$$\pi = 4\int_0^1 \frac{1}{1+x^2}\mathrm{d}x$$

编写复化梯形公式的实现程序，分别取剖分段数 $n = 10, 20, 40, 80, 160$，计算积分值与 π 的误差并作图；

例子

```
% 复化梯形求积例子
clc;clear all;format long;
f = @(x) 4./(1+x.^2);

N = [10 20 40 80 160];
delta = zeros(1,5);
k = 1;
for n = N
    T = 复化梯形求积_v1(0,1,n,f);
    delta(k) = abs(pi-T);
    k++;
end
     plot(N,delta,'b');
disp(delta);
```

将上述代码保存为 `复化梯形例子.m`

运行后得到 积分值与 π 的误差

```
  列 1 -- 3
   0.001666664682634    0.000416666635664    0.000104166666182
  列 3 -- 5
   0.000026041666659    0.000006510416666
```

通过观察图像可以发现，随着对积分区间 $[0,1]$ 划分的越来越细致，每个子区间步长减少，最终计算得到的误差越来越小。

观察上面三个图像，可以发现，复化求积法下对积分的计算是非常稳定的，避开了 Newton-Cotes 公式下随阶数升高带来的不稳定性。

也可以发现复化梯形在精度和收敛速度方面都大幅度弱于复化辛普森公式。

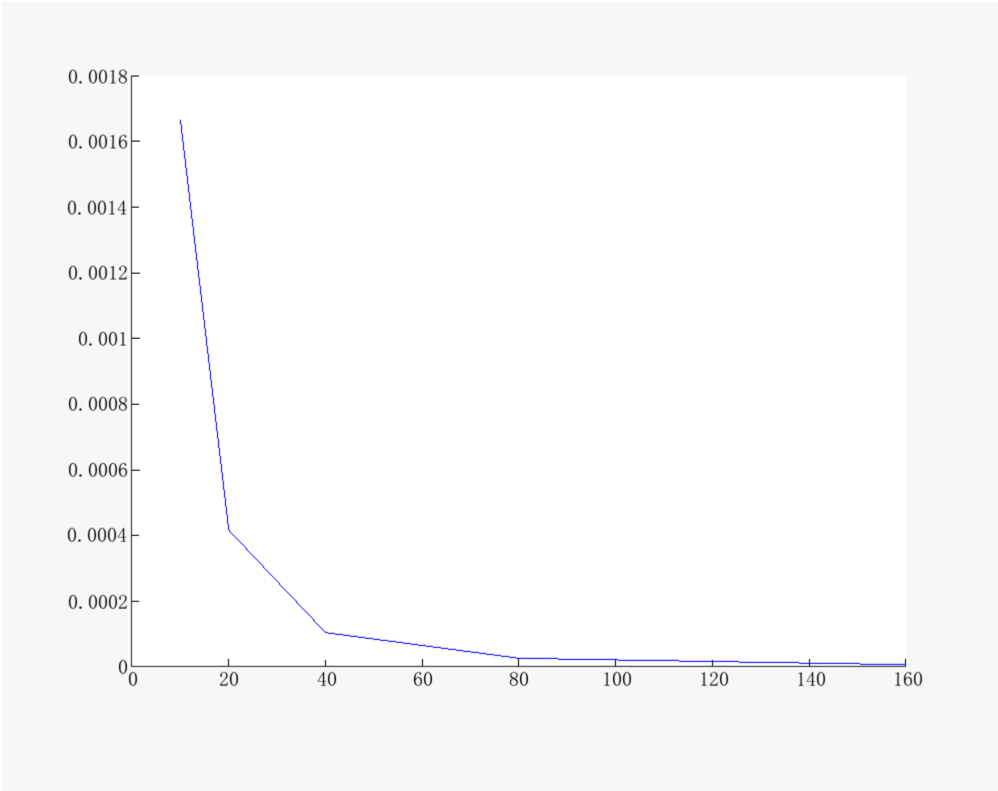

图 **1**　复化梯形求积公式下积分值与 π 的误差随 **n** 的变化

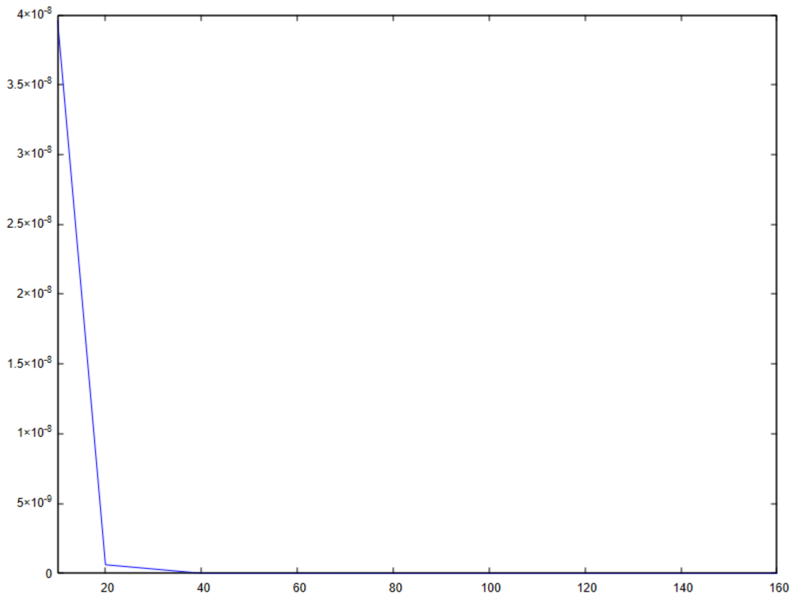

图 **2**　复化 **Simpson** 求积公式下积分值与 π 的误差随 **n** 的变化

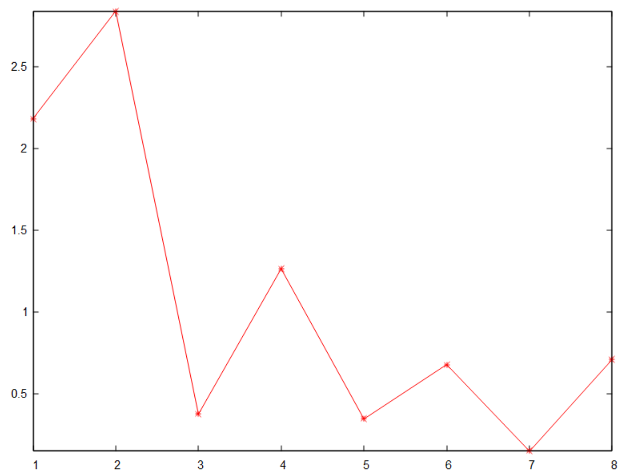

图 **3　Newton—Cotes** 公式下的一个图像参照