数值分析 Numerical Analysis

Lecture 13：插值型数值积分

Numerical Integration

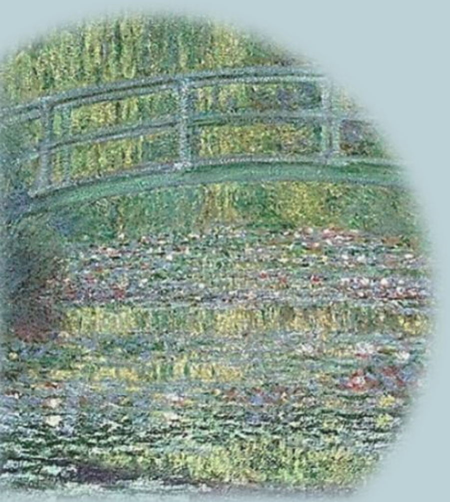
骞微著

苏州大学，计算机科学与技术学院

邮箱：wzqian@suda.edu.cn

办公室：理工楼543

目 录



4.1 插值型求积公式的代数精度

记 $R_n[f]=I-I_n$ 为采用插值型求积公式进行积分近似的截断误差，则由多项式插值公式的误差估计式（5-1）得

$$R_n[f]=I-I_n=\int_a^b f(x)dx-\int_a^b L_n(x)dx=\int_a^b \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_{n+1}(x)dx \qquad (5-14)$$

因此，当 $f(x)$ 为次数不超过 n 次的多项式时，插值型求积公式精确成立，由此引出“代数精度”的概念。

定义 5.1（代数精度） 若近似于定积分的数值积分公式当且仅当 $f(x)$ 为次数不高于 n 次的代数多项式时恒等于定积分 I，则称该数值积分公式的**代数精度**为 n（次）。

由定义知，插值型求积公式的代数精度至少为 n。

4.1 插值型求积公式的代数精度

定理 5.1 设 I_n 为由 N-C 公式（5-10）计算生成，则当 n 为奇数时，I_n 的代数精度为 n；当 n 为偶数时，I_n 的代数精度为 $n+1$，且当 $f^{(n+1)}(x)$ 在区间 $[a, b]$ 上连续时，我们有如下误差估计式

$$R_n(f)=\begin{cases}\dfrac{f^{(n+1)}(\eta)}{(n+1)!}\displaystyle\int_a^b \omega_{n+1}(x)dx & (n\text{为奇数}) \\[2ex] \dfrac{f^{(n+2)}(\eta)}{(n+2)!}\displaystyle\int_a^b (x-\dfrac{a+b}{2})\omega_{n+1}(x)dx & (n\text{为偶数})\end{cases} \qquad (5—15)$$

这里 $\eta\in(a,b)$

4.1 插值型求积公式的代数精度

特别地，

$$I-T=R_1[f]=\frac{f''(\eta)}{2!}\int_a^b(x-a)(x-b)dx$$

$$=-\frac{f''(\eta)}{12}(b-a)^3 \qquad (5-16a)$$

$$I-S=R_2[f]=\frac{f^{(4)}(\eta)}{4!}\int_a^b(x-a)(x-\frac{a+b}{2})^2(x-b)dx$$

$$=-\frac{f^{(4)}(\eta)}{2880}(b-a)^5 \qquad (5-16b)$$

$$I-C=R_4[f]=-\frac{f^{(6)}(\eta)}{1935360}(b-a)^7 \qquad (5-16c)$$

4.1 插值型求积公式的代数精度

从而可得

$$I - T_n = -\frac{b-a}{12} f''(\eta) h^2 \qquad (5-17a)$$

$$I - S_n = -\frac{b-a}{2880} f^{(4)}(\eta) h^4 \qquad (5-17b)$$

$$I - C_n = -\frac{b-a}{1935360} f^{(6)}(\eta) h^6 \qquad (5-17c)$$

4.2步长减半算法

为便于估计误差，实际计算时常常采用步长逐次减半的算法，下面介绍其思想。

由 $$I-T_n=-\frac{b-a}{12}f''(\eta)h^2$$ 得 $$\frac{I-T_{2n}}{I-T_n}\approx\frac{1}{2^2}$$

$$I-T_{2n}=-\frac{b-a}{12}f''(\tilde{\eta})\left(\frac{h}{2}\right)^2$$

所以 $$I-T_{2n}\approx\frac{1}{4-1}(T_{2n}-T_n) \qquad (5-18a)$$

类似可得 $$I-S_{2n}\approx\frac{1}{4^2-1}(S_{2n}-S_n) \qquad (5-18b)$$

$$I-C_{2n}\approx\frac{1}{4^3-1}(C_{2n}-C_n) \qquad (5-18c)$$

4.2 步长减半算法

因此，可先用 $T_1=\frac{b-a}{2}[f(a)+f(b)]$ 计算出T_1，并把步长减半算出T_2，若 $\frac{1}{3}|T_2-T_1|<\varepsilon$ (ε 为精度要求)，则T_2即为所求的近似值，否则再把步长减半，算出T_4，根据式 (5-18a) 进行事后误差估计$\frac{1}{3}|T_4-T_2|,\cdots\cdots$，如此递推计算，直到某个 n 满足 $\frac{1}{3}|T_{2n}-T_n|<\varepsilon$ 为止，取 T_{2n} 为所求的近似值，这就是**梯形公式的步长逐次减半算法**。

类似地，可对 Simpson 公式和 Cotes 公式分别利用（5-18b）和（5-18c）进行事后误差估计，建立步长逐次减半的算法。

4.2 步长减半算法举例

例 5.3 试用梯形公式的步长逐次减半算法计算定积分 $\int_0^1 \sin x\, e^{-x^2}\mathrm{d}x$，使误差小于 $\varepsilon=10^{-6}$。

解 对于 $f(x)=\sin x\, e^{-x^2}$，我们有 $f(0)=0, f(1)=e^{-1}\sin 1$，

$$f\left(\frac{1}{2}\right)=\left(\sin\frac{1}{2}\right)e^{-\frac{1}{4}}$$

所以
$$T_{2^0}=\frac{1-0}{2}[f(0)+f(1)]\approx\frac{1}{2}(0+0.309\ 559\ 9)=0.154\ 779\ 9$$

$$T_{2^1}=\frac{1}{2}T_{2^0}+\frac{1}{2}f\left(\frac{1}{2}\right)\approx\frac{1}{2}0.154\ 779\ 9+\frac{1}{2}0.373\ 377\ 0\approx 0.264\ 078\ 5$$

表 5-4 梯形公式的步长减半算法的计算结果

k	T_{2^k}
0	0.154 779 9
1	0.264 078 5
2	0.287 239 4
3	0.292 845 0
4	0.294 235 6
5	0.294 582 6
6	0.294 669 3
7	0.294 691 0
8	0.294 696 4
9	0.294 697 8
10	0.294 698 0

由于 $|T_{2^{10}}-T_{2^9}|<\varepsilon$，故取 $\int_0^1 \sin x\, e^{-x^2}\mathrm{d}x \approx 0.294\ 698\ 0$

4.3 龙贝格积分算法

Motivation： 基于梯形公式的步长减半算法虽然简单，但收敛速度较慢.

龙贝格Romberg积分算法：基于步长减半算法的加速方法.

由式（5-18a）得，

$$I = \left[T_{2n} + \frac{1}{3}(T_{2n} - T_n)\right] \approx 0$$

在T_n，T_{2n}已知的条件下，构造以下近似值来增加精度：

$$\bar{T} = \frac{1}{3}(4T_{2n} - T_n)$$

进一步观察可得$\bar{T} = S_n$，即

$$S_n = \frac{1}{3}(4T_{2n} - T_n)$$

4.3 龙贝格积分算法

这说明收敛较快的Simpson步长减半序列$\{S_{2^k}\}$可以由梯形公式序列$\{T_{2^k}\}$方便地得到。

由式（5-18b）得，

$$I-\frac{1}{4^2-1}(4^2S_{2n}-S_n)\approx 0$$

验证得：

$$C_n=\frac{1}{4^2-1}(4^2S_{2n}-S_n)$$

从式（5-18c)得

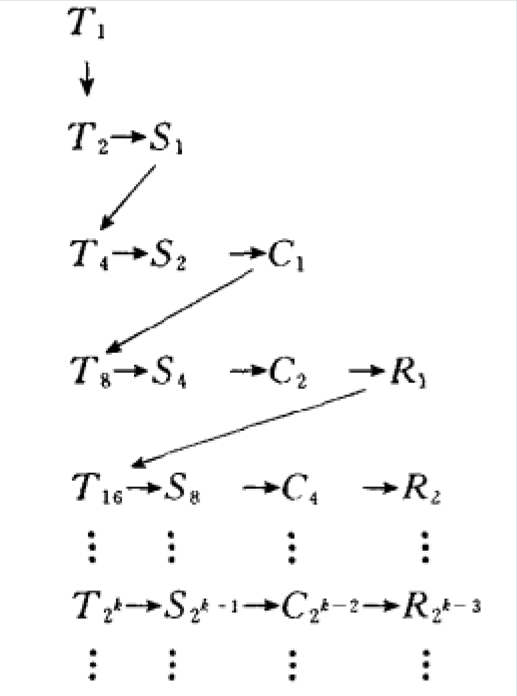
$$I-\frac{1}{4^3-1}(4^3C_{2n}-C_n)\approx 0$$

4.3 龙贝格积分算法

由此建立计算公式

$$R_n = \frac{1}{4^3 - 1}(4^3 C_{2n} - C_n)$$

这就是龙贝格积分公式。相应的龙贝格积分算法：



上述每个公式只需两次乘除运算。因此，龙贝格积分算法计算到$R_{2^{k-3}}$时所需的乘除工作量为

$$(3k+2)+2[k+(k-1)+(k-2)] = 9k-4$$

所涉及的函数值为2^k+1个。

4.3 龙贝格积分算法

算法 5.1（龙贝格积分法）

(1)输入积分限 a,b，精度要求 ϵ 和最大区间分隔数 N；

(2)令 $h=b-a$；

(3)计算 $y_1=\frac{h}{2}[f(a)+f(b)]$；

(4)令 $m=1,n=1$；

(5)计算 $P=\frac{1}{2}[y_1+h\sum_{i=0}^{n-1}f(a+(i+\frac{1}{2})h)]$；

（注：$P=T_{2n}$）

(6)令 $S=1$；

(7)对 $k=1,2,\cdots,m$ 计算

$S=4S,Q=(SP-y_k)/(S-1)$；$y_k=P,P=Q$；

$$\left(\begin{array}{l}\text{注：}K=1\text{ 时，}y_1=T_{2n},P=Q=S_n(m\geqslant 1)\\ \quad K=2\text{ 时，}y_2=S_n,P=Q=C_{n/2}(m\geqslant 2)\\ \quad K=3\text{ 时，}y_3=C_{\frac{n}{2}},P=Q=R_{n/4}(m\geqslant 3)\end{array}\right)$$

(8)若 $m=3$，转入(12)；否则转入(9)；

(9) 若 $|Q-y_m|<\epsilon$，转入(19)；否则转入(10)；

(10)$m=m+1$；

(11)$y_m=Q,n=2n,h=\frac{h}{2}$，转入(5)；

(12)$n=2n,h=\frac{h}{2}$；

(13)计算 $P=\frac{1}{2}[y_1+h\sum_{i=0}^{n-1}f(a+(i+\frac{1}{2})h]$；

(14)令 $S=1$；

(15)对 $k=1,2,\cdots,m$ 计算

$S=4S,Q=(SP-y_k)/(S-1),y_k=P,P=Q$；

(16)若 $|Q-P|<\epsilon$ 则转入(19)；否则转入(17)；

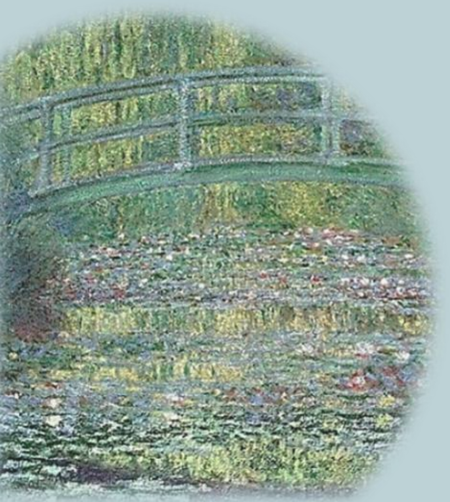
(17)若 $n\leqslant N$，则转入(12)；否则转入(18)；

(18)输出失败信息，终止计算；

(19)$T=Q$；

(20)打印 T。

注　为编程方便，在计算到 R_2 之前，我们分别采用较为粗糙的判则 $|T_2-S_1|<\epsilon$，$|S_2-C_1|<\epsilon$，$|C_2-R_1|<\epsilon$ 作为计算终止原则，当计算到 R_2 之后，则采用 $|R_{2n}-R_n|<\epsilon$ 作为计算终止原则。



4.3 龙贝格积分算法举例

例 5.4 试用龙贝格积分法求解定积分 $\int_0^1 \sin x \, e^{-x^2} dx$，使误差小于 $\epsilon = 1-10^{-6}$。

解 用龙贝格积分法求解，得到如表 5-5 所示的结果。

表 5-5 龙贝格积分法的计算结果

k	T_{2^k}	$S_{2^{k-1}}$	$C_{2^{k-2}}$	$R_{2^{k-3}}$
0	0.154 779 9			
1	0.264 078 5	0.300 511 3		
2	0.287 239 4	0.294 959 7	0.294 589 5	
3	0.292 845 0	0.294 713 5	0.294 697 1	0.294 698 8
4	0.294 235 6	0.294 699 1	0.29 4 698 2	0.294 698 2

由于 $|R_2 - R_1| < 10^{-6}$，故取 $\int_0^1 \sin x \, e^{-x^2} dx \approx 0.294\ 698\ 2$。

作业13

7. 给定积分 $I=\int_a^b f(x)\mathrm{d}x, f(x)=1+2x+4x^2$

验证关系式 $\begin{cases} I-T=-\dfrac{f''(\eta)}{12}(b-a)^3 \\ I=S \end{cases} \quad \eta\in(a,b)$

8. 分别用梯形公式的逐次减半算法和龙贝格积分法求解以下定积分，使误差不超过 10^{-4}。

(1) $\int_0^1 \dfrac{2}{1+x^3}\mathrm{d}x$

(2) $\int_0^{0.8} \mathrm{e}^{-x^2}\mathrm{d}x$

作业以word格式将**推导**，**代码**，（可视化）**运行结果**等（控制文件大小，不要粘贴代码图片，文件名：学号_姓名_作业13）发到助教（小沈）邮箱：20245227049@stu.suda.edu.cn。


Thanks