



清华大学 深圳国际研究生院
Tsinghua Shenzhen International Graduate School

工程硕士数学

第七章 数值积分与数值微分

胡振中 副教授

海洋大楼 702-3, 15999640239 (微信同号)

邮箱: huzhenzhong@tsinghua.edu.cn

个人网站: <http://www.huzhenzhong.net>

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

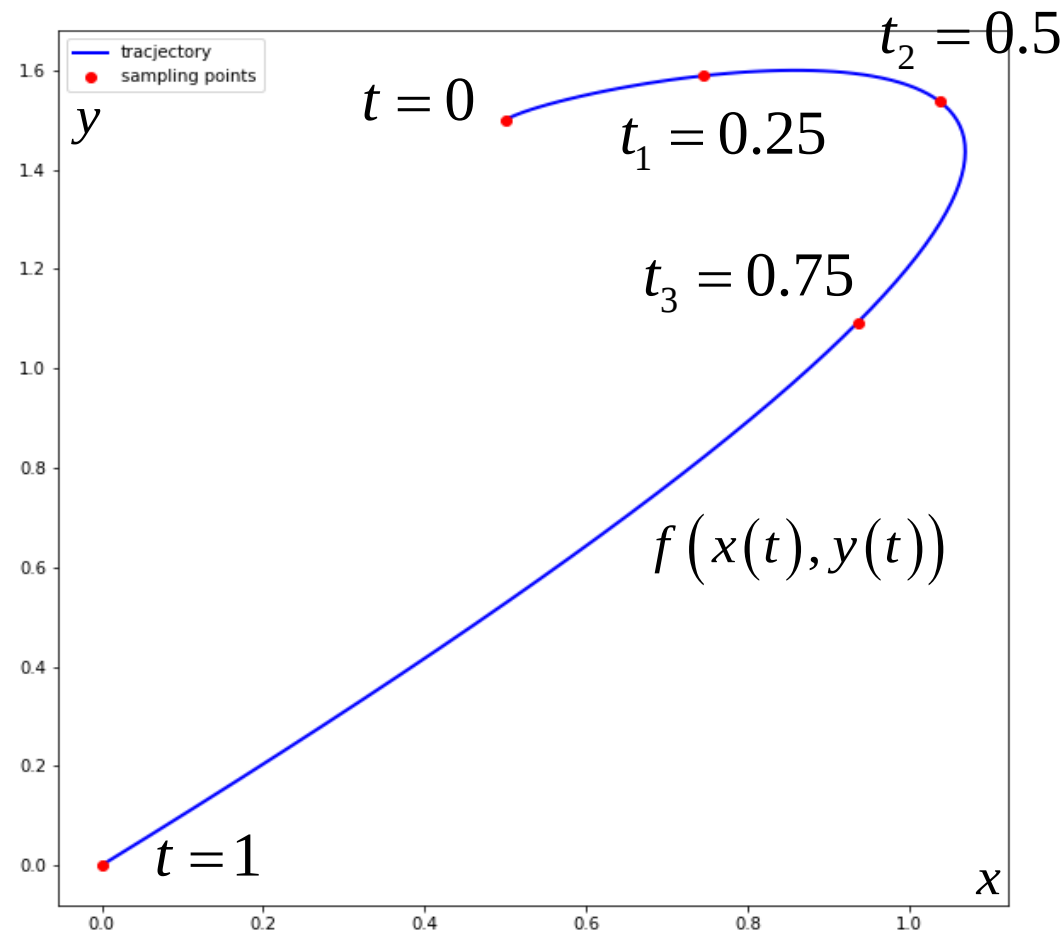
编程实践

数值积分问题

计算运动轨迹在某段时间
间隔内的弧长

$$\int_{t_i}^{t_{i+1}} \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2} dt = \int_{t_i}^{t_{i+1}} \sqrt{v_x^2(t) + v_y^2(t)} dt$$

本章即主要研究这一类**一元定积分**问题的数值计算方法



数控加工、机器人导航、离子运动

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

记一元定积分： $I(x) = \int_a^b f(x)dx$ ，如果可以写出 $f(x)$ 的原函数 $F(x)$ ，则

$$I(x) = \int_a^b f(x)dx = F(x) \Big|_a^b = F(b) - F(a) \quad \textbf{Newton-Leibniz公式}$$

我们只要通过符号运算（也可以写成计算机程序）写出原函数即可计算积分，为什么还需要数值积分？

有些函数无法用初等函数表达原函数，如 $\int e^{-x^2} dx$ 、 $\int \frac{\sin x}{x} dx$

在工程上我们往往连 $f(x)$ 的表达式都没有，只能得到一系列离散点： $x_0, x_1, \dots, x_n \rightarrow f(x_0), f(x_1), \dots, f(x_n)$

只能数值计算！

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

要计算 $I(x) = \int_a^b f(x)dx$, 既然 $f(x)$ 或 $F(x)$ 不易获得, 能否找到一个 $f(x)$ 近似 $\varphi(x)$

使得: $I(x) = \int_a^b \varphi(x)dx$ 比较容易计算? 近似?



最简单的方法, 直接在 $[a, b]$ 区间上采用线性插值, 插值节点: $x_0 = a, x_1 = b$

回忆一下一次Lagrange插值多项式:

$$L_1(x) = \frac{x-b}{a-b} f(a) + \frac{x-a}{b-a} f(b) \quad \Rightarrow \quad I(f) \approx I_1(f) = \int_a^b L_1(x) dx$$

插值型求积公式—梯形公式



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\int_a^b L_1(x) dx = \frac{f(a)}{a-b} \int_a^b (x-b) dx + \frac{f(b)}{b-a} \int_a^b (x-a) dx$$

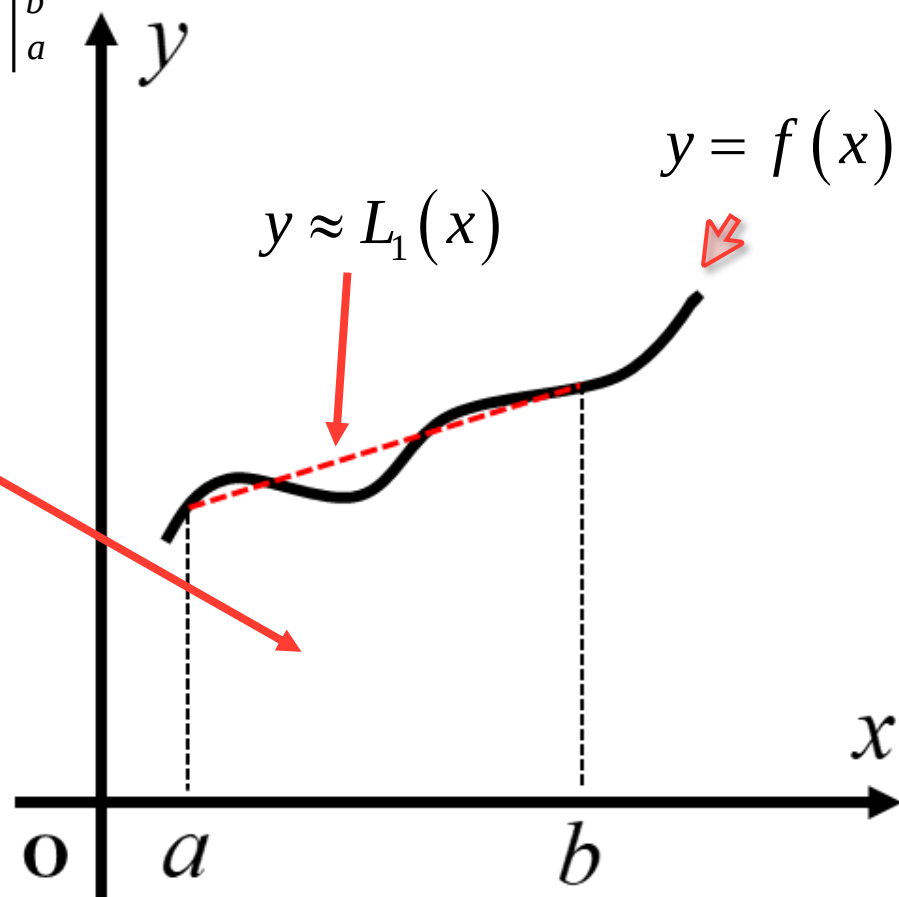
$$= \frac{f(a)}{a-b} \frac{1}{2} (x-b)^2 \Big|_a^b + \frac{f(b)}{b-a} \frac{1}{2} (x-a)^2 \Big|_a^b$$

$$= \frac{b-a}{2} [f(a) + f(b)]$$

相当于用这个梯形的面积近似复杂函数所包含的面积



这种近似的性能如何？



插值型求积公式—梯形公式误差



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

介值定理

设 $f(x)$ 在 $[a, b]$ 上连续, 且在端点取不同的函数值 $f(a) = A, f(b) = B$ 对于 A 和 B 之间的任意数 C , 在开区间 (a, b) 内必定存在 ξ 使得: $f(\xi) = C, a < \xi < b$

定积分中值定理

设 $f(x)$ 在 $[a, b]$ 上连续, 则在闭区间 $[a, b]$ 内必定存在 ξ 使得: $\int_a^b f(x) dx = f(\xi)(b - a)$

定理8.1.1 (教材编号)

设 $f(x), g(x)$ 在 $[a, b]$ 上连续, 且 $g(x)$ 在区间内不变号, 则存在 $\xi \in [a, b]$, 使得

$$\int_a^b f(x) g(x) dx = f(\xi) \int_a^b g(x) dx$$

积分第一中值定理

可由积分中值定理推广而来

插值型求积公式—梯形公式误差



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.1.2 (教材编号)

设 $f(x) \in C^2[a, b]$, 那么梯形求积公式的误差可以表示成下面形式

$$I - I_1 = \int_a^b f(x) dx - \frac{b-a}{2} [f(a) + f(b)] = -\frac{h^3}{12} f''(\xi) \quad \xi \in [a, b], h = b - a$$

$$E_1 = \int_a^b [f(x) - L_1(x)] dx = \int_a^b \underbrace{(x-a)(x-b)}_{\text{非正, 保号}} \frac{f(x) - L_1(x)}{\underbrace{(x-a)(x-b)}} dx$$

所以 $E_1 = \frac{f(z) - L_1(z)}{(z-a)(z-b)} \int_a^b (x-a)(x-b) dx, z \in [a, b]$

$(x-a) \frac{1}{2} (x-b)^2 \Big|_a^b - \int_a^b \frac{1}{2} (x-b)^2 dx = \frac{h^3}{6}$ 得证!

硬算或部分积分

$$R_1(z) = \frac{f''(\xi)}{2} (z-a)(z-b)$$

ξ 并不好找

$$|I(f) - I_1(f)| \leq \frac{1}{12} \|f''\| h^3$$

插值型求积公式—Simpson公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

思路：选取节点 $x_0 = a, x_1 = \frac{a+b}{2}, x_2 = b$ ，构造插值函数 L_2

$$L_2 = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} f(x_0) + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} f(x_1) + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} f(x_2)$$

注意： $h = b - a$ ，等间距 $x_1 - x_0 = x_2 - x_1 = \frac{h}{2}$

$$L_2 = \frac{2}{h^2} f(x_0) \underbrace{(x-x_1)(x-x_2)}_{\frac{h^3}{12}} - \frac{4}{h^2} f(x_1) \underbrace{(x-x_0)(x-x_2)}_{-\frac{h^3}{6}} + \frac{2}{h^2} f(x_2) \underbrace{(x-x_0)(x-x_1)}_{(x-x_0)\frac{1}{2}(x-x_1)^2} \Big|_a^b - \int_a^b \frac{1}{2}(x-x_1)^2 dx = \frac{h^3}{12}$$

$$I \approx I_2 = \int_a^b L_2(x) dx = \frac{h}{6} \left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right]$$

Simpson求积公式

插值型求积公式—Simpson公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.1.3 (教材编号)

设 $f(x) \in C^4[a, b]$, 那么Simpson求积公式的误差为

$$E_2 = I - I_2 = -\frac{h^5}{2880} f^{(4)}(\xi), \xi \in [a, b]$$

证明

构造 $p_3(x) = L_2(x) + \frac{4}{h^2} [L'_2(x_1) - f'(x_1)] \omega_3(x)$, $p_3(x)$ 满足

$$(1) \quad p_3(x_k) = f(x_k), k = 0, 1, 2$$

$$(2) \quad p'_3(x_1) = f'(x_1)$$

$$(x - x_0)(x - x_1)(x - x_2)$$

$$\Rightarrow \int_a^b (x - x_0)(x - x_1)(x - x_2) dx$$



插值型求积公式—Simpson公式



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$u(x) = (x - x_0)(x - x_2), v'(x) = (x - x_1), v(x) = \frac{1}{2}(x - x_1)^2$$

$$\int_a^b (x - x_0)(x - x_1)(x - x_2) dx = uv \Big|_a^b - \int_a^b u'v dx = 0 \quad u' = x - x_0 + x - x_2 = 2x - 2x_1$$

$$E_2 = \int_a^b [f(x) - L_2(x)] dx = \int_a^b [f(x) - \underline{p_3(x)}] dx$$

三次插值多项式, x_1 处为二重节点

$$E_2 = \int_a^b \underbrace{(x - x_0)(x - x_1)^2(x - x_2)}_{\text{非正, 保号}} \underbrace{\frac{f(x) - p_3(x)}{(x - x_0)(x - x_1)^2(x - x_2)}}_{\text{连续}} dx$$

非正, 保号

连续

插值型求积公式—Simpson公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$E_2 = \frac{f(z) - p_3(z)}{(z - x_0)(z - x_1)^2(z - x_2)} \int_a^b (x - x_0)(x - x_1)^2(x - x_2) dx = -\frac{h^5}{120}$$

带一个重节点的插值余项：

$$R(x) = \frac{f^{(4)}(\xi)}{4!} (z - x_0)(z - x_1)^2(z - x_2) \Rightarrow E_2 = -\frac{h^5}{2880} f^{(4)}(\xi), \xi \in [a, b]$$

同样道理， ξ 并不好找，因而一般可以估计误差上限

$$|E_2| = |I - I_2| \leq \frac{h^5}{2880} \|f^{(4)}\|_{\infty}$$

插值型求积公式——一般推广公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

事实上，刚才的两种插值公式都是由Lagrange插值而来，我们可以直接推广至任意 n 次Lagrange插值的近似

在积分区间 $[a, b]$ 上给定 $n+1$ 个节点 $a = x_0 < x_1 < \dots < x_n = b$ 及相应的函数值，则

$$f(x) = L_n(x) + R_n(x) \quad L_n(x) = \sum_{k=0}^n f(x_k) l_k(x) \quad l_k(x) = \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x - x_j}{x_k - x_j}$$

$$\int_a^b f(x) dx = \int_a^b L_n(x) dx + \int_a^b R_n(x) dx = \sum_{k=0}^n f(x_k) \underbrace{\int_a^b l_k(x) dx}_{A_k} + \underbrace{\int_a^b R_n(x) dx}_{E_n(f)}$$

一系列值 $A_k, k = 0, 1, \dots, n$

一个值 $E_n(f)$

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k f(x_k)$$

插值型求积公式！

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

在讨论不同的 A_k 构造之前，我们先讨论余项的积分 $E_n(f)$

$$E_n(f) = \frac{1}{(n+1)!} \int_a^b f^{(n+1)}(\xi) \omega_{n+1}(x) dx$$

定义8.1.1（教材编号）

对于插值型求积公式，如果对所有 $p \in P^m[a, b]$ 都准确成立，即 $E_n(p) = 0$ 而存在一个 $q \in P^{m+1}[a, b]$ ，使得 $E_n(q) \neq 0$ ，则称

求积公式具有 **m 次代数精度**

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

由 $E_n(f)$ 的定义可以知道, 对于任意两个函数 $f, g \in C[a, b]$

$$E_n(\alpha f + \beta g) = \alpha E_n(f) + \beta E_n(g)$$

所以, $p \in P^m[a, b], E_n(p) = 0$ 等价于 $E_n(x^k) = 0, k = 0, 1, \dots, m$

$q \in P^{m+1}[a, b], E_n(q) \neq 0$ 等价于 $E_n(x^{m+1}) \neq 0$

定义8.1.2 (教材编号)

如果 $E_n(x^k) = 0, k = 0, 1, \dots, m$, 则称求积公式**至少**具有 m 次代数精度; 再有 $E_n(x^{m+1}) \neq 0$, 则称求积公式具有 m 次代数精度

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.1.4 (教材编号)

n 次插值多项式求积, **至少**具有 n 次代数精度

对于梯形公式: $E_1(f) = -\frac{h^3}{12}f''(\xi)$, 若 $f \in P^1[a, b]$ 则 $E_1(f) = 0$,

至少具有1次代数精度

再如果 $f \in P^2[a, b]$, 一般可以满足 $E_1(f) \neq 0$, 则梯形公式

确定具有1次代数精度

那么Simpson求积公式具有多少代数精度



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

接下来，讨论 A_k 的构造

将 $[a, b]$ 用 $n+1$ 个节点等分成 n 份，记 $h = \frac{1}{n}(b - a)$ ， h 为步长

主要研究此时的

$$A_k = \int_a^b l_k(x) dx = \frac{1}{\prod_{\substack{j=0 \\ j \neq k}}^n (x_k - x_j)} \int_a^b \prod_{\substack{j=0 \\ j \neq k}}^n (x - x_j) dx$$

这项有点难



换一下变量 $x = a + th, t \in [0, n]$

$$x_0, x_1, \dots, x_{k-1}, \cancel{x_k}, x_{k+1}, \dots, x_n$$

$$\underbrace{kh \times (k-1)h \times \dots \times h \times (-h) \times (-2h) \times \dots \times (-(n-k)h)}_{k \uparrow \quad h^k k!} \quad \underbrace{(-1)^{n-k} h^{n-k} (n-k)!}_{n-k \uparrow \quad (-1)^{n-k} h^{n-k} (n-k)!}$$

$$\Rightarrow (-1)^{n-k} h^n k! (n-k)!$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\int_a^b \prod_{\substack{j=0 \\ j \neq k}}^n (x - x_j) dx = h^n \int_0^n t(t-1)\dots(t-k+1)(t-k-1)\dots(t-n) h dt$$

某些版本此处教材
P232页有误

$$\text{令: } C_k^{(n)} = \frac{A_k^{(n)}}{b-a} = \frac{(-1)^{n-k}}{k!(n-k)!} \frac{1}{n} \int_0^n \prod_{\substack{j=0 \\ j \neq k}}^n (t-j) dt$$

Cotes求积系数

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k^{(n)} f(x_k) = (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k)$$

Newton-Cotes求积公式

特点

①系数 $C_k^{(n)}$ 不仅和 $f(x)$ 无关, 和 $[a,b]$ 区间也无关, 仅仅和阶数 n 有关

②系数是**对称的**, 即

$$C_k^{(n)} = C_{n-k}^{(n)}, k = 0, 1, \dots, \left\lfloor \frac{n}{2} \right\rfloor \quad \begin{array}{l} n \text{ 为奇数, } n+1 \text{ 为偶数, 一半 } (n+1)/2 \\ n \text{ 为偶数, } n+1 \text{ 为奇数, 中心 } (n/2)+1 \end{array}$$

Cotes系数



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

n	$C_k^{(n)}$				
1	$\frac{1}{2}$				
2	$\frac{1}{6}$	$\frac{4}{6}$			
3	$\frac{1}{8}$	$\frac{3}{8}$			
4	$\frac{7}{90}$	$\frac{16}{45}$	$\frac{2}{15}$		
5	$\frac{19}{288}$	$\frac{25}{96}$	$\frac{25}{144}$		
6	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	
7	$\frac{751}{17280}$	$\frac{3577}{17280}$	$\frac{1323}{17280}$	$\frac{2989}{17280}$	
8	$\frac{989}{28350}$	$\frac{5888}{28350}$	$\frac{-928}{28350}$	$\frac{10496}{28350}$	$\frac{-4540}{28350}$

→ 梯形公式

→ Simpson公式

→ Simpson 3/8公式

→ Cotes公式

$$\int_a^b f(x) dx \approx \sum_{k=0}^n A_k^{(n)} f(x_k) = (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k)$$

系数与若干函数点的加权平均！

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.1.5 (教材编号)

Newton-Cotes的求积公式的误差 $E_n(f)$ 分为两种情况

① n 为**偶数**, 设 $f \in C^{n+2}[a, b]$

$$E_n(f) = C_n h^{n+3} f^{(n+2)}(\xi), \xi \in [a, b] \quad C_n = \frac{1}{(n+2)!} \int_0^n \mu^2 (\mu-1) \dots (\mu-n) d\mu$$

② n 为**奇数**, 设 $f \in C^{n+1}[a, b]$

$$E_n(f) = C_n h^{n+2} f^{(n+1)}(\xi), \xi \in [a, b] \quad C_n = \frac{1}{(n+1)!} \int_0^n \mu (\mu-1) \dots (\mu-n) d\mu$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$n=1$, 即梯形公式

$$E_1(f) = h^3 f^{(2)}(\xi) \frac{1}{2} \int_0^1 \mu(\mu-1) d\mu = h^3 f^{(2)}(\xi) \frac{1}{2} \left(\frac{1}{3} \mu^3 - \frac{1}{2} \mu^2 \right)_0^1 = -\frac{1}{12} h^3 f^{(2)}(\xi)$$

$n=2$, 即Simpson公式

$$E_2(f) = \frac{1}{24} h^5 f^{(4)}(\xi) \int_0^2 \mu^2(\mu-1)(\mu-2) d\mu = -\frac{(b-a)^5}{24 \times 2^5} \frac{4}{15} f^{(4)}(\xi) = -\frac{(b-a)^5}{2880} f^{(4)}(\xi)$$

注意这里的 h

$$E_3(f) = -\frac{3}{80} h^5 f^{(4)}(\xi) \quad 3\text{次代数精度}$$

$$E_4(f) = -\frac{8}{945} h^7 f^{(6)}(\xi) \quad 5\text{次代数精度}$$

$n=2k+1$ 和 $2k(k=1,2,..)$ 具有相同次数的代数精度!

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

刚才的插值区间 $a = x_0 < x_1 < \dots < x_n = b$ 为闭区间

若选择 $a < x_0 < x_1 < \dots < x_n < b$, 则得到开型Newton-Cotes积分公式

考虑等间距 h 节点 $a = x_{-1} < x_0 < x_1 < \dots < x_n < x_{n+1} = b$

$$x_0 = a + h, x_n = b - h, x_k = a + (k+1)h$$

$$\int_a^b f(x) dx = (b-a) \sum_{k=0}^n B_k^{(n)} f(x_k) + \bar{E}_n(f) \quad B_k^{(n)} = \frac{1}{b-a} \int_a^b l_k(x) dx, k = 0, 1, \dots, n$$

开型Newton-Cotes求积公式

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.1.6 (教材编号)

和闭型公式的差别?

开型Newton-Cotes的求积公式的误差 $\bar{E}_n(f)$ 分为两种情况

① n 为**偶数**, 设 $f \in C^{n+2}[a, b]$

$$\bar{E}_n(f) = \bar{C}_n h^{n+3} f^{(n+2)}(\xi), \xi \in [a, b] \quad \bar{C}_n = \frac{1}{(n+2)!} \int_{-1}^{n+1} \mu^2(\mu-1)\dots(\mu-n)d\mu$$

可以定义 **$n=0$** !

② n 为**奇数**, 设 $f \in C^{n+1}[a, b]$

$$\bar{E}_n(f) = \bar{C}_n h^{n+2} f^{(n+1)}(\xi), \xi \in [a, b] \quad \bar{C}_n = \frac{1}{(n+1)!} \int_{-1}^{n+1} \mu(\mu-1)\dots(\mu-n)d\mu$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

中点公式或中矩形公式

$$n=0$$

$$x_{-1} = a, x_0 = \frac{a+b}{2}, x_1 = b$$

$$l_0 = f\left(\frac{a+b}{2}\right)$$

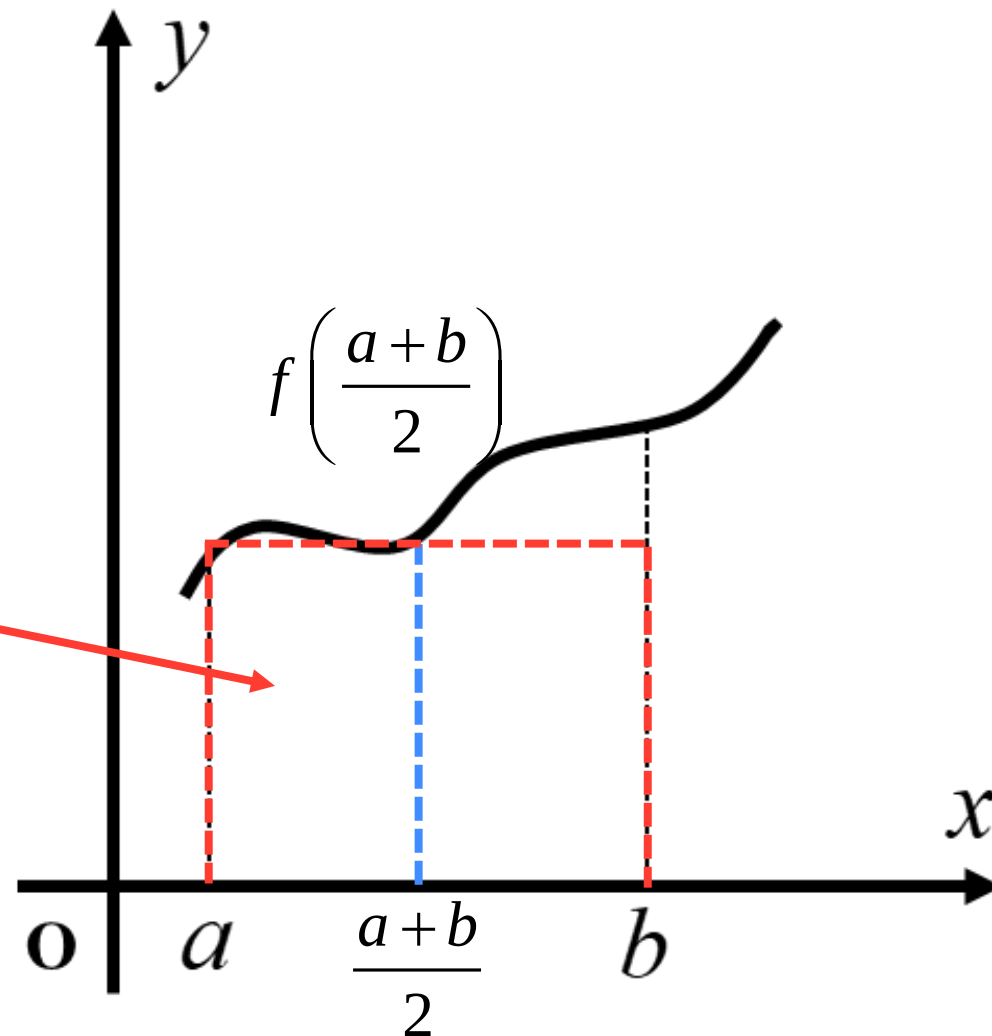


$$\int_a^b f(x) dx \approx (b-a) f\left(\frac{a+b}{2}\right) \quad \text{意义?}$$

用矩形面积近似!

$$\bar{E}_0 = \int_{-1}^1 \mu^2 du \frac{1}{2} h^3 f^{(2)}(\xi) = \frac{1}{3} h^3 f^{(2)}(\xi)$$

也具有1次代数精度!



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

对于闭型求积公式

$$\int_a^b f(x) dx = (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k) + E_n(f)$$

讨论特殊情况 $f(x) \equiv 1$, 显然任何次求积公式都在理论上可以获得精确值

$$\sum_{k=0}^n C_k^{(n)} = 1 \Rightarrow \int_a^b f(x) dx = b-a, E_n(f) = 0$$

假定计算 $f(x_k)$ 时存在误差, 得到的是近似值 $\tilde{f}(x_k)$

$$\sum_{k=0}^n C_k^{(n)} \tilde{f}(x_k) \approx \sum_{k=0}^n C_k^{(n)} f(x_k) \quad \text{记 } \varepsilon = \max_{0 \leq k \leq n} |\tilde{f}(x_k) - f(x_k)|$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\left| \sum_{k=0}^n C_k^{(n)} \tilde{f}(x_k) - \sum_{k=0}^n C_k^{(n)} f(x_k) \right| = \left| \sum_{k=0}^n C_k^{(n)} [\tilde{f}(x_k) - f(x_k)] \right|$$
$$\leq \sum_{k=0}^n |C_k^{(n)}| |\tilde{f}(x_k) - f(x_k)| \leq \varepsilon \sum_{k=0}^n |C_k^{(n)}|$$

$$\text{当 } C_k^{(n)} > 0, k = 0, 1, \dots, n \text{ 时, } \sum_{k=0}^n |C_k^{(n)}| = \sum_{k=0}^n C_k^{(n)} = 1$$

$$\left| \sum_{k=0}^n C_k^{(n)} \tilde{f}(x_k) - \sum_{k=0}^n C_k^{(n)} f(x_k) \right| \leq \varepsilon$$

算法稳定!

但是观察 $n=8$ 时的Cotes系数, 有正有负 $\sum_{k=0}^n |A_k| \rightarrow \infty, n \rightarrow \infty$

算法不稳定!

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

由以上分析可知随着次数的增加，理论的代数精度会提升；**但是**过高阶的求积也会面临数值稳定性问题



如何解决？

分段低次插值



分段低次求积

复合求积方法！

将区间 $[a, b]$ 分为 n 等份， $x_k = a + kh, h = \frac{b-a}{n}, k = 0, 1, \dots, n$,

在每一个子区间 $[x_{k-1}, x_k]$

用梯形公式
$$\int_a^b f(x)dx = \sum_{k=1}^n \int_{x_{k-1}}^{x_k} f(x)dx = \frac{h}{2} \sum_{k=1}^n [f(x_{k-1}) + f(x_k)] + \tilde{E}_n(f)$$

复合求积公式—梯形公式



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\int_a^b f(x) \approx T_n(f) = \frac{h}{2} \sum_{k=1}^n [f(x_{k-1}) + f(x_k)] = \frac{h}{2} \left[\underbrace{f(a) + 2 \sum_{k=1}^n f(x_k) + f(b)} \right]$$

两个端点用一次，所有中间点用两次

对于每一个子区间 $[x_{k-1}, x_k]$ ，可以直接由梯形公式的分析获得误差 $-\frac{h^3}{12} f''(\xi_k)$, $\xi_k \in [x_{k-1}, x_k]$

$$\text{所以 } \tilde{E}_n(f) = -\frac{h^3}{12} \sum_{k=1}^n f''(\xi_k) = -\frac{b-a}{12} h^2 \left[\frac{1}{n} \sum_{k=1}^n f''(\xi_k) \right]$$

$$\min_{1 \leq k \leq n} f''(\xi_k) \leq \frac{1}{n} \sum_{k=1}^n f''(\xi_k) \leq \max_{1 \leq k \leq n} f''(\xi_k)$$



$$\min_{x \in [a, b]} f''(x) \leq \frac{1}{n} \sum_{k=1}^n f''(\xi_k) \leq \max_{x \in [a, b]} f''(x)$$

复合求积公式—梯形公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

根据介值定理, 一定存在一个 $\xi \in [a, b]$, 使得

$$f''(\xi) = \frac{1}{n} \sum_{k=1}^n f''(\xi_k) \Rightarrow \tilde{E}_n(f) = -\frac{b-a}{12} h^2 f''(\xi)$$

当 $h \rightarrow 0, n \rightarrow \infty$ 时, $\tilde{E}_n(f) \rightarrow 0$

$$\lim_{n \rightarrow \infty} T_n(f) = \int_a^b f(x) dx$$

为了更加精确的计算, 还可以在 $[x_{k-1}, x_k]$ 之间增加 $x_{k-\frac{1}{2}}$

使用梯形公式计算

复合求积公式—Simpson公式



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

若将区间 $[a, b]$ 分为 n 等份, $x_k = a + kh, h = \frac{b-a}{n}, k = 0, 1, \dots, n$

在每一个子区间 $[x_{k-1}, x_k]$ 用Simpson公式

$$x_{k-\frac{1}{2}} = \frac{1}{2}(x_{k-1} + x_k)$$

$$\int_a^b f(x)dx = \sum_{k=1}^n \int_{x_{k-1}}^{x_k} f(x)dx = \frac{h}{6} \sum_{k=1}^n \left[f(x_{k-1}) + 4f(x_{k-\frac{1}{2}}) + f(x_k) \right] + \tilde{E}_n(f)$$

同样可以得到:

$$S_n(f) = \frac{h}{6} \left[f(a) + 4f(x_{k-\frac{1}{2}}) + 2 \sum_{k=1}^n f(x_k) + f(b) \right]$$

$$\tilde{E}_n(f) = -\frac{b-a}{2880} h^4 f^{(4)}(\xi)$$

Romberg求积公式—外推



清华大学深圳国际研究生院
Tsinghua Shenzhen International Graduate School
海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

梯形求积公式: $T_n(f) = \frac{h}{2} \sum_{k=1}^n [f(x_{k-1}) + f(x_k)]$

当区间确定, T 是 f 的函数, 也是 n 或者 h 的函数 $h = \frac{b-a}{n}$

因而也可以直接记: $T(h)$

$$I(f) - T(h) = \tilde{E}_n(f) = -\frac{b-a}{12} h^2 f''(\xi)$$

$$f(\xi) = f(\xi_0) + f'(\xi_0)(\xi - \xi_0) + \frac{1}{2} f''(\xi_0)(\xi - \xi_0)^2 + \dots$$

后续会剩下 $(\xi - \xi_0), (\xi - \xi_0)^2, \dots$ 这些项
可以认为与 $h, h^2, \dots$ 同阶, 其中奇数阶
取出一个 h 与 $b-a$ 合并成系数, 加到偶数
阶即可

对这个导两次

可以进一步证明, E 也可以写成 h 的函数

$$E(h) = a_2 h^2 + a_4 h^4 + \dots + O(h^{2m+2}), \forall f \in C^{2m+2}[a, b]$$

E 和 h 的平方同阶

Romberg求积公式—外推



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

如果 h 减少一半

消去该项

$$I(f) - T\left(\frac{h}{2}\right) = E\left(\frac{h}{2}\right) = \cancel{a_2\left(\frac{h}{2}\right)^2} + a_4\left(\frac{h}{2}\right)^4 + \dots + O(h^{2m+2})$$

$$I(f) - T(h) = E(h) = \cancel{a_2h^2} + a_4h^4 + \dots + O(h^{2m+2})$$

$$\begin{aligned} 4\left[I(f) - T\left(\frac{h}{2}\right)\right] - [I(f) - T(h)] &= \cancel{3} \left[I(f) - \frac{1}{3} \left[4T\left(\frac{h}{2}\right) - T(h) \right] \right] \\ &= b_4h^4 + \dots + O(h^{2m+2}) \end{aligned}$$

和 h 的4次方同阶

Romberg求积公式—外推



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$4 \left[I(f) - T\left(\frac{h}{2}\right) \right] - [I(f) - T(h)] = 3 \left[I(f) - \frac{1}{3} \left[4T\left(\frac{h}{2}\right) - T(h) \right] \right] = b_4 h^4 + \dots + O(h^{2m+2})$$

$$S_n(h) = \frac{h}{6} \left[f(a) + 4 \sum_{k=1}^n f(x_{k-\frac{1}{2}}) + 2 \sum_{k=1}^n f(x_k) + f(b) \right]$$

$$T(h) = \frac{h}{2} \left[f(a) + 2 \sum_{k=1}^n f(x_k) + f(b) \right]$$

$$T\left(\frac{h}{2}\right) = \frac{h}{4} \left[f(a) + 2 \sum_{k=1}^n f(x_{k-\frac{1}{2}}) + 2 \sum_{k=1}^n f(x_k) + f(b) \right]$$

Romberg求积公式—外推



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

从复合梯形递推至复合Simpson, 那么再往下呢?



$$h, \frac{h}{2}, \frac{h}{4}, \dots, \frac{h}{2^j}, \dots$$

$$\begin{array}{ccccc} & X_{k-\frac{3}{4}} & & X_{k-\frac{1}{4}} & \\ & \text{-----} & & \text{-----} & \\ X_{k-1} & & X_{k-\frac{1}{2}} & & X_k \end{array}$$

1阶, 2阶, 4阶, ..., 2^j 阶, ...

Richardson外推算法

记 $T(h, k-1)$ 为步长 h 外推了 $k-1$ 步 $I = T(h, k-1) + a_1 h^{2k}, \dots$

$$I = T\left(\frac{h}{2}, k-1\right) + a_1 \left(\frac{h}{2}\right)^{2k}, \dots$$

$$T(h, k) = \frac{4^k T\left(\frac{h}{2}, k-1\right) - T(h, k-1)}{4^k - 1}$$

教材定理8.3.1

消去该项

Romberg求积公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

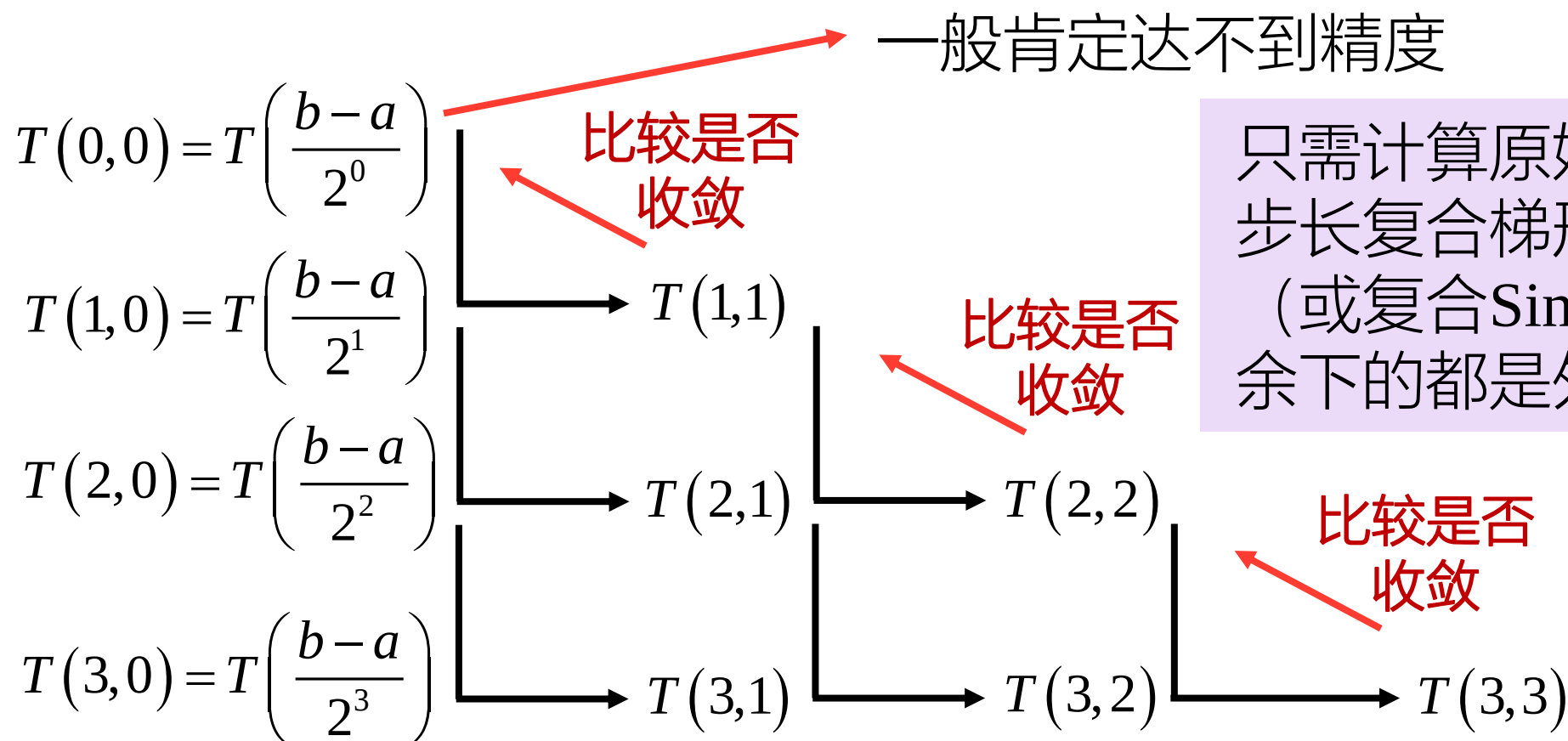
数值微分

Taylor展开构造

插值求导与外推

编程实践

取 $h = \frac{b-a}{2^j}$, 此时可以用 j 代替 h : $T(j, k)$



Romberg求积公式 可证收敛性

$$\lim_{j \rightarrow \infty} T(j, j) = \int_a^b f(x) dx$$

Romberg求积公式一步骤（复合梯形为例）



清华大学深圳国际研究生院
Tsinghua Shenzhen International Graduate School
海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$(1) \quad h = \frac{b-a}{2^0}, T(0,0) = \frac{h}{2} [f(a) + f(b)]$$

程序计算很方便，手算也不难！

$$(2) \quad \text{将区间半分} \quad T(1,0) = T\left(\frac{b-a}{2}\right) = \frac{h}{2} \left[f(a) + 2f\left(\frac{a+b}{2}\right) + f(b) \right]$$

$$T(1,1) = \frac{4T(1,0) - T(0,0)}{3}$$

$$(3) \quad |T(j,j) - T(j-1,j-1)| > \varepsilon \text{ 时, } j+1 \rightarrow j$$

$$T(j,0) = T\left(\frac{b-a}{2^j}\right) = \frac{h}{2} \left[f(a) + 2 \sum_{k=1}^{2^j-1} f(x_k) + f(b) \right]$$

$$T(j,k) = \frac{4^k T(j,k-1) - T(j-1,k-1)}{4^k - 1} \quad k = 1, 2, \dots, j$$

✓ Newton-Cotes公式 (NewtonCotes类)

插值多项式次数

```
public static double Integral(UnaryFunction f, double low, double up, int degree = 1)
{
```

```
    if (degree == 0)
        return (up - low) * f(low);
```

```
    else if (degree == 1)
        return (up - low) * (f(low) + f(up)) / 2;
```

```
    else if (degree == 2)
        return (up - low) * (f(low) +
            4 * f((low + up) / 2) + f(up)) / 6;
```

```
    else if (degree == 3)
```

```
    // .....
```

```
    else
```

```
        throw new Exception("次数超过7时, Newton-Cotes求积公式不稳定, 不宜采用!");
```

```
}
```

n	$C_k^{(n)}$				
1	$\frac{1}{2}$				
2	$\frac{1}{6}$	$\frac{4}{6}$			
3	$\frac{1}{8}$	$\frac{3}{8}$			
4	$\frac{7}{90}$	$\frac{16}{45}$	$\frac{2}{15}$		
5	$\frac{19}{288}$	$\frac{25}{96}$	$\frac{25}{144}$		
6	$\frac{41}{840}$	$\frac{9}{35}$	$\frac{9}{280}$	$\frac{34}{105}$	
7	$\frac{751}{17280}$	$\frac{3577}{17280}$	$\frac{1323}{17280}$	$\frac{2989}{17280}$	
8	$\frac{989}{28350}$	$\frac{5888}{28350}$	$\frac{-928}{28350}$	$\frac{10496}{28350}$	$\frac{-4540}{28350}$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ Newton-Cotes公式 (NewtonCotes类)

梯形公式

```
public static double Trapezoid(UnaryFunction f, double low, double up)
{
    return Integral(f, low, up, 1);
}
```

Simpson公式

```
public static double Simpson(UnaryFunction f, double low, double up)
{
    return Integral(f, low, up, 2);
}
```

Boole公式或Cotes公式

```
public static double Boole(UnaryFunction f, double low, double up)
{
    return Integral(f, low, up, 4);
}
```

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 复合求积公式 (CompositeIntegral类)

```
public static double Trapezoid(UnaryFunction f, double low, double up, int n = 1000)
```

```
{
```

```
    double h = (up - low) / n;
```

子区间长度

```
    double x = low, ans = 0;
```

```
    for (int i = 1; i < n; i++)
```

```
    {
```

```
        x += h;
```

```
        ans += f(x);
```

```
    }
```

中间项

```
    ans += f(low) / 2 + f(up) / 2;
```

边界项

```
    ans *= h;
```

```
    return ans;
```

```
}
```

将积分区间均匀
划分成n个子区间

$$T_n(f) = \frac{h}{2} \left[f(a) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b) \right]$$

$$= h \left[\sum_{k=1}^{n-1} f(x_k) + \frac{f(a)}{2} + \frac{f(b)}{2} \right]$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 复合求积公式 (CompositeIntegral类)

```
public static double Simpson(UnaryFunction f, double low, double up, int n = 1000)
```

```
{
```

```
    double h = (up - low) / n;
```

```
    double half = h / 2;
```

```
    double x = low, ans = 0;
```

```
    for (int i = 1; i < n; i++)
```

```
    {
```

```
        x += h;
```

```
        ans += 2 * f(x - half);
```

```
        ans += f(x);
```

```
    }
```

```
    ans += 2 * f(up - half) + f(low) / 2 + f(up) / 2;
```

```
    ans *= h / 3;
```

```
    return ans;
```

```
}
```

$$S_n(f) = \frac{h}{6} \left[f(a) + 4 \sum_{k=1}^n f(x_{k-0.5}) + 2 \sum_{k=1}^{n-1} f(x_k) + f(b) \right]$$
$$= \frac{h}{3} \left\{ \sum_{k=1}^{n-1} [2f(x_{k-0.5}) + f(x_k)] \right. \\ \left. + 2f(x_{n-0.5}) + \frac{f(a)}{2} + \frac{f(b)}{2} \right\}$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ Romberg求积公式 (Romberg类)

```
public static double Integral(UnaryFunction f, double low, double up,
    double limit = 1e-10, int maxCount = 15)
```

```
{
```

```
// .....
```

```
while (error > limit)
```

```
{
```

```
    n *= 2;
```

```
    oldTs = Ts;
```

```
    Ts = new Vector(count + 1);
```

```
    Ts[0] = CompositeIntegral.Trapezoid(f, low, up, n);
```

```
    for (int i = 1; i <= count; i++)
```

```
    {
```

```
        double beta = Math.Pow(4, i);
```

```
        Ts[i] = (beta * Ts[i - 1] - oldTs[i - 1]) / (beta - 1);
```

```
    }
```

```
    error = Math.Abs(Ts[count] - oldTs[count - 1]);
```

```
}
```

```
return Ts[count];
```

```
}
```

$$T(j, 0) = T\left(\frac{b-a}{2^j}\right) = T\left(\frac{b-a}{n}\right)$$

$$T(j, k) = \frac{4^k \cdot T(j, k-1) - T(j-1, k-1)}{4^k - 1}$$

$$|T(j, j) - T(j-1, j-1)|$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 求积方法对比 (Romberg类)

```
public static void Sample()  
{
```

```
    double w = 1;  
    UnaryFunction f = x => Math.Cos(w * x);  
    UnaryFunction If = x => 1 / w * Math.Sin(w * x);
```

$$f(x) = \cos(\omega x) \quad I(x) = \frac{1}{\omega} \sin(\omega x)$$

待积函数与原函数

```
    double low = 0, up = 2;  
    double I = If(up) - If(low);  
    Console.WriteLine("准确值: " + I + "\n");
```

积分上下界以及准确值

```
    int degree = 1;  
    double I1 = NewtonCotes.Integral(f, low, up, degree);  
    Console.WriteLine(degree + "次Newton-Cotes: " + I1);  
    Console.WriteLine("误差: " + (I1 - I) + "\n");
```

Newton-Cotes求积

```
    int n = 1;  
    double I2 = CompositeIntegral.Trapezoid(f, low, up, n);  
    Console.WriteLine(n + "段复合求积: " + I2);  
    Console.WriteLine("误差: " + (I2 - I) + "\n");
```

复合梯形求积

```
    double I3 = Romberg.Integral(f, low, up);  
    Console.WriteLine("Romberg: " + I3);  
    Console.WriteLine("误差: " + (I3 - I) + "\n");
```

Romberg求积

}

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

- 理解**插值型求积公式**的概念
- 掌握**代数精度**的概念和分析方法
- 理解**复合求积公式**的思路和计算过程
- 掌握**复合求积公式**的误差分析
- 理解**Romberg求积公式**的方法

作业

教材P273-1、 P273-2、 P274-6 、 P274-7

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

Gauss求积公式—再看代数精度



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

对于插值型求积公式：
$$\int_a^b f(x)dx \approx \sum_{k=0}^n A_k^{(n)} f(x_k) = (b-a) \sum_{k=0}^n C_k^{(n)} f(x_k)$$

$a = x_0 < x_1 < \dots < x_n = b$ 为等距离节点，此时

① 因为是 n 次多项式，所以**至少**具有 n 次代数精度

② 实际上，对于Lagrange插值型求积公式

n 为**偶数**次时，具有 $n+1$ 次代数精度（比如Simpson公式）

n 为奇数时，具有 n 次代数精度（比如梯形公式）



插值型求积公式的代数精度是否还能提高？

Gauss求积公式—再看代数精度



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

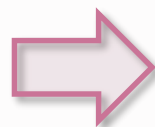
47/80

例8.5.1 (教材编号)

对如下求积公式, 试着理论分析, 确定 A_0, x_0, A_1, x_1 使得其代数精度尽可能高

$$\int_{-1}^1 f(x) dx \approx A_0 f(x_0) + A_1 f(x_1)$$

$$\left\{ \begin{array}{l} A_0 + A_1 = \int_{-1}^1 dx = 2 \\ A_0 x_0 + A_1 x_1 = \int_{-1}^1 x dx = 0 \\ A_0 x_0^2 + A_1 x_1^2 = \int_{-1}^1 x^2 dx = \frac{2}{3} \\ A_0 x_0^3 + A_1 x_1^3 = \int_{-1}^1 x^3 dx = 0 \end{array} \right.$$



令 $f(x) = 1, x, x^2, x^3 \dots$ 都准确

$$x_0 = -\frac{\sqrt{3}}{3}, x_1 = \frac{\sqrt{3}}{3} \quad A_0 = A_1 = 1$$

$$\int_{-1}^1 f(x) dx \approx f\left(-\frac{\sqrt{3}}{3}\right) + f\left(\frac{\sqrt{3}}{3}\right)$$

非等距的两点得到了比梯形公式更加高的代数精度!

什么是代数精度?



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

48/80

讨论加权式的积分： $\int_a^b \rho(x)f(x) dx \approx \sum_{k=0}^n A_k^{(n)} f(x_k)$

其中： $\rho(x)$ 为 $[a, b]$ 上的权函数

$a \leq x_0 < x_1 < \dots < x_n \leq b$ 为插值节点

不再限制等间距！

也不一定是闭区间！

显然根据之前的讨论可知： $A_k = \int_a^b \rho(x)l_k(x)dx$

余项： $E_n(f) = \frac{1}{(n+1)!} \int_a^b \rho(x) f^{(n+1)}(\xi) \omega_{n+1}(x) dx$

至少具有 n 次代数精度！

如果需要提高代数精度，则意味着对于 $f \in P^m[a, b], m > n$ 能得到 $E_n(f) = 0$

根据内积的定义， $f^{(n+1)}$ 构成的 $m-(n+1)$ 阶多项式和 $\omega_{n+1}(x)$ 正交！

还能取多少次？



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

记 $f^{(n+1)}$ 构成的多项式为 $p(x)$ ，我们先猜测 $p \in P^n[a, b]$ ，
即 $f \in P^{2n+1}[a, b]$ **代数精度为 $2n+1$**

根据正交多项式性质2，对任意 $p \in P^n$ ，只要找到一个 $n+1$
阶正交多项式 $\varphi_{n+1}(x)$

$$(\varphi_{n+1}, p) = \int_a^b \rho(x) \varphi_{n+1}(x) p(x) dx = 0 \quad \text{一定得到满足}$$

根据正交多项式性质3， $\varphi_{n+1}(x)$ 在**开区间** (a, b) 上有 $n+1$ 个
不同的零点

想一想 $\omega_{n+1}(x)$

$$\varphi_{n+1}(x) = a_{n+1}x^{n+1} + a_nx^n \dots = a_{n+1} \underline{(x - t_0)(x - t_1) \dots (x - t_n)}$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

结论：只要 $\omega_{n+1}(x)$ 中的节点 $x_0, x_1, \dots, x_n$ 和正交多项式 $\varphi_{n+1}(x)$ 的零点 $t_0, t_1, \dots, t_n$ 一致

$$\int_a^b \rho(x) \omega_{n+1}(x) p(x) dx = \frac{1}{a_{n+1}} \int_a^b \rho(x) \varphi_{n+1}(x) p(x) dx = 0$$

插值型求积公式具有 $2n+1$ 次代数精度

定理8.5.1（教材编号）

如上插值型求积公式具有 $2n+1$ 次代数精度的**充分必要**条件是求积公式的**节点**是求积区间上，权函数为 $\rho(x)$ 的 $n+1$ 次正交多项式的**零点**

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

证明

正交多项式零点 $\longleftrightarrow$ $2n+1$ 次代数精度

①必要性 如果插值求积具有 $2n+1$ 次代数精度, 令 $\omega_{n+1}(x) = (x - x_0)(x - x_1)\dots(x - x_n)$, 对任意 $g(x) \in P^n$, 有

$$\int_a^b \rho(x) \boxed{g(x) \omega_{n+1}(x)} dx = \sum_{k=0}^n A_k \overset{2n+1\text{次多项式}}{g(x_k) \omega_{n+1}(x_k)} = 0 \quad \text{准确相等, 不带余项!}$$

②充分性 如果 $x_0, x_1, \dots, x_n$ 是区间上带权的正交多项式零点, 则

$$q, r \in P^n, f(x) = q(x) \omega_{n+1}(x) + r(x) \quad = \sum_{k=0}^n A_k r(x_k) = \sum_{k=0}^n A_k f(x_k)$$

$$\int_a^b \rho(x) f(x) dx = \int_a^b \rho(x) \cancel{q(x) \omega_{n+1}(x)} dx + \int_a^b \rho(x) r(x) dx \quad \text{准确相等, 不带余项!}$$

$f \in P^{2n+1}$ 即求积公式具有 $2n+1$ 代数精度

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定义8.5.1 (教材编号)

若 $n+1$ 个节点的插值求积公式具有最高 $2n+1$ 的代数精度, 则称这种求积为**Gauss型求积公式**, 节点 $x_0, x_1, \dots, x_n$ 称为**Gauss点**

设 $[a, b]$ 区间上的**任意**相异节点 $x_0, x_1, \dots, x_n$, 令

$$f(x) = \omega_{n+1}^2(x) = (x - x_0)^2(x - x_1)^2 \dots (x - x_n)^2$$

$2n+2$ 次多项式

显然 $\int_a^b \rho(x)f(x)dx > 0$, 且 $\sum_{k=0}^n A_k f(x_k) = 0$

$$\Rightarrow \int_a^b \rho(x)f(x)dx \neq \sum_{k=0}^n A_k f(x_k)$$



无法提高到 $2n+2$ 次代数精度了! 有没有更好的代数精度?

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

定理8.5.2 (教材编号)

设 $f \in C^{2n+2}[a, b]$, 那么Gauss型求积公式的余项表达式为

$$E_n(f) = \frac{1}{(2n+2)!} f^{(2n+2)}(\xi) \int_a^b \rho(x) \omega_{n+1}^2(x) dx, \xi \in [a, b]$$

证明

根据Hermite插值公式



因为Lagrange插值不够用了, 我们的另外一个插值工具: 考虑导数!

$$H_{2n+1}(x_i) = f(x_i), H'_{2n+1}(x_i) = f'(x_i), i = 0, 1, \dots, n$$

$$f(x) = H_{2n+1}(x) + f[x_0, x_0, x_1, x_1, \dots, x_n, x_n, x] \omega_{n+1}^2(x)$$

用重节点Newton均差构建余项

Gauss型求积公式—余项



清华大学深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\int_a^b \rho(x) f(x) dx = \int_a^b \rho(x) H_{2n+1}(x) dx + \int_a^b \rho(x) f[x_0, x_0, x_1, x_1, \dots, x_n, x_n, x] \omega_{n+1}^2(x) dx$$

连续

$$\int_a^b \rho(x) H_{2n+1}(x) dx = \sum_{k=0}^n A_k H_{2n+1}(x_k) = \sum_{k=0}^n A_k f(x_k)$$

2n+1代数精度

插值条件

利用积分中值定理

上节课

$$E_1 = \int_a^b (x-a)(x-b) \frac{f(x) - L_1(x)}{(x-a)(x-b)} dx$$

$$E_n(f) = f[x_0, x_0, x_1, x_1, \dots, x_n, x_n, \eta] \int_a^b \rho(x) \omega_{n+1}^2(x) dx$$

$$= \frac{1}{(2n+2)!} f^{(2n+2)}(\xi) \int_a^b \rho(x) \omega_{n+1}^2(x) dx, \xi \in [a, b]$$

得证!

差分性质3

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

Newton-Cotes求积公式中，高阶的求积系数会出现负值，那Gauss求积公式如何？

定理8.5.3（教材编号）

Gauss型求积公式的求积系数 $A_k, k = 0, 1, \dots, n$ 皆为正

证明

假定带权的 $n+1$ 次正交多项式 $\varphi_{n+1}(x)$ 在区间内的 $n+1$ 个 $x_0, x_1, \dots, x_n$ 零点

构造 $f(x) = \underline{[l_k(x)]^2} = \prod_{\substack{j=0 \\ j \neq k}}^n \left(\frac{x - x_j}{x_k - x_j} \right)^2$

哪里来的思路？

2n次多项式

2n+1次多项式

回忆Hermite插值多项式的构造，令 $\alpha_i(x) = (a_i x + b_i) l_i^2(x)$

稳定性与收敛性—稳定性



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$f \in P^{2n} \Rightarrow \int_a^b \rho(x) f(x) dx = \sum_{i=0}^n A_i f(x_i) \quad \text{求积公式准确成立!}$$

根据插值基函数基本特点 $f(x_i) = \prod_{\substack{j=0 \\ j \neq k}}^n \left(\frac{x_i - x_j}{x_k - x_j} \right)^2 = \begin{cases} 1, i = k \\ 0, i \neq k \end{cases}$

$$A_k = \int_a^b \rho(x) f(x) dx = \int_a^b \rho(x) [l_k(x)]^2 dx > 0 \quad \text{得证!}$$

在上面证明中，隐含了对 A_k 新的表达，且 k 是任取的，即

$$A_k = \int_a^b \rho(x) [l_k(x)]^2 dx, k = 0, 1, \dots, n$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

记: $Q_n(f) = \sum_{k=0}^n A_k f(x_k)$

定理8.5.4 (教材编号)

设 $\tilde{f}(x_k)$ 为 $f(x_k)$, $k = 0, 1, \dots, n$ 的近似值, 则有

$$\left| Q_n(\tilde{f}) - Q_n(f) \right| \leq \max_{0 \leq k \leq n} \left| \tilde{f}(x_k) - f(x_k) \right| \int_a^b \rho(x) dx$$

证明

$$\begin{aligned} \left| Q_n(\tilde{f}) - Q_n(f) \right| &= \left| \sum_{k=0}^n A_k \tilde{f}(x_k) - \sum_{k=0}^n A_k f(x_k) \right| \\ &= \left| \sum_{k=0}^n A_k [\tilde{f}(x_k) - f(x_k)] \right| \leq \sum_{k=0}^n |A_k| \left| \tilde{f}(x_k) - f(x_k) \right| \leq \max_{0 \leq k \leq n} \left| \tilde{f}(x_k) - f(x_k) \right| \sum_{k=0}^n |A_k| \end{aligned}$$

稳定性与收敛性—收敛性



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

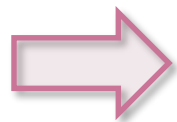
数值微分

Taylor展开构造

插值求导与外推

编程实践

$$\int_a^b \rho(x) f(x) dx \approx \sum_{l=0}^n A_l f(x_l)$$



取 $f(x) \equiv 1$, 积分公式准确成立 $\int_a^b \rho(x) dx = \sum_{l=0}^n A_l$

$$\left| Q_n(\tilde{f}) - Q_n(f) \right| \leq \max_{0 \leq k \leq n} \left| \tilde{f}(x_k) - f(x_k) \right| \sum_{k=0}^n |A_k| = \max_{0 \leq k \leq n} \left| \tilde{f}(x_k) - f(x_k) \right| \int_a^b \rho(x) dx$$

得证!

定理8.5.5 (教材编号)

设 $f \in C[a, b]$, 令 $Q_n(f) = \sum_{k=0}^n A_k^{(n)} f(x_k^{(n)})$, 则有

$$\lim_{n \rightarrow \infty} Q_n(f) = \int_a^b \rho(x) f(x) dx$$

随着次数 n 的增加, Gauss型求积公式**收敛**!

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

刚才例题中，采用正交化，也可以采用Legendre多项式直接表达，**只差一个首项系数！**

设区间 $[a, b] = [-1, 1]$ ，权函数 $\rho(x) \equiv 1$ ，则相应的正交多项式为Legendre多项式

$$P_0(x) = 1, P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n, n = 1, 2, \dots$$

$$(n + 1)P_{n+1}(x) = (2n + 1)xP_n(x) - nP_{n-1}(x)$$

$$P_1 = \frac{1}{2} \frac{d}{dx} (x^2 - 1) = x, P_2(x) = \frac{1}{2} (3x^2 - 1), P_3(x) = \dots$$

求 P_{n+1} 的根 x_k 及相应的 $A_k, k = 0, 1, \dots, n$, $\int_{-1}^1 f(x) dx \approx \sum_{k=0}^n A_k f(x_k)$

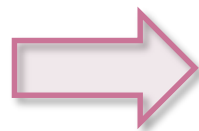
Gauss-Legendre求积公式



如果区间不是 $[a, b] = [-1, 1]$ 怎么办?

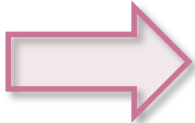
令任意区间 $[a, b]$

$$x = \frac{a+b}{2} + \frac{b-a}{2}t, t \in [-1, 1]$$



$$x(t = -1) = a, x(t = 1) = b$$

$$\int_a^b f(x)dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt$$


$$\int_a^b f(x)dx \approx \frac{b-a}{2} \sum_{k=0}^n A_k f\left(\frac{a+b}{2} + \frac{b-a}{2}t_k\right)$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

Gauss-Chebyshev求积公式



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

同样可以用Chebyshev多项式构造

权函数怎么办?

设区间 $[a, b] = [-1, 1]$, 权函数 $\rho(x) = \frac{1}{\sqrt{1-x^2}}$, 则相应的正交多项式为Chebyshev多项式:

$$T_{n+1}(x) = \cos((n+1) \cos^{-1} x)$$

虽然是个多项式, 其实又是一个三角函数!

零点: $(n+1) \cos^{-1}(x) = \frac{\pi}{2} + k\pi$

无需递推, 也无需查表

$$x_k = \cos\left(\frac{2k+1}{2(n+1)}\pi\right), k = 0, 1, \dots, n$$

$$A_k = \frac{\pi}{n+1}, k = 0, 1, \dots, n$$

$$\int_{-1}^1 \frac{f(x)}{\sqrt{1-x^2}} dx \approx \frac{\pi}{n+1} \sum_{k=0}^n f(x_k)$$

$$E_n(f) = \frac{\pi}{2^{2n+1}(2n+2)!} f^{(2n+2)}(\xi)$$

Gauss-Chebyshev求积公式



清华大学 深圳国际研究生院

Tsinghua Shenzhen International Graduate School

海洋工程研究院 胡振中 《工程硕士数学》 第七章

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

考虑变量代换以后 $\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt$

构造符合Chebyshev多项式的的权函数

$$\frac{b-a}{2} \int_{-1}^1 \frac{1}{\sqrt{1-t^2}} \sqrt{1-t^2} f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right) dt \approx \frac{b-a}{2} \frac{\pi}{n+1} \sum_{k=0}^n \sqrt{1-t_k^2} f\left(\frac{a+b}{2} + \frac{b-a}{2}t_k\right)$$

$t_k = \cos\left(\frac{2k+1}{2(n+1)}\pi\right), k=0,1,\dots,n$

x_k

另外还有Gauss-Laguerre、Gauss-Hermite型式的求积公式，积分限可以取至 $(0, +\infty)$ 或者 $(-\infty, +\infty)$ ，不再赘述

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

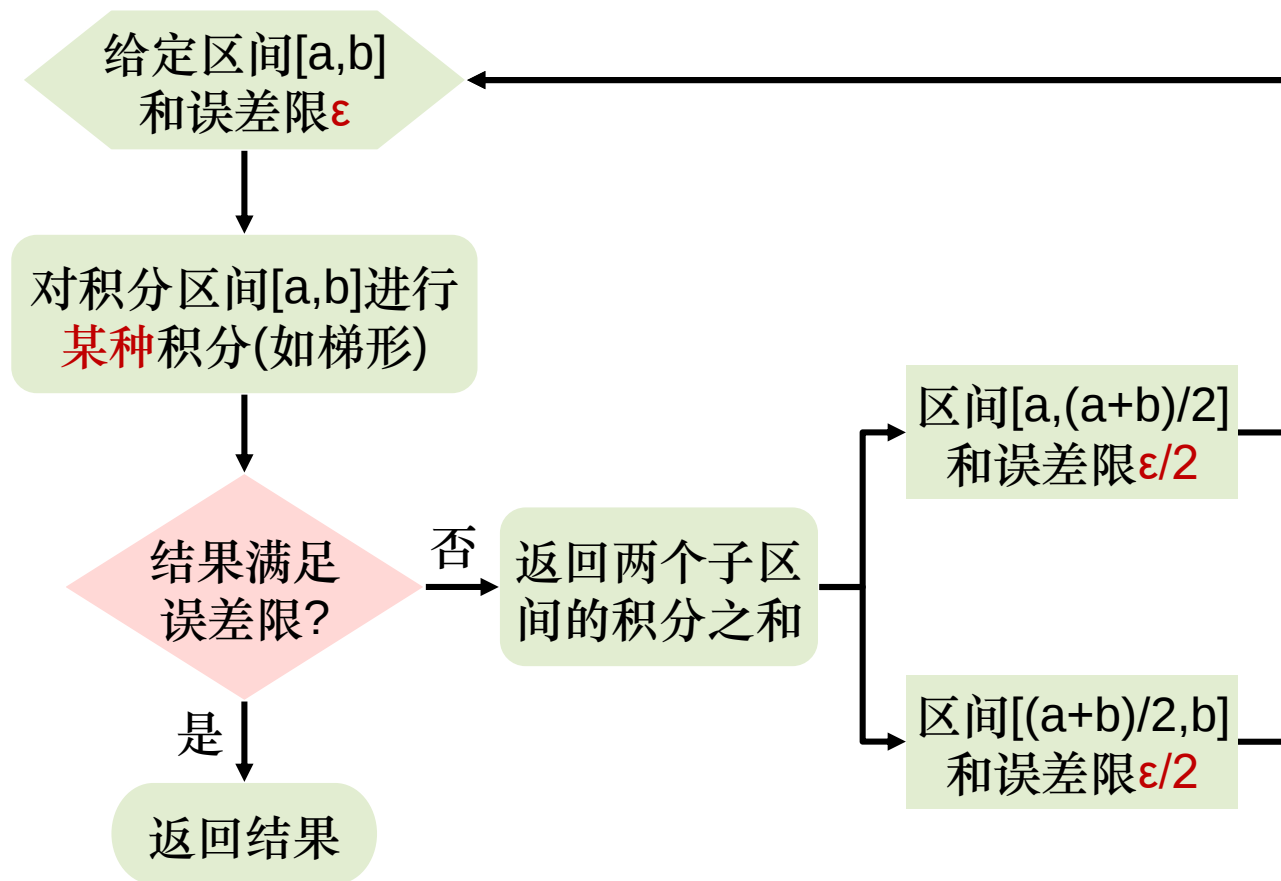
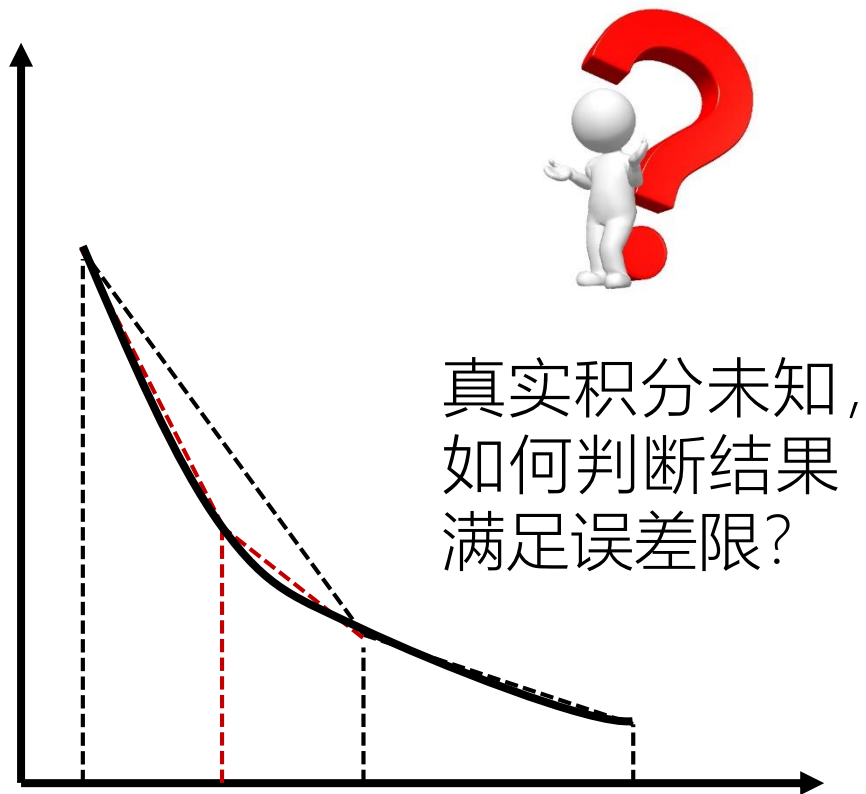
数值微分

Taylor展开构造

插值求导与外推

编程实践

被积函数可能变化不同，采用均匀划分不够经济



变步长的复合梯形（复合Simpson）

自适应积分思路

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

Simpson求积公式的误差

$$e_1 = I(f) - I_1(f) = -\frac{(b-a)^5}{2880} f^{(4)}(\xi)$$

2等分复合Simpson求积公式的误差

$$e_2 = I(f) - I_2(f) = -\frac{b-a}{2880} h^4 f^{(4)}(\eta) = -\frac{1}{16} \times \frac{(b-a)^5}{2880} f^{(4)}(\eta)$$

假定 $f^{(4)}(\xi) \approx f^{(4)}(\eta)$ $e_1 \approx 16e_2$ $\Rightarrow$ $e_2 \approx \frac{e_1 - e_2}{15} = \frac{I_2 - I_1}{15}$

某种积分 $\rightarrow$ 2等分复合Simpson积分

可以用两积分之差表示!

$|I_2 - I_1| < 15\varepsilon$ 时, 可认为 $|e_2| < \varepsilon$, 偏保守取条件 $|I_2 - I_1| < 10\varepsilon$

注意 I_1 为Simpson求积结果, 仅用于估计误差

自适应Simpson积分

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

如果要计算 $f(x)$ 在 x_0 的导数, 可以令一个充分小的步长 h

Taylor展开

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{1}{2} f''(\xi^+) h^2 \Rightarrow f'(x_0) \approx \frac{f(x_0 + h) - f(x_0)}{h}$$

截断误差 $-\frac{1}{2} f''(\xi^+) h$ 向前差分公式

$$f(x_0 - h) = f(x_0) - f'(x_0)h + \frac{1}{2} f''(\xi^-) h^2 \Rightarrow f'(x_0) \approx \frac{f(x_0) - f(x_0 - h)}{h}$$

截断误差 $\frac{1}{2} f''(\xi^-) h$ 向后差分公式

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

Taylor展开

$$f(x_0 + h) = f(x_0) + f'(x_0)h + \frac{1}{2}f''(x_0)h^2 + \frac{1}{6}f'''(\xi^+)h^3$$

$$f(x_0 - h) = f(x_0) - f'(x_0)h + \frac{1}{2}f''(x_0)h^2 - \frac{1}{6}f'''(\xi^-)h^3$$

$$f'(x_0) \approx \frac{f(x_0 + h) - f(x_0 - h)}{2h} \quad \text{截断误差: } \frac{1}{6}f'''(\xi)h^2$$

也称**中点方法**，是前两种方法的**算术平均**

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

如果可以找到一个插值多项式 $f(x) \approx P(x)$, 则

$$f'(x) \approx P'(x) = \sum_{k=0}^n f(x_k) l'_k(x) \quad \text{插值型求导公式}$$

三点公式 $f'(x) \approx P'_2(x) = \frac{1}{2h} [-f(x-h) + f(x+h)] - \frac{h^2}{6} f'''(\xi)$

实用五点公式

$$f'(x) \approx P'_4(x) \approx \frac{1}{12h} [f(x-2h) - 8f(x-h) + 8f(x+h) - f(x+2h)]$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

另一方面, 如果 $f'(x) \approx G(h) = \frac{f(x_0+h)-f(x_0-h)}{2h}$

$$f'(x) - G(h) = a_1 h^2 + a_2 h^4 + \dots \Rightarrow \begin{cases} G_0(h) = G(h) \\ G_k(h) = \frac{4^k G_{k-1}\left(\frac{h}{2}\right) - G_{k-1}(h)}{4^k - 1} \end{cases}$$

数值微分的Richardson外推

示例: 外推法计算 $f(x) = xe^x$ 在 $x = 2$ 处的导数值, 取 $h = 0.2$

$$G(h) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} = \frac{1}{2h} [(2 + h)e^{2+h} - (2 - h)e^{2-h}]$$

插值型求导公式与外推算法



概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

$$G(h) = 22.414160$$

$$G\left(\frac{h}{2}\right) = 22.228786$$

$$G\left(\frac{h}{4}\right) = 22.182564$$

$$G_1(h) = 22.166995$$

$$G_1\left(\frac{h}{2}\right) = 22.167157$$

$$G_2(h) = 22.167168$$

准确值 $f'(2.0) = 22.167168$, $G\left(\frac{h}{4}\right)$ 仅有3位有效数字, $G_2(h)$ 有8位

示例: 外推法计算 $f(x) = xe^x$ 在 $x = 2$ 处的导数值, 取 $h = 0.2$

$$G(h) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} = \frac{1}{2h} [(2 + h)e^{2+h} - (2 - h)e^{2-h}]$$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

7.1 概论及背景知识

7.2 Newton-Cotes求积公式和复合求积公式

7.3 Gauss型求积公式

7.4 自适应积分法*

7.5 数值微分*

7.6 编程实践（穿插在两周中）

✓ Gauss-Legendre公式 (GaussLegendre类)

代码与Newton-Cotes公式类似

```
public static double Integral(UnaryFunction f, double low, double up, int degree = 1)
```

{

```
    Vector xs, ks;
```

```
    if (degree == 0)
```

{

```
        xs = new Vector(0.0);
```

```
        ks = new Vector(2.0);
```

}

```
    else if (degree == 1)
```

{

```
        xs = new Vector(-0.5773502692, 0.5773502692);
```

```
        ks = new Vector(1, 1);
```

}

```
    // .....
```

```
    double scale = (up - low) / 2;
```

```
    return xs.Mapping(x => f((x + 1) * scale + low)) * ks * scale;
```

}

n	x_k	A_k	n	x_k	A_k
0	0	2	5	$\pm 0.932\ 469\ 514\ 2$	0.171 324 492 4
1	$\pm 0.577\ 350\ 269\ 2$	1		$\pm 0.661\ 209\ 386\ 5$	0.360 761 573 0
2	$\pm 0.774\ 596\ 669\ 2$	0.555 555 555 6		$\pm 0.238\ 619\ 186\ 1$	0.467 913 934 6
	0	0.888 888 888 9	6	$\pm 0.949\ 107\ 912\ 3$	0.129 484 966 2
3	$\pm 0.861\ 136\ 311\ 6$	0.347 854 845 1		$\pm 0.741\ 531\ 185\ 6$	0.279 705 391 5
	$\pm 0.339\ 981\ 043\ 6$	0.652 145 154 9		$\pm 0.405\ 845\ 151\ 4$	0.381 830 050 5
4	$\pm 0.906\ 179\ 845\ 9$	0.236 926 885 1		0	0.417 959 183 7
	$\pm 0.538\ 469\ 310\ 1$	0.478 628 670 5	7	$\pm 0.960\ 289\ 856\ 5$	0.101 228 536 3
	0	0.568 888 888 9		$\pm 0.796\ 666\ 477\ 4$	0.222 381 034 5
				$\pm 0.525\ 532\ 409\ 9$	0.313 706 645 9
				$\pm 0.183\ 434\ 642\ 5$	0.362 683 783 4

高斯点位置

每个点对应的求积系数

$$\frac{b-a}{2} \sum f \left[\frac{b-a}{2} (x_k + 1) + a \right] \cdot A_k$$

区间变换

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

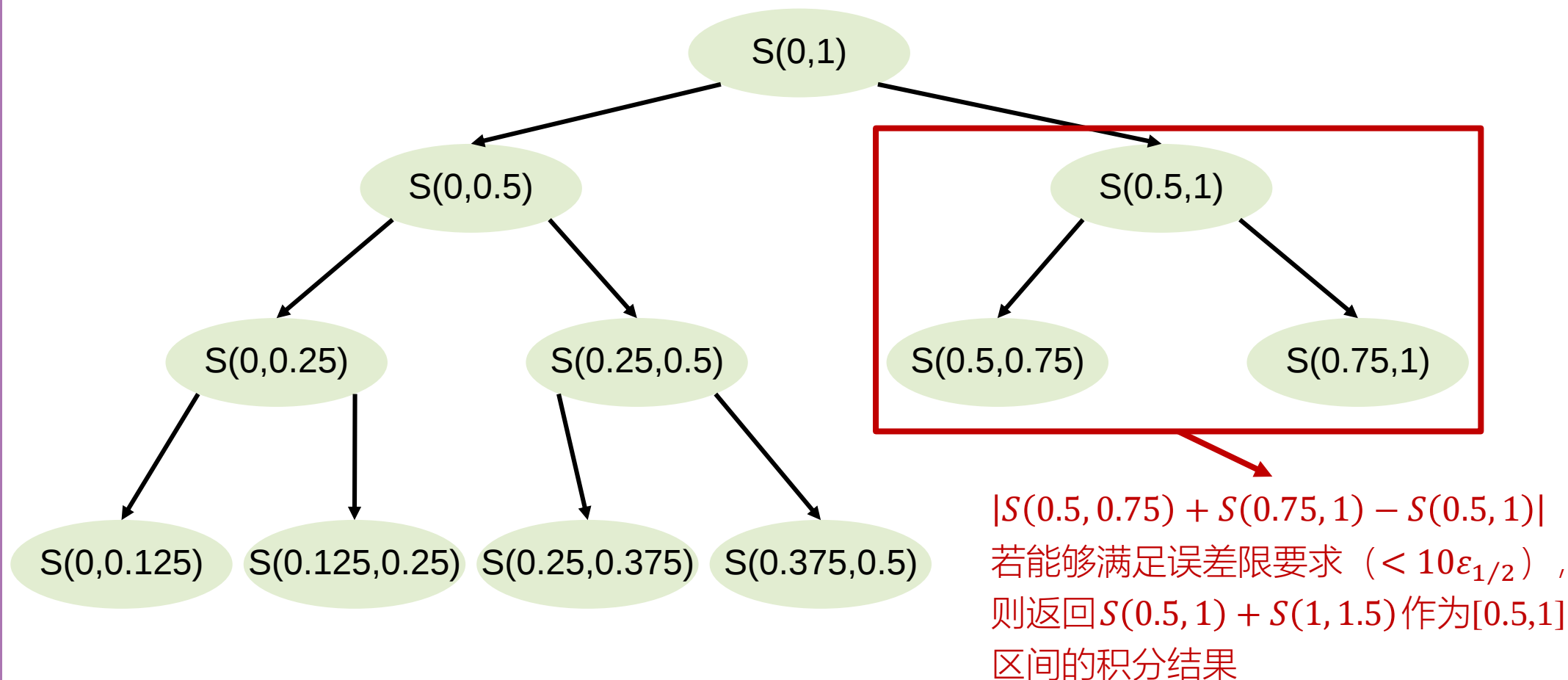
数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 自适应积分法 (AdaptiveIntegral类)



相当于构造一棵二叉树，其中 $S(0,1)$ 为根节点，每个节点的左、右子节点分别为其左、右半区间的Simpson积分结果

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 自适应积分法 (AdaptiveIntegral类)

简化写法, 供参考编程思路

```
protected static double SimpsonSimpRe(UnaryFunction f, double low, double up,
    double parent, double limit, int maxCount, ref int count)
{
    count++;
    double mid = (low + up) / 2;
    double left = NewtonCotes.Simpson(f, low, mid);
    double right = NewtonCotes.Simpson(f, mid, up);
    if (Math.Abs(left + right - parent) < 10 * limit)
        return left + right;
    return SimpsonSimpRe(f, low, mid, left, limit / 2, maxCount, ref count)
        + SimpsonSimpRe(f, mid, up, right, limit / 2, maxCount, ref count);
}

protected static double SimpsonSimp(UnaryFunction f, double low, double up,
    double limit = 1e-10, int maxCount = 10000)
{
    double root = NewtonCotes.Simpson(f, low, up);
    int count = 0;
    return SimpsonSimpRe(f, low, up, root, limit, maxCount, ref count);
}
```

记录函数被调用次数, 即区间划分数

计算当前节点的左、右子节点

满足要求则返回子节点之和

不满足要求
则对子节点
进一步划分

计算根节点S(a,b)

调用递归函数

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 求积方法对比 (AdaptiveIntegral类)

```
public static void Sample()  
{
```

```
    UnaryFunction f = x => 0.5 / Math.Sqrt(x);  
    UnaryFunction If = x => Math.Sqrt(x);
```

$$f(x) = \frac{1}{2\sqrt{x}} \quad I(x) = \sqrt{x}$$

待积函数与原函数

```
    double low = 0.1, up = 1;  
    double I = If(up) - If(low);  
    Console.WriteLine("准确值: " + I + "\n");
```

积分上下界以及准确值

```
// .....
```

```
    double I4 = GaussLegendre.Integral(f, low, up, degree);  
    Console.WriteLine(degree + "次Gauss-Legendre: " + I4);  
    Console.WriteLine("误差: " + (I4 - I) + "\n");
```

Gauss-Legendre求积

```
// .....
```

```
    double I5 = AdaptiveIntegral.Simpson(f, low, up);  
    Console.WriteLine("自适应Simpson: " + I5);  
    Console.WriteLine("误差: " + (I5 - I) + "\n");
```

自适应Simpson求积

```
}
```

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 求导 (Differential类)

```
public static double ThreePoint(UnaryFunction f, double x0, double h = 1e-5)    三点公式
{
    return (f(x0 + h) - f(x0 - h)) / h / 2;
}
```

```
public static double FivePoint(UnaryFunction f, double x0, double h = 1e-5)    五点公式
{
    return (f(x0 - 2 * h) - 8 * f(x0 - h) + 8 * f(x0 + h) - f(x0 + 2 * h)) / h / 12;
}
```

```
public static double Order(UnaryFunction f, double x0, int order = 1, double h = 1e-5);

    求函数任意阶导数的方法，误差为 $O(h^2)$ 
```

```
public static double Richardson(UnaryFunction f, double x0, int order = 1,
    double limit = 1e-10, int maxCount = 15)
```

基于Order方法，采用Richardson外推的求函数任意阶导数，通过误差限确定外推次数
微分步长 h 与外推次数 n 的关系为： $h = 1/2^n$

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 求导 (Differential类)

```
public static void Sample()
```

```
{
```

```
    UnaryFunction f = x => Math.Exp(x);
```

待求导函数

```
    double x0 = 0, f1 = 1;
```

目标点及对应的导数值

```
    double h = 1;
```

```
    // 三点公式
```

```
    Console.WriteLine("三点公式: " + (ThreePoint(f, x0, h) - f1));
```

```
    // 五点公式
```

```
    Console.WriteLine("五点公式: " + (FivePoint(f, x0, h) - f1));
```

用三点公式和五点公式计算一阶导数

```
    int degree = 2;
```

```
    Console.WriteLine(degree + "阶导数");
```

```
    Console.WriteLine("常规求导: " + (Order(f, x0, degree, h) - f1));
```

```
    Console.WriteLine("外推求导: " + (Richardson(f, x0, degree, limit: 1e-5) - f1));
```

计算degree阶导数

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

✓ 附：简化代码的方法（扩展方法）

```
public static class UnaryFunctionExtension
```

```
{
```

```
    public static double Integral(this UnaryFunction f, double low, double up,  
        double limit = 1e-10, int maxCount = 10000)
```

自适应积分

```
    => AdaptiveIntegral.Simpson(f, low, up, limit, maxCount);
```

```
    public static double Differential(this UnaryFunction f, double x0, int order = 1,  
        double limit = 1e-10, int maxCount = 15)
```

外推高阶微分

```
    => NumNet.Differential.Richardson(f, x0, order, limit, maxCount);
```

```
    public static double FindZero(this UnaryFunction f, double low, double up)
```

```
    => Bisection.FindZero(f, low, up);
```

二分法求零点

```
    public static double FindZero(this UnaryFunction f, double x0)
```

```
    => NewtonIteration.FindZero(f, x0, x0 + 1e-5);
```

割线法求零点

```
}
```

可以直接调用`f.Integral(low, up)`、`f.Differential(x0)`、`f.FindZero(x0)`进行一元函数的积分、微分以及求零点等数值计算了！

概论

研究问题

插值型求积公式

代数精度

N-C求积公式

Cotes系数

误差与稳定性

复合公式

Romberg公式

Gauss求积公式

Gauss求积

稳定性与收敛性

G-L求积

G-C求积

自适应积分法

数值微分

Taylor展开构造

插值求导与外推

编程实践

- 理解**插值型求积公式**的概念
- 掌握**代数精度**的概念和分析方法
- 理解**复合求积公式**的思路和计算过程
- 掌握**复合求积公式**的误差分析
- 理解**Romberg求积公式**的方法
- 掌握**Gauss(-L/-C)求积公式**的计算方法
- 了解**自适应积分法**和**数值微分**的思路

作业

教材P274-10, 补充: 用三点G-C公式计算 $\int_{-1}^1 (1-x^2)^{3/2} \cos x \, dx$