



# 计算方法

## Ch03 数值积分

程勇

buctcourse@163.com  
<http://www.buct.edu.cn>

Dept. of Computer  
Beijing University of Chemical Technology

March 1, 2017



# 提纲

- ① Ch01 概论
- ② Ch02 插值方法
- ③ Ch03 数值积分
- ④ Ch04 方程求根的迭代法
- ⑤ Ch05 线性方程组的迭代法
- ⑥ Ch06 线性方程组的直接法
- ⑦ Ch07 常微分方程的差分法



# 本章提纲

机械求积

Newton-Cotes 公式

复化求积公式

龙贝格算法

Gauss 公式



## 传统求积方法

对于积分

$$\int_a^b f(x) dx$$

如果知道  $f(x)$  的原函数  $F(x)$ , 则由 Newton-Leibniz 公式可求得:

$$\int_a^b f(x) dx = F(x) \big|_a^b = F(b) - F(a)$$

但事实上, 我们不难发现, 微积分方法求积分也确有其局限性:

- ①  $f(x)$  根本不存在, 往往只是给出一张数据表来表示函数的关系;
- ② 被积函数没有初等函数表示的原函数;
- ③  $f(x)$  的表达式结构复杂, 求原函数较困难;



## 数值积分求解方法

显然这些情况下将无法运用 Newton-Leibniz 来求积分，而需要建立定积分的近似计算公式。这类方法很多，但最常用的一种方法是利用插值多项式来构造数值求积公式，具体步骤如下：

- ① 构造一个多项式  $p(x)$  逼近被积函数  $f(x)$ ；
- ② 以  $p(x)$  的积分近似代替  $f(x)$  的积分；即

$$\int_a^b f(x) dx \approx \int_a^b p(x) dx$$

多项式  $p(x)$  的积分则是大家都会计算的，下面就根据这一想法构造计算定积分的各种近似计算公式。



# 机械求积

依据积分中值定理, 对于连续函数  $f(x)$ , 在  $[a, b]$  内存在一点  $\xi$ , 成立

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

即底为  $b - a$  而高为  $f(\xi)$  的矩形面积恰恰等于所求曲边梯形的面积。称  $f(\xi)$  为区间  $[a, b]$  上的平均高度, 而这一平均高度是很难计算的, 但可以采用另一类办法来解决这一问题, 就是对平均高度  $f(\xi)$  提供一种近似算法。

$$\int_a^b f(x) dx \approx (b - a) \sum_{i=0}^n \lambda_i f(x_i)$$

一般地, 在求积分时, 取  $[a, b]$  内若干个节点  $x_i$  处的高度  $f(x_i)$ , 通过加权平均的方法得出平均高度  $f(\xi)$  的近似值, 这类求积方法称机械求积, 式中  $x_i$  称为求积节点,  $\lambda_i$  称为求积系数。



## 梯形求积公式

过两点  $a, b$  作直线 (两点 Lagrange 插值)

$$p_1(x) = \frac{x-b}{a-b}f(a) + \frac{x-a}{b-a}f(b)$$

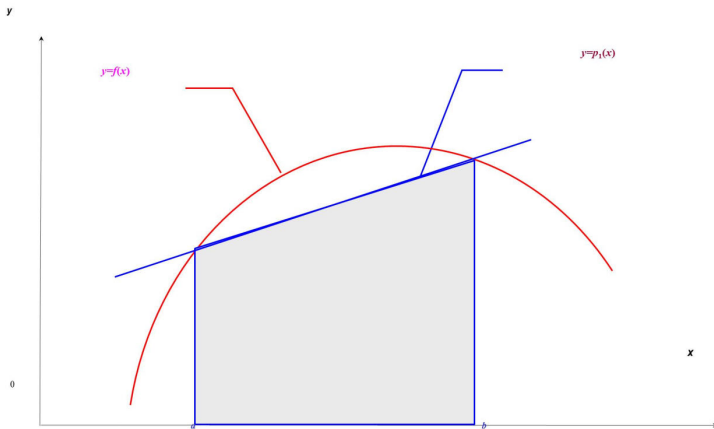
用  $p_1(x)$  代替  $f(x)$  得:

$$\begin{aligned} \int_a^b f(x)dx &\approx \int_a^b p_1(x)dx \\ &= \int_a^b \left( \frac{x-b}{a-b}f(a) + \frac{x-a}{b-a}f(b) \right) dx \\ &= (b-a) \left( \frac{1}{2}f(a) + \frac{1}{2}f(b) \right) \\ &= \frac{b-a}{2} (f(a) + f(b)) \end{aligned}$$

称为梯形求积公式 (两点公式)。



# 梯形求积公式几何意义



梯形求积





## 梯形求积例子

例：利用梯形求积公式计算定积分：

$$\int_{0.5}^1 \sqrt{x} dx$$

解：依梯形求积公式有：

$$\begin{aligned}\int_{0.5}^1 \sqrt{x} dx &\approx \frac{1 - 0.5}{2} (\sqrt{0.5} + \sqrt{1}) \\ &= 0.4267767\end{aligned}$$

积分准确值为：

$$\begin{aligned}\int_{0.5}^1 \sqrt{x} dx &= \frac{2}{3} x^{\frac{3}{2}} \Big|_{0.5}^1 \\ &= 0.43096441\end{aligned}$$



# 求积公式的代数精度

## 机械求积公式

$$\int_a^b f(x) dx \approx (b-a) \sum_{i=0}^n \lambda_i f(x_i)$$

具有  $m$  阶 (代数) 精度, 如果它对于任意次数不超过  $m$  次多项式  $f(x) = p_k(x) (0 \leq k \leq m)$  是准确的, 但对于  $m+1$  次多项式  $f(x) = p_{m+1}(x)$  不准确。即有:

$$\int_a^b p_k(x) dx = (b-a) \sum_{i=0}^n \lambda_i p_k(x_i) \quad k = 0, 1, 2, \dots, m$$

$$\int_a^b p_{m+1}(x) dx \neq (b-a) \sum_{i=0}^n \lambda_i p_{m+1}(x_i)$$



## 求积公式的代数精度 (续)

求积公式代数精度的判断可以简化如下：如果求积公式对于任意次数不超过  $m$  次的幂函数  $f(x) = x^k (0 \leq k \leq m)$  是准确的，但对于  $m+1$  次幂函数  $f(x) = x^{m+1}$  不准确，则可判定求积公式：

$$\int_a^b f(x) dx \approx (b-a) \sum_{i=0}^n \lambda_i f(x_i)$$

具有  $m$  阶 (代数) 精度。

**例：判断梯形求积公式的代数精度。**

对梯形求积公式：

$$\int_a^b f(x) dx \approx \frac{b-a}{2} (f(a) + f(b))$$



## 求积公式的代数精度 (续)

当  $f(x) = 1$  时

$$\int_a^b f(x) dx = \int_a^b 1 dx = b - a$$
$$\frac{b-a}{2}(f(a) + f(b)) = \frac{b-a}{2}(1 + 1) = b - a$$

因此有:

$$\int_a^b f(x) dx = \frac{b-a}{2}(f(a) + f(b))$$

当  $f(x) = x$  时

$$\int_a^b f(x) dx = \int_a^b x dx = \frac{1}{2}(b^2 - a^2)$$
$$\frac{b-a}{2}(f(a) + f(b)) = \frac{b-a}{2}(a + b) = \frac{1}{2}(b^2 - a^2)$$



## 求积公式的代数精度 (续)

因此有：

$$\int_a^b f(x) dx = \frac{b-a}{2} (f(a) + f(b))$$

当  $f(x) = x^2$  时

$$\begin{aligned} \int_a^b f(x) dx &= \int_a^b x^2 dx = \frac{1}{3}(b^3 - a^3) \\ \frac{b-a}{2}(f(a) + f(b)) &= \frac{b-a}{2}(a^2 + b^2) \end{aligned}$$

因此：

$$\int_a^b f(x) dx \neq \frac{b-a}{2} (f(a) + f(b))$$

由定义可知，梯形求积公式具有 1 次代数精度。



## Newton-Cotes 公式

Newton-Cotes 公式是指在等距节点下使用 Lagrange 插值多项式建立的数值求积公式。

设将求积区间  $[a, b]$  划分为  $n$  等分, 选取等分点

$$x_i = a + ih, \quad h = \frac{b - a}{n}, \quad i = 0, 1, 2, \dots, n$$

作为求积节点构造求积公式

$$\int_a^b f(x) dx \approx (b - a) \sum_{i=0}^n \lambda_i f(x_i)$$

这种求积公式称为 Newton-Cotes 公式。



## Simpson 求积公式

将求积区间  $[a, b]$  两等分, 过三点  $a, (a+b)/2, b$  作抛物线  
(三点 Lagrange 插值)

$$\begin{aligned} p_2(x) = & \frac{2}{(b-a)^2} \left( x - \frac{a+b}{2} \right) (x-b) f(a) \\ & - 2 \frac{2}{(b-a)^2} (x-a) (x-b) f\left(\frac{a+b}{2}\right) \\ & + \frac{2}{(b-a)^2} (x-a) \left( x - \frac{a+b}{2} \right) f(b) \end{aligned}$$

用  $p_2(x)$  代替  $f(x)$  得:

$$\int_a^b f(x) dx \approx \int_a^b p_2(x) dx = \frac{b-a}{6} \left( f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right)$$

称为 Simpson 公式, 这是 Newton-Cotes 公式的一个特例。



## Simpson 公式计算例子

例：利用 Simpson 公式计算定积分：

$$\int_{0.5}^1 \sqrt{x} dx$$

解：由 Simpson 公式：

$$\begin{aligned} \int_{0.5}^1 \sqrt{x} dx &\approx \frac{1 - 0.5}{6} (\sqrt{0.5} + 4\sqrt{0.75} + \sqrt{1}) \\ &= 0.43093403 \end{aligned}$$

积分准确值为：

$$\int_{0.5}^1 \sqrt{x} dx = \frac{2}{3} x^{\frac{3}{2}} \Big|_{0.5}^1 = 0.43096441$$

同样可以证明 Simpson 求积公式具有三次代数精度。





## Newton-Cotes 公式

将求积区间  $[a, b]$  划分为  $n$  等分, 选取等分点

$$x_i = a + ih, \quad h = \frac{b-a}{n}, \quad i = 0, 1, 2, \dots, n$$

构造  $n$  次 Lagrange 插值

$$L_n(x) = \sum_{i=0}^n \varphi_i(x) f(x_i) = \sum_{i=0}^n \left( \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} \right) f(x_i)$$

用  $L_n(x)$  代替  $f(x)$  得:

$$\begin{aligned} \int_a^b f(x) dx &\approx \int_a^b L_n(x) dx \\ &= \int_a^b \sum_{i=0}^n \left( \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} \right) f(x_i) dx \end{aligned}$$



## Newton-Cotes 公式 (续)

$$= \sum_{i=0}^n A_i f(x_i) = (b-a) \sum_{i=0}^n \lambda_i f(x_i)$$

其中

$$A_i = \int_a^b \left( \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j} \right) dx$$

称为 Newton-Cotes 公式。梯形求积公式和 Simpson 公式是 Newton-Cotes 公式当  $n=1$  和  $n=2$  时的两个特例。

$$\int_a^b f(x) dx = \frac{b-a}{2} (f(a) + f(b)) \quad \text{梯形公式}$$

$$\int_a^b f(x) dx = \frac{b-a}{6} \left( f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right) \quad \text{Simpson 公式}$$



# Newton-Cotes 公式求积系数

$n$	求积系数					
1	1/2	1/2				
2	1/6	4/6	1/6			
3	1/8	3/8	3/8	1/8		
4	7/90	16/45	2/15	16/45	7/90	
5	19/288	25/96	25/144	25/144	25/96	19/288



## Newton-Cotes 求积公式例子

例题：利用 Newton-Cotes 公式 (取  $n = 4$ , 此时称为 Cotes 公式) 计算定积分：

$$\int_{0.5}^1 \sqrt{x} dx$$

解：根据 Newton-Cotes 公式：

$$\begin{aligned} \int_{0.5}^1 \sqrt{x} dx &\approx \frac{1 - 0.5}{90} (7\sqrt{0.5} + 32\sqrt{0.625} \\ &\quad + 12\sqrt{0.75} + 32\sqrt{0.875} + 7) \\ &= 0.43096407 \end{aligned}$$

积分准确值为：

$$\begin{aligned} \int_{0.5}^1 \sqrt{x} dx &= \frac{2}{3} x^{\frac{3}{2}} \Big|_{0.5}^1 \\ &= 0.43096441 \end{aligned}$$



## Newton-Cotes 公式代数精度

可以证明 Newton-Cotes 公式的代数精度至少为  $n$ 。且当  $n$  为偶数时，代数精度可达  $n+1$ 。如：

$n=1$  时 (梯形公式)，代数精度为 1；

$n=2$  时 (Simpson 公式)，代数精度为  $2+1=3$ ；

$n=4$  时 (Cotes 公式)，代数精度为  $4+1=5$ ，以此类推。



## 复化求积公式

为了提高求积公式的精度，又使算法简单易行，往往使用复合方法，即将积分区间  $[a, b]$  分成若干个子区间，然后在每个小区间上使用低阶 Newton-Cotes 公式，最后将每个小区间上的积分的近似值相加作为积分的近似值，这种方法称为复化求积法。

复化梯形公式

将求积区间  $n$  等分，步长  $(b-a)/n$ ，分点为  $x_i = a + ih (i = 0, 1, 2, \dots, n)$ ，对每个小区间  $[x_i, x_{i+1}]$  使用梯形求积公式，则

$$\begin{aligned} I = \int_a^b f(x) dx &= \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \approx \sum_{i=0}^{n-1} \frac{x_{i+1} - x_i}{2} (f(x_i) + f(x_{i+1})) \\ &= \frac{b-a}{2n} (f(a) + 2 \sum_{i=1}^{n-1} f(x_i) + f(b)) = T_n \end{aligned}$$



## 复化梯形求积例子

例题：已知  $f(x) = x^3 - 2x + 1$  由表给出

$x$	1	3/2	2	5/2	3
$f(x)$	0	11/8	5	93/8	22

用复化梯形公式近似计算定积分  $\int_1^3 f(x) dx$ 。

解：此时  $n = 4$

$$\begin{aligned}
 T_4 &= \frac{b-a}{2n} (f(a) + 2 \sum_{i=1}^{n-1} f(x_i) + f(b)) \\
 &= \frac{3-1}{2 \cdot 4} (f(1) + 2(f(3/2) + f(2) + f(5/2)) + f(3)) \\
 &= \frac{1}{4} (0 + 2(11/8 + 5 + 93/8) + 22) = 14.5
 \end{aligned}$$

实际上  $\int_1^3 (x^3 - 2x + 1) dx = 14$



## 复化 Simpson 公式

将求积区间  $n$  等分的基础上, 再将每个小区间  $[x_i, x_{i+1}]$  2 等份, 记内分点为  $x_{i+1/2}$   
 $x_i = a + ih, x_{i+1/2} = a + (i + 1/2)h, (i = 0, 1, 2, 3, \dots, n)$ 。对小区间  $[x_i, x_{i+1}]$  使用 Simpson 求积公式, 得到:

$$\begin{aligned} I &= \int_a^b f(x) dx = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \\ &\approx \sum_{i=0}^{n-1} \frac{x_{i+1} - x_i}{6} \left( f(x_i) + 4f(x_{i+\frac{1}{2}}) + f(x_{i+1}) \right) \\ &= \frac{b-a}{6n} \left( f(a) + 4 \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) + 2 \sum_{i=1}^{n-1} f(x_i) + f(b) \right) \\ &= S_n \end{aligned}$$





## 复化 Cotes 公式

将求积区间  $n$  等分的基础上，再将每个小区间  $[x_i, x_{i+1}]$  分为 4 等分，记内分点为  $x_{i+1/4}, x_{i+1/2}, x_{i+3/4}$ ，即：

$$x_i = a + ih$$

$$x_{i+\frac{1}{4}} = a + (i + \frac{1}{4})h$$

$$x_{i+\frac{1}{2}} = a + (i + \frac{1}{2})h$$

$$x_{i+\frac{3}{4}} = a + (i + \frac{3}{4})h$$

$$x_{i+1} = a + (i + 1)h \quad (i = 0, 1, 2, \dots, n)$$



## 复化 Cotes 公式 (续)

对小区间  $[x_i, x_{i+1}]$  使用 Cotes 求积公式, 则

$$\begin{aligned}
 I &= \int_a^b f(x) dx = \sum_{i=0}^{n-1} \int_{x_i}^{x_{i+1}} f(x) dx \\
 &\approx \sum_{i=0}^{n-1} \frac{x_{i+1} - x_i}{90} (7f(x_i) + 32f(x_{i+\frac{1}{4}}) \\
 &\quad + 12f(x_{i+\frac{1}{2}}) + 32f(x_{i+\frac{3}{4}}) + 7f(x_{i+1})) \\
 &= \frac{b-a}{90n} (7f(a) + 32 \sum_{i=0}^{n-1} f(x_{i+\frac{1}{4}}) + 12 \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) \\
 &\quad + 32 \sum_{i=0}^{n-1} f(x_{i+\frac{3}{4}}) + 7f(b)) = C_n
 \end{aligned}$$



## 复化求积例子

例题：用函数  $f(x) = \frac{\sin x}{x}$  的数据表计算积分  $I = \int_0^1 \frac{\sin x}{x} dx$ 。

$x$	$f(x)$
0	1.0000000
1/8	0.9973978
2/8	0.9896158
3/8	0.9767267
4/8	0.9588510
5/8	0.9361556
6/8	0.9088516
7/8	0.8771925
1	0.8414709



## 复化求积例子 (续)

解：此时  $n = 8$

$$\begin{aligned}
 T_8 &= \frac{b-a}{2n} (f(a) + 2 \sum_{i=0}^{n-1} f(x_i) + f(b)) \\
 &= \frac{1-0}{2 \cdot 8} (f(a) + 2 \sum_{i=1}^{8-1} f(x_i) + f(b)) \\
 &= 0.945690806 \\
 S_4 &= \frac{b-a}{6n} \left( f(a) + 4 \sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) + 2 \sum_{i=1}^{n-1} f(x_i) + f(b) \right) \\
 &= \frac{1-0}{6 \cdot 4} (f(a) + 4 \sum_{i=0}^{4-1} f(x_{i+\frac{1}{2}}) + 2 \sum_{i=1}^{4-1} f(x_i) + f(b)) \\
 &= 0.946083254
 \end{aligned}$$



## 复化求积例子 (续)

同理，应用复合 Cotes 公式可求得  $C_2$  值如下：

$$C_2 = 0.94608307$$

该积分的“精确值”为

$$I = \int_0^1 \frac{\sin x}{x} dx = 0.9460831$$

对上述三个结果进行比较可以看出， $T_8$  精度最低、 $S_4$  精度次高， $C_2$  精度最高。上述积分都需要提供 9 个节点上的函数值，工作量基本相同，然而精度却相差很大，同积分的精确值比较，复化梯形法的结果只有 2 位有效数值，而复化 Simpson 法和 Cotes 的结果有 6 位有效数值。



## 梯形法加速

复化梯形法的算法简单但精度低。能否加工梯形值来提高精度呢？考查二分前后的梯形公式：

$$\begin{aligned} T_1 &= \frac{b-a}{2}(f(a) + f(b)) \\ T_2 &= \frac{1}{2}T_1 + \frac{b-a}{2}f\left(\frac{a+b}{2}\right) \\ &= \frac{b-a}{4}\left(f(a) + 2f\left(\frac{a+b}{2}\right) + f(b)\right) \\ S_1 &= \frac{b-a}{6}\left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right) \end{aligned}$$

因此，我们得到

$$S_1 = \frac{4}{3}T_2 - \frac{1}{3}T_1$$



## Romberg 算法

推广到复合求积公式，有：

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

同理，我们可以得到：

$$C_n = \frac{16}{15}S_{2n} - \frac{1}{15}S_n$$

$$R_n = \frac{64}{63}C_{2n} - \frac{1}{63}C_n$$

此方法称为 Romberg 公式，它是由复化梯形公式、复化 Simpson 公式依次递推求得的，松弛因子是  $\omega = 1/63$ 。它可将粗糙的梯形公式逐步加工成高精度的 Romberg 公式，称这种算法为 Romberg 算法。综合表示如下：



## Romberg 算法 (续)

$$\begin{cases} T_1 = \frac{b-a}{2}[f(a) + f(b)] \\ T_{2n} = \frac{1}{2}T_n + \frac{h}{2}\sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) \\ S_n = \frac{4}{3}T_{2n} - \frac{1}{3}T_n \\ C_n = \frac{16}{15}S_{2n} - \frac{1}{15}S_n \\ R_n = \frac{64}{63}C_{2n} - \frac{1}{63}C_n \end{cases}$$

$n = 2^k, k = 0, 1, 2, 3, \dots$ , 以上整个过程称为 Romberg 算法。

$k$	$T_{2^k}$	$S_{2^{k-1}}$	$C_{2^{k-2}}$	$R_{2^{k-3}}$
0	$T_1$			
1	$T_2$	$S_1$		
2	$T_4$	$S_2$	$C_1$	
3	$T_8$	$S_4$	$C_2$	$R_1$
4	$T_{16}$	$S_8$	$C_4$	$R_2$





## Romberg 算法例子

例题：利用 Romberg 算法计算积分

$$I = \int_0^1 \frac{\sin x}{x} dx$$

解：计算求得

$k$	$T_{2^k}$	$S_{2^{k-1}}$	$C_{2^{k-2}}$	$R_{2^{k-3}}$
0	0.9207355			
1	0.9397933	0.9461459		
2	0.9445135	0.9460869	0.9460830	
3	0.9456909	0.9460834	0.9460831	0.9460831

得到  $R_1 = 0.9460831$ 。



## Gauss 公式

Newton-Cotes 公式在构造时，限定积分区间的等分点作为求积节点，这样做在简化处理的同时也限制了代数精度。如果求积节点也可自由选择，即机械求积公式：

$$\int_a^b f(x) dx \approx (b-a) \sum_{i=0}^n \lambda_i f(x_i)$$

中的  $\lambda_i, x_i$  共  $2n+2$  个均为待定参数，适当选取这些参数可以使公式具有  $2n+1$  次代数精度。这种高精度的求积公式称为 Gauss 公式。



## 积分区间变换

不失一般性，把积分区间取成  $[-1, 1]$ ，这是因为利用变换：

$$x = \frac{a+b}{2} + \frac{b-a}{2}t$$

总可将区间  $[a, b]$  变换成  $[-1, 1]$ ，而积分变为：

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

其中

$$g(t) = f\left(\frac{a+b}{2} + \frac{b-a}{2}t\right)$$



## 一点 Gauss 公式

对于一点 Gauss 公式 ( $n = 0$ , 具有一次代数精度)

$$\int_{-1}^1 f(x) dx \approx 2 \sum_{i=0}^0 \lambda_i f(x_i) = 2\lambda_0 f(x_0)$$

因此当  $f(x) = 1, f(x) = x$  时上式精确成立, 有:

$$\begin{cases} 2\lambda_0 = 2 \\ 2\lambda_0 x_0 = 0 \end{cases}$$

解得:  $\lambda_0 = 1, x_0 = 0$ , 因此得到:

$$\int_{-1}^1 f(x) dx \approx G_1 = 2f(0)$$



## 两点 Gauss 公式

对于两点 Gauss 公式 ( $n = 1$ , 具有三次代数精度)

$$\int_{-1}^1 f(x) dx \approx 2 \sum_{i=0}^1 \lambda_i f(x_i) = 2(\lambda_0 f(x_0) + \lambda_1 f(x_1))$$

因此当  $f(x) = 1, x, x^2, x^3$  时上式精确成立, 有:

$$\begin{cases} \lambda_0 + \lambda_1 = 1 \\ \lambda_0 x_0 + \lambda_1 x_1 = 0 \\ \lambda_0 x_0^2 + \lambda_1 x_1^2 = \frac{1}{3} \\ \lambda_0 x_0^3 + \lambda_1 x_1^3 = 0 \end{cases}$$

解得:  $\lambda_0 = \lambda_1 = \frac{1}{2}, x_1 = -x_0 = \frac{1}{\sqrt{3}}$ , 因此得到:

$$\int_{-1}^1 f(x) dx \approx G_2 = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$



## 三点 Gauss 公式

对于三点 Gauss 公式 ( $n=2$ , 具有五次代数精度)

$$\int_{-1}^1 f(x) dx \approx 2 \sum_{i=0}^2 \lambda_i f(x_i) = 2(\lambda_0 f(x_0) + \lambda_1 f(x_1) + \lambda_2 f(x_2))$$

因此当  $f(x) = 1, x, x^2, x^3, x^4, x^5$  时上式精确成立, 有:

$$\begin{cases} \lambda_0 + \lambda_1 + \lambda_2 = 1 \\ \lambda_0 x_0 + \lambda_1 x_1 + \lambda_2 x_2 = 0 \\ \lambda_0 x_0^2 + \lambda_1 x_1^2 + \lambda_2 x_2^2 = \frac{1}{3} \\ \lambda_0 x_0^3 + \lambda_1 x_1^3 + \lambda_2 x_2^3 = 0 \\ \lambda_0 x_0^4 + \lambda_1 x_1^4 + \lambda_2 x_2^4 = \frac{1}{5} \\ \lambda_0 x_0^5 + \lambda_1 x_1^5 + \lambda_2 x_2^5 = 0 \end{cases}$$



## 三点 Gauss 公式 (续)

解得:

$$\begin{cases} \lambda_0 = \frac{5}{18} \\ \lambda_1 = \frac{4}{9} \\ \lambda_2 = \frac{5}{18} \\ x_0 = -\sqrt{\frac{3}{5}} \\ x_1 = 0 \\ x_2 = \sqrt{\frac{3}{5}} \end{cases}$$

因此得到:

$$\int_{-1}^1 f(x) dx \approx G_3 = \frac{5}{9}f\left(-\sqrt{\frac{3}{5}}\right) + \frac{8}{9}f(0) + \frac{5}{9}f\left(\sqrt{\frac{3}{5}}\right)$$

# 一般高斯求积公式

取积分区间为  $[-1, 1]$ ，考察如下形式的求积公式：

$$\int_{-1}^1 f(x) dx \approx 2 \sum_{i=0}^n \lambda_i f(x_i)$$

使其成为 Gauss 型的，具有  $2n+1$  次代数精度。即满足：

$$\left\{ \begin{array}{l} \int_{-1}^1 1 dx = 2 \sum_{i=0}^n \lambda_i \cdot 1 \\ \int_{-1}^1 x dx = 2 \sum_{i=0}^n \lambda_i \cdot x_i \\ \int_{-1}^1 x^2 dx = 2 \sum_{i=0}^n \lambda_i \cdot x_i^2 \\ \dots \\ \int_{-1}^1 x^{2n+1} dx = 2 \sum_{i=0}^n \lambda_i \cdot x_i^{2n+1} \end{array} \right.$$



## 一般高斯求积公式 (续)

对上述积分计算可得到  $2n + 2$  个方程:

$$\left\{ \begin{array}{l} \sum_{i=0}^n \lambda_i = 1 \\ \sum_{i=0}^n \lambda_i x_i = 0 \\ \sum_{i=0}^n \lambda_i x_i^2 = \frac{1}{3} \\ \dots \\ \sum_{i=0}^n \lambda_i x_i^k = \frac{1+(-1)^k}{2(k+1)} \\ \dots \\ \int_{-1}^1 \lambda_i x_i^{2n+1} dx = 0 \end{array} \right.$$

因此, 可解出  $\lambda_i$  和  $x_i$  共  $2n + 2$  个未知数。



# 一般积分区间的高斯公式

一般情况下，对于积分区间  $[a, b]$ ，则可由

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

得：

$$\int_{-1}^1 f(x) dx \approx G_1 = 2f(0)$$

$$\int_a^b f(x) dx \approx G_1 = (b-a)f\left(\frac{a+b}{2}\right)$$

$$\int_{-1}^1 f(x) dx \approx G_2 = f\left(-\frac{1}{\sqrt{3}}\right) + f\left(\frac{1}{\sqrt{3}}\right)$$

$$\int_a^b f(x) dx \approx G_2 = \frac{b-a}{2} \left( f\left(\frac{a+b}{2} - \frac{b-a}{2\sqrt{3}}\right) + f\left(\frac{a+b}{2} + \frac{b-a}{2\sqrt{3}}\right) \right)$$



## 一般积分区间的高斯公式 (续)

$$\int_{-1}^1 f(x) dx \approx G_3 = \frac{5}{9}f(-\sqrt{\frac{3}{5}}) + \frac{8}{9}f(0) + \frac{5}{9}f(\sqrt{\frac{3}{5}})$$

$$\int_a^b f(x) dx \approx G_3 = \frac{b-a}{2} \left( \frac{5}{9}f\left(\frac{a+b}{2} - \sqrt{\frac{3}{5}} \frac{b-a}{2}\right) + \frac{8}{9}f\left(\frac{a+b}{2}\right) + \frac{5}{9}f\left(\frac{a+b}{2} + \sqrt{\frac{3}{5}} \frac{b-a}{2}\right) \right)$$

例题：试构造如下形式的 Gauss 型求积公式：

$$\int_0^1 \sqrt{x} f(x) dx \approx A_0 f(x_0) + A_1 f(x_1)$$

## 高斯公式例子

解：此时  $n = 1$ ，另它对于  $f(x) = 1, x, x^2, x^3$  精确成立，得到如下方程组：

$$\begin{cases} A_0 + A_1 = \frac{2}{3} \\ A_0 x_0 + A_1 x_1 = \frac{2}{5} \\ A_0 x_0^2 + A_1 x_1^2 = \frac{2}{7} \\ A_0 x_0^3 + A_1 x_1^3 = \frac{2}{9} \end{cases}$$

求得：

$$\begin{aligned} x_0 &= 0.821163, x_1 = 0.289949, \\ A_0 &= 0.389111, A_1 = 0.277556 \end{aligned}$$

因此，求得的高斯积分公式为：

$$\int_0^1 \sqrt{x} f(x) dx \approx 0.389111 \cdot f(0.821163) + 0.277556 \cdot f(0.289949)$$



## 本章小结

- 机械求积;
- Newton-Cotes 公式;
- 复化求积公式;
- 龙贝格算法;
- Gauss 公式;



## 练习题

- ① 编程实现 Newton-Cotes 求积算法。
- ② 编程实现 Simpson 复化求积公式。
- ③ 编程实现 Gauss 求积算法。



# 谢谢!

AUTHOR: Cheng Yong

ADDRESS: Dept. of Computer  
Beijing University of Chemical Technology  
Beijing, 100029, China

EMAIL: [buctcourse@163.com](mailto:buctcourse@163.com)