

# 高阶高斯积分节点的高精度数值计算

张庆礼<sup>1</sup>, 王晓梅<sup>2</sup>, 殷绍唐<sup>1</sup>, 江海河<sup>3</sup>

(1. 中国科学院安徽光学精密机械研究所, 合肥 230031; 2. 中国科学院环境光学与技术重点实验室, 合肥 230031;  
3. 中国科学院合肥物质科学研究院, 合肥 230031)

[摘要] 在工程数值计算、X射线衍射线形分析、光谱学等领域常使用高斯数值积分, 高斯积分的节点及权重因子是数值积分的必须数据。研究了高次勒让德、拉盖尔和厄米多项式的零点, 即高斯-勒让德、高斯-拉盖尔、高斯-厄米积分的节点的计算方法, 给出了一种有效的高精度数值算法——搜索迭代方法(scan-iteration method, SIM)。根据勒让德、拉盖尔、厄米多项式的特点, 对拉盖尔多项式、厄米多项式的定义稍做变化后, 获得了计算多项式值的稳定递推关系。求它们的根时, 先在一定范围内以一定的步长搜索根所在的区间, 获得所有根的各自区间范围后, 再通过常用的迭代方法如割线法、二分法进行求解。数值实验表明, 这种方法是行之有效的, 可获得高次勒让德、拉盖尔、厄米多项式的全部高精度根值。

[关键词] 高斯积分; 勒让德多项式; 拉盖尔多项式; 厄米多项式; 求根

[中图分类号] O174.6; O241 [文献标识码] A [文章编号] 1009-1742(2008)02-0035-06

## 1 前言

高斯数值积分是一种有效的高精度数值计算方法。在工程数值计算、固体物理学、光电子学等领域常用来对难以获得解析解的积分进行数值计算。例如, 在工程电磁场计算中常用的贝塞尔函数<sup>[1]</sup>, 如果采用其级数解来计算, 需考虑很复杂的递推稳定性问题; 反之, 若从它的积分形式出发来计算, 采用数个节点的高斯积分, 即可获得满意的数值精度。在X射线衍射中的线形分析和光谱学中, Voigt函数能较好地描述X射线衍射峰和光谱峰的线形, 但该函数没有解析的积分解; 固体物理学中热容表达式中的德拜函数也无解析的积分解。用高斯数值积分则可方便地计算这些积分。

$n$ 个节点数值积分可表示为

$$\int_a^b \rho(x) f(x) dx \approx \sum_{i=1}^n w_i f(x_i) \quad (1)$$

式中  $\rho(x)$  是在积分区间  $(a, b)$  上的正权重函数,  $x_i$  和  $w_i$  是与函数  $f(x)$  无关的节点和权重因子。合理地选择不同的节点  $x_1, x_2, \dots, x_n$ , 使数值积分公

式(1)对次数  $\leq 2n-1$  的多项式精确成立, 即为高斯积分公式, 它的代数精度为  $2n-1$ 。在所有的  $n$  个节点积分公式中, 高斯积分有最高的代数精度。

可以证明, 积分公式(1)为高斯积分的充要条件是积分节点为正交多项式的根<sup>[2]</sup>。不同的高斯积分公式如高斯-勒让德、高斯-拉盖尔、高斯-厄米、高斯-切比雪夫积分的积分区间和正权重函数不同<sup>[3]</sup>。高斯-勒让德、高斯-拉盖尔、高斯-厄米、高斯-切比雪夫的正权重函数和积分区间分别为 1 和  $[-1, 1]$ ,  $e^{-x}$  和  $[0, \infty)$ ,  $e^{-x^2}$  和  $(-\infty, +\infty)$ ,  $(1-x^2)^{-1/2}$  和  $[-1, 1]$ 。

很多文献可以查到高斯积分公式的节点  $x_i$  和权重因子  $w_i$  值<sup>[4~6]</sup>。对于非振荡的被积函数, 只需要用数个节点来计算, 便可获得良好的积分精度, 更高的精度可用更高次的公式来获得。现能查到的高斯-勒让德、高斯-拉盖尔、高斯-厄米积分的最高次节点和权重因子分别为 96, 15 和 20<sup>[4]</sup>。

为了获得更精确的积分值, 需要有更高次的积分节点和权重因子。由于计算机的精度有限, 其舍入误差往往会在多项式的次数很高或自变量很大的

[收稿日期] 2006-08-14; 修回日期 2007-02-14

[基金项目] 国家自然科学基金资助项目(60478025, 50472104)

[作者简介] 张庆礼(1973-), 男, 云南会泽县人, 博士, 中国科学院安徽光学精密机械研究所副研究员, 主要从事晶体物理研究

情况下,给多项式值的计算带来很大误差,导致高阶积分节点及权重因子的计算很困难<sup>[7]</sup>。计算正交多项式的节点可以用牛顿法或魏尔斯特拉斯法<sup>[8]</sup>,但计算表明,对于高阶正交多项式,要获得全部的根很困难。G. V. Milovanovic等给出了一个计算正交多项式根的算法<sup>[9]</sup>,但需给出足够精确的初始值。对于  $n \rightarrow \infty$  的勒让德正交多项式  $P_n(x)$ ,也可用渐近展开以获得其零点,  $P_n(x)$  的第  $k$  个零点渐近展开通常有两类,第一类展开需用到贝塞尔函数  $J_0(x)$  的第  $k$  个根,第二类则包含了三角函数<sup>[10]</sup>。数值实验表明,对于给定的  $n$ ,这些近似展开的精度在很大程度上依赖于  $k$  的取值范围 ( $1 \leq k \leq [n/2]$ )。F. G. Lether 等发展了勒让德多项式  $P_n(x)$  的第  $n$  个正零点的极小化最大三角近似<sup>[10]</sup>,可获得  $n \geq 2$  时 4.2 个有效位数,用于估计勒让德多项式根的初始值,以便迭代求解。P. N. Swartrauber 比较了计算高斯-勒让德积分的节点和权重因子的三种方法:本征值方法, Yakimiw 方法和 Fourier-Newton 法<sup>[11]</sup>。本征值方法的节点和权重误差随  $n$  的增长分别为  $O(n)$  和  $O(n^2)$ ; Yakimiw 方法的节点和权重误差则为  $O(1)$ ,逐点相关误差是一致的,即使在积分端点处节点趋近于 0 时,所有权重  $r$  仍具有相同的有效位数,但在  $x = \pm 1$  处,由于机器精度的原因,高斯-勒让德节点的二次群聚使得根产生合并甚至交换; Fourier-Newton 法基于直接计算变换点  $\theta_i = \arccos x_i$ ,不发生群聚。E. Babolian 等把积分限  $a, b$  看作两个变量,以获得较精确的单项式  $f(x) = x^j$  的高斯积分公式<sup>[12]</sup>。然而,此方法的应用范围很有限。

为了获得高次勒让德多项式、拉盖尔多项式和厄米多项式的根,以用于高斯积分,笔者也尝试了数值方法中的多种非线性求根方法,如二分法、割线法、抛物线法(穆勒法)、 $x = g(x)$  的简单迭代法、牛顿法、牛顿下山法、林士谔-赵访熊法(劈因子法),这些方法在求解高次正交多项式时,都存在如下问题:无法找到全部根,迭代精度进行到一定程度便无法进行迭代,不能获得所需的迭代精度,或者迭代过程无法收敛。

笔者提出了高精度高次勒让德、拉盖尔和厄米多项式的一种求根方法:先在不影响它们的根的情况下,对其定义稍作变化,然后在一定范围内尝试以不同步长来搜索其全部根所在的区间范围,再以常用的根迭代办法来求根。实验表明这种方法——笔者称作搜索迭代法(SIM, scan-iteration method)是非

常有效的,能正确获得高次正交多项式的全部根,计算速度很快,精度高。

## 2 高次多项式的计算

### 2.1 勒让德多项式

勒让德多项式定义为

$$\left. \begin{aligned} P_0 &= 1 \\ P_n(x) &= \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n \end{aligned} \right\} \quad (2)$$

式中  $-1 \leq x \leq 1$ 。

勒让德多项式有递推关系:

$$P_n(x) = \frac{2n-1}{n} x P_{n-1}(x) - \frac{n-1}{n} P_{n-2}(x) \quad (3)$$

假设仅在第 1 和第 2 个计算步骤中存在舍入误差,由于该误差传播的影响,  $P_n(x)$ ,  $P_{n-1}(x)$ ,  $P_{n-2}(x)$  分别变为  $P_n(x)$ ,  $P_{n-1}(x)$ ,  $P_{n-2}(x)$ , 满足公式(3)。让  $\epsilon_n = P_n(z) - P_n(z)$ , 则可得到方程

$$\epsilon_n - b_n x \epsilon_{n-1} + c_n \epsilon_{n-2} = 0 \quad (4)$$

式中  $b_n = (2n-1)/n$ ,  $c_n = -(n-1)/n$ 。称式(4)为式(3)的误差方程。

当  $n$  很大时,  $b_n \rightarrow 2$ ,  $c_n \rightarrow 1$ , 式(4)近似为

$$\epsilon_n - 2x \epsilon_{n-1} + \epsilon_{n-2} = 0 \quad (5)$$

式(5)为线性齐次差分方程,其特征方程为

$$r^2 - 2xr + 1 = 0 \quad (6)$$

式(6)的根为  $r_1 = x + i(1-x^2)^{1/2}$ ,  $r_2 = x - i(1-x^2)^{1/2}$ 。

此处  $i$  为虚数单位。  $|r_1| = 1$ ,  $|r_2| = 1$ 。根据递推稳定性定理<sup>[1]</sup>,可知递推关系式(3)是稳定的。由  $P_0 = 1$ ,  $P_1 = x$  和式(3),勒让德多项式的值能在可控的误差下做向前递推计算。

高斯-勒让德积分的权重因子为

$$w_i = \frac{2}{(1-x_i^2)(P_n'(x_i))} \quad (7)$$

式中  $P_n'(x)$  是勒让德多项式  $P_n$  的导数,其计算式为

$$P_n'(x) = (nP_{n-1}(x) - n x P_n(x))/(1-x^2) \quad (8)$$

### 2.2 拉盖尔多项式

在大多数文献中,拉盖尔多项式定义为

$$L_n(x) = e^x \frac{d^n}{dx^n} (e^{-x} x^n) = n! \sum_{k=0}^n \frac{(-1)^k n! x^k}{(k!)^2 (n-k)!} \quad (9)$$

式中  $x \in [0, \infty)$ 。可以看到  $L_n(x) \propto (n!)^2$ , 它的值随着  $n$  值的增加迅速增大,致使计算机发生计算溢出错误。现在看一个具体算例,拉盖尔多项式的系

数  $a_k$  可由如下递推关系计算:

$$\left. \begin{aligned} a_{k+1} &= -\frac{n-k}{(k+1)^2} a_k \quad (n-k \leq (k+1)^2) \\ a_k &= -\frac{(k+1)^2}{n-k} a_{k+1} \quad (n-k > (k+1)^2) \\ a_0 &= n! \\ a_n &= (-1)^n \end{aligned} \right\} (10)$$

计算得  $a_k (k=0, \dots, n), x^k$ , 把所有的  $a_k \times x^k$  的值相加, 记为算法 1; 再用较精确的算法 (记为算法 2) 来计算拉盖尔多项式<sup>[6]</sup>:

$$L_n(x) = \prod_{k=0}^n a_0 \left[ 1 - \frac{n-k}{(k+1)^2} x \right] = a_0 \left\{ 1 - \frac{n}{1^2} x \left[ 1 - \frac{n-1}{2^2} x \left[ 1 - \frac{n-2}{3^2} x (1 - \dots) \right] \right] \right\} \quad (11)$$

表 1 列出了幂次为 5, 100 的拉盖尔多项式用双精度 (其大小为 10 字节, 数值范围为  $3.6 \times 10^{-4.951} \sim 1.1 \times 10^{4.932}$ , 有效小数点为 9~20 位) 的计算结果。

从表 1 可以看到, 对于  $L_5(x)$ , 算法 1 和算法 2 给出了一致的计算结果。而对于  $L_{100}(x)$ , 在小变量

表 1 两种算法计算  $L_{100}(x)$  和  $dL_5(x)$  的结果比较

Table 1 Comparison of computational results between  $L_{100}(x)$  and  $L_5(x)$  obtained by two algorithms

x	$L_5(x)$		$L_{100}(x)$	
	算法 1	算法 2	算法 1	算法 2
0.1	65.802 49	65.802 49	9.287 839 773 571 34 $\times 10^{157}$	2.275 758 258 685 12 $\times 10^{157}$
1	-56	-56	-2.432 280 202 501 34 $\times 10^{164}$	2.526 022 607 702 77 $\times 10^{157}$
10	4 120	4 120	-4.025 728 841 128 17 $\times 10^{172}$	2.117 376 968 894 17 $\times 10^{165}$
100	-7 694 059 880	-7 694 059 880	-2.629 059 773 497 1 $\times 10^{208}$	1.238 012 237 254 73 $\times 10^{208}$

$x=0.1$  时尚给出相同数量级的计算结果, 而当  $x=1, 10, 100$  时, 两种计算结果的差别是巨大的, 连符号都相反, 因此计算结果不可信。

式(9)定义的拉盖尔多项式的递推关系为

$$L_{n+1}(x) = (2n+1-x)L_n(x) - n^2 L_{n-1}(x) \quad (12)$$

此递推关系是不稳定的。如果采用式(12)来计算拉盖尔多项式  $L_n(x)$ , 舍入误差将会随着递推的进行而不断被放大, 致使难以获得多项式的精确值。

笔者要获得的是拉盖尔多项式的根, 为此可对拉盖尔多项式稍做修改, 将拉盖尔多项式定义为

$$l_n(x) = \frac{L_n(x)}{n!n!} \quad (13)$$

方程(13)与方程(9)有相同的根, 但它的值不再正比于  $(n!)^2$ 。相应地, 式(13)的递推关系为

$$l_n(x) = \frac{2n-1-x}{n} l_{n-1}(x) - \frac{1}{n} l_{n-2}(x) \quad (14)$$

式(14)的误差方程为

$$\epsilon_n - \frac{2n-1-x}{n} \epsilon_{n-1} + \frac{n-1}{n} \epsilon_{n-2} = 0 \quad (15)$$

当  $n$  是很大的自变量时, 对于一定的  $x$  值, 式(15)变为

$$\epsilon_n(x) \rightarrow 0 \quad (16)$$

这表明向前递推关系式(14)是稳定的。由

$$\left. \begin{aligned} l_0(x) &= 1 \\ l_1(x) &= 1-x \end{aligned} \right\} \quad (17)$$

则能在可控制的误差下由递推关系式(14)来计算拉盖尔多项式  $l_n(x)$ 。

高斯—拉盖尔的积分重因子计算式为

$$w_i = \frac{1}{x_i n! [l'_n(x_i)]^2} \quad (18)$$

### 2.3 厄米多项式

厄米多项式定义为

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} (e^{-x^2}) = n! \sum_{k=0}^{[n/2]} \frac{(-1)^k x^k}{k! (n-2k)!} (2x)^{n-2k} \quad (19)$$

式中  $x \in (-\infty, +\infty)$ ;

$$[n/2] = \begin{cases} n/2, & n \text{ 为偶数} \\ (n-1)/2, & n \text{ 为奇数} \end{cases}$$

其关系为  $H_n(x) = (-1)^n H(x)$ 。

如果  $x_i$  是  $H_n(x)$  的根,  $-x_i$  也是它的根。当求方程(19)的根时, 可以先在区间  $[0, +\infty)$  上求解  $[n/2]$  个根, 然后借此关系获得它所有的根。

厄米多项式(19)的递推关系为

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x) \quad (20)$$

从式(19)可以看出, 厄米多项式的值正比于  $n!$ , 这使得在计算它的值时容易发生计算溢出。为了避免计算溢出, 把厄米多项式的定义稍做变换, 记为  $h_n$ ,

$$h_n(x) = \frac{1}{2^n n!} H_n(x) \quad (21)$$

由式(20)、式(21)可得  $h_n$  的递推关系为

$$h_{n+1}(x) = \frac{x}{n+1} h_n(x) - \frac{1}{2(n+1)} h_{n-1}(x) \quad (22)$$

$H_n$  和  $h_n$  具有相同的根。方程(22)的误差方程为

$$\epsilon_{n+1} - \frac{x}{n+1} \epsilon_n + \frac{1}{2(n+1)} \epsilon_{n-1} = 0 \quad (23)$$

当  $n$  值很大时,方程(23)变为

$$\epsilon_{n+1} \rightarrow 0 \quad (24)$$

这表明式(22)向前递推计算是稳定的。根据式(22),得

$$\left. \begin{aligned} h_0(x) &= 1 \\ h_1(x) &= x \end{aligned} \right\} \quad (25)$$

可在可控误差下计算  $h_n(x)$  的值。

高斯-厄米积分权重因子的计算式为

$$w_i = \frac{2\sqrt{\pi}}{2^n n [h_n'(x_i)]^2} \quad (26)$$

### 3 由搜索迭代法求高次多项式的根

现在来讨论上述多项式的一种求根方法。用这种方法可以很容易获得高精度的高次勒让德、拉盖尔和厄米多项式的根。

上述  $n$  次正交多项式在其定义区间有  $n$  个互异的根。 $n$  次勒让德多项式和厄米多项式分别在  $(0,1)$  和  $(0,+\infty)$  区间有  $[n/2]$  个根  $x_i$ , 剩余的根则为  $-x_i (i=0,1,2,\dots,[n/2])$  和  $0$  (当  $n$  是奇数时);  $n$  阶拉盖尔多项式则在  $(0,+\infty)$  区间有  $n$  个根。对于任一有限值  $n$ , 这些多项式的根都在一有限的范围内。例如当  $n=10,100,1\,000$  时,拉盖尔多项式的根全部分别位于  $(0.1,30)$ 、 $(0.01,374.99)$  和  $(0.001,3\,943.248)$  区间,厄米多项式的根则全部分别位于  $(0.3,3.5)$ 、 $(0.1,13.5)$  和  $(0.03,44.21)$  区间。

另外,对于任何根  $x_i$  而言,多项式的值在根两侧  $x, y (x < x_i < y)$  处的正负号相反。

根据多项式的这些特点,对  $n$  次多项式,可以在一定的区间范围  $(0, R)$  确定出  $[n/2]$  个根所在的区间范围  $(r_{x,i}, r_{y,i}) (i=0,1,2,\dots,[n/2])$ , 对于拉盖尔多项式,  $[n/2]$  应为  $n$ 。由于高次多项式的根很多,用多项式曲线来确定各个根的区间的方法不实用,因此采用了自动搜索方法,如图1所示(图中“RootNo”用于存储根数目,“MaxNo”用于存储程序所允许的最大搜索次数,“ $P(x_j)$ ”表示变量为“ $x_j$ ”的多项式的值,“ $\text{Sign}(P(x_j))$ ”表示  $P(x_j)$  的符号。对于勒让德多项式,即  $R \equiv 1$ )。先假设一个相对较小

的范围  $R$  和搜索步进  $S$  进行搜索。如果  $[n/2]$  个根区间  $(r_{x,i}, r_{y,i}) (i=0,1,2,\dots,[n/2])$  被搜索到,则搜索停止;否则再减小步进  $S$  重新进行搜索。如此反复,直到  $[n/2]$  个根区间全部搜索到或者搜索次数  $\{R/S\}$  大于最大的允许搜索次数  $(\{R/S\}$  表示取  $R/S$  值的整数部分)时停止搜索。

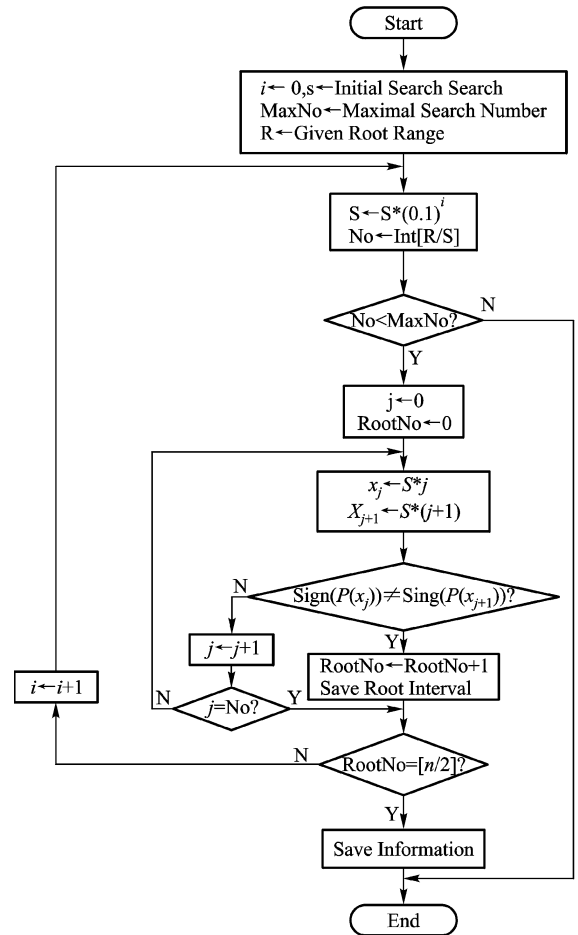


图1 搜索根区间流程图

Fig.1 Flow diagram of searching root

可以看出,只要设定的  $(0, R)$  区间范围包含了所有  $n$  次正交多项式的根,则只需步进  $S$  的值足够小,根的区间总可以被全部找到。

找到多项式的  $[n/2]$  或  $n$  个根区间后,再用通常的迭代方法如割线法、牛顿法、二分法进行求解所需精度的根。笔者称这种求根方法为搜索迭代求解法(SIM)。

以下讨论以 SIM 方法求解正高精度正交多项式的根中出现的问题。

割线迭代法的格式为

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k) \quad (27)$$

在多项式求根迭代过程中,当迭代进行到一定程度后,会出现  $f(x_k) \approx f(x_{k-1})$  的情况,此时  $f(x_k) - f(x_{k-1}) \rightarrow 0$ ,容易导致计算机的计算溢出,迭代过程无法进行。此时,可尝试用牛顿迭代法以避免出现此种情形,即

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad (28)$$

可以看到,式(28)中  $x_k$  的收敛由  $f(x_k)/f'(x_k)$  决定,当  $x_k$  达到某一精度时,由于有  $f(x_k)/f'(x_k) \approx f(x_{k+1})/f'(x_{k+1})$  的近似关系,迭代收敛速度将显著下降。这种情况在求拉盖尔多项式的根时常出现。例如对于求  $l_{1000}$  的根时中间的一个迭代系列:

$x$	$l_{1000}(x)$	$l'_{1000}(x)$	$l_{1000}(x)/l'_{1000}(x)$
0.001 445 074 067 548 58	0.008 273 208 881 908 62	-0.041 546 330 816 725 4	-5.021 791 593 776 453 53
0.001 445 074 067 548 59	0.008 273 208 881 908 62	-0.041 546 330 816 721 9	-5.021 791 593 776 030 48

在这种情形下,要获得更高精度的根非常困难。此时,采用二分法(bisection method)可获得令人满意的迭代精度。用牛顿法和二分法来求解拉盖尔多项

式的根的一个比较例子见表2。可以看出,在此情形下,二分法的收敛速度显著快于牛顿法。

表2 用牛顿法和二分法求  $l_{1000}(x)$  的根时两种方法的比较

Table 2 Comparison of Newton and bisection method to extract roots of  $l_{1000}(x)$

牛顿法			
$k$	$x_{k+1}$	$x_k$	$x_{k+1} - x_k$
500	0.001 445 074 067 548 59	0.001 445 074 067 550 45	-1.868 080 343 192 52E-1
5 000	0.001 445 074 067 550 99	0.001 445 074 067 549 77	1.214 740 114 052 76E-15
50 000	0.001 445 074 067 547 59	0.001 445 074 067 547 69	-1.016 981 637 791 4E-16
二分法			
$k$	$x_{k+1}$	$x_k$	$x_{k+1} - x_k$
5	0.001 453 125	0.001 468 75	-1.562 5E-5
50	0.001 445 074 067 541 51	0.001 445 074 067 541 51	4.440 570 225 980 67E-19
60	0.001 445 074 067 541 51	0.001 445 074 067 541 51	-6.882 142 696 441 19E-21

如果  $f'(x_k) \rightarrow 0$ ,可采用割线法迭代格式(25)进行迭代求根,以避免由于  $f(x_k)$  被  $f'(x_k)$  除而导致的计算溢出。

数值实验表明,用搜索一割线法迭代求解高斯一勒让德、高斯一厄米积分的节点、用搜索一二分法迭代求解的高斯一拉盖尔的积分节点,可获得满意的迭代精度。

基于以上搜索迭代算法,编写了一个计算程序 PolynominalRoot 对勒让德多项式、拉盖尔多项式、厄米多项式求根。对于  $n=80$  和  $96$  的勒让德多项式、 $n=12$  和  $15$  的拉盖尔多项式、 $n=20$  的厄米多项式的求根和权重因子的结果与文献[4]给出的结果完全一致。只要给出正确的寻根范围  $R$ ,计算的速度也很快。例如,对于  $n < 100$  的多项式的求根,在 2 392 MHz CPU、256 M 内存的 IBM 计算机上求解时,

所需时间不超过 1 min;  $n=1000$  时,设定勒让德多项式、拉盖尔多项式、厄米多项式的根范围  $(0, R)$  为  $(0, 1)$ ,  $(0, 3944)$  和  $(0, 50)$ ,迭代精度为  $10^{-20}$ ,求根所花费的时间  $< 10$  min。这些结果表明,搜索迭代求解算法是一种有效的正交多项式求根方法。

#### 4 结语

研究了高次勒让德多项式、拉盖尔多项式和厄米多项式的高精度根的数值求解方法。先对拉盖尔多项式、厄米多项式的定义稍做变化后,获得了计算多项式值的稳定递推关系。通过搜索一迭代算法,先在一定的根范围内搜索到多项式所有的根区间,再以常用的迭代求根方法如割线法、二分法或牛顿法等进行求解,可获得所需精度的根。在获得高精度的正交多项式的根后,可给出高次高斯积分的积

分节点和积分权重因子。

#### 参考文献

- [ 1 ] 雷银照. 时谐电磁场解析方法[M]. 北京: 科学出版社, 2000. 253, 266
- [ 2 ] Kincaid D, Cheney W. Numerical Analysis: Mathematics of Scientific Computing [M]. Beijing: China Machine Press, 2003. 493
- [ 3 ] 易大义, 陈道琦. 数值分析引论 [M]. 杭州: 浙江大学出版社, 1998. 176~178
- [ 4 ] Abramowitz M, Stegun I A. Handbook of Mathematical Functions With Formulas Graphs, and Mathematical Tables [M]. The Fifth Printing, Washington: U.S. Government Printing Office, 1964. 919~924
- [ 5 ] 数学手册编写组. 数学手册[M]. 北京: 高等教育出版社, 1979. 297~300
- [ 6 ] 图马 J J, 沃尔什 R A. 工程数学手册[M]. 欧阳芳锐, 张玉平译, 北京: 科学出版社, 2002. 282, 424.
- [ 7 ] Reckenwald G. Numerical Methods With Matlab: Implementation and Application [M]. Translated by Wu Weiguo, Wan Qun, Zhang Hui, et al. Beijing: China Machine Press, 2004. 439
- [ 8 ] Laurie D P. Computation of Gauss-type quadrature formulas [J]. Computational and Applied Mathematics, 2001, 127: 201~217
- [ 9 ] Milovanović G V. Calculation of Gaussian-type quadratures with multiple nodes [A]. Spalević M M, Cvetković A S. Mathematical and Computer Modelling [C]. 2004, 39: 325~347
- [ 10 ] Lether F G, Wenston P R. Minimax approximations to the zeros of  $P_n(x)$  and Gauss-Legendre quadrature [J]. Computational and Applied Mathematics, 1995, 59: 245~252
- [ 11 ] Swarztrauber P N. On computing the points and weights for Gauss-Legendre quadrature [J]. Siam J Sci Comput, 2002, 24(3): 945~954
- [ 12 ] Babolian E, Masjedjamei M, Eslahchi M R. One numerical improvement of Gauss-Legendre quadrature rules [J]. Applied Mathematics and Computation, 2005, 160: 779~789

## High-precision Numerical Computation of High-degree Gauss-quadrature Nodes

Zhang Qingli<sup>1</sup>, Wang Xiaomei<sup>2</sup>, Yin Shaotang<sup>1</sup>, Jiang Haihe<sup>3</sup>

(1. Anhui Institute of Optics and Fine Mechanics, CAS, Hefei 230031, China;

2. Key Lab of Environment Optics and Technology of CAS, Hefei 230031, China;

3. Hefei Institute of Physical Sciences, CAS, Hefei 230031, China)

**[Abstract]** Gauss quadrature is used widely in many fields such as the engineering numerical computation, X-ray diffraction profile analysis, spectroscopy, and so on. The nodes and weight factors of Gauss-quadrature are essential data to the numerical integration. A method to compute the zeroes of the high-degree Legendre, Laguerre and Hermite polynomials, which are the nodes of Gauss-Legendre, Gauss-Laguerre and Gauss-Hermite Quadrature, respectively, is studied, and a very efficient algorithm scan-iteration method(SIM) is given. According to the properties of Legendre, Laguerre and Hermite polynomials, their definitions are modified a little, and the stable recursive relations to compute their value are obtained. To extract these polynomials, their root intervals are searched with a certain step within a certain range. After the intervals of all roots are obtained, the roots with the desired precision can be gotten by the general iteration methods such as secant or bisection method. Numerical experiments indicate that the method is very efficient and the high-precise roots of Legendre, Laguerre and Hermite polynomials can be extracted.

**[Key words]** Gauss quadrature; Legendre polynomial; Laguerre polynomial; Hermite polynomial; extract roots