

## 第一章 Monte Carlo 方法基础

Monte Carlo 方法起源于二次大战中的原子弹设计研究。中子在非均匀介质（如反应堆芯或原子弹）中的散射和输运问题由一组复杂的积分-微分方程描述，当时 Metropolis 和 Ulam 等人采用了一种数值的实验方法来计算而非直接求解输运方程。为了保密起见，他们将该方法以著名的赌城-摩纳哥王国的 Monte Carlo 城（该城现在主要以 F-1 车赛的主办地之一而闻名）而命名（另一种说法是因为 Ulam 的叔叔每年要去 Monte Carlo 赌博）。实际上，它是采用随机数于各种物理计算和模拟实验，以类似于赌博中掷骰子的方法来随机决定其中某个单独事件的结果，因此模拟本身颇有游戏的色彩。对于模拟产生的大量随机事件发生后的最终结局应是服从基本物理规律和统计规律的，从而得到物理问题的正确答案。通常，这类物理问题本身相当复杂，很难用解析的方法进行分析和计算，而 Monte Carlo 方法特别适用于如基于统计力学和量子力学等领域中的复杂问题，是一很好的数值计算和数值模拟的方法，在凝聚态物理（表面物理、临界现象、非晶态）、应用物理（金属学、扩散和偏析）、理论物理（量子场论、统计理论、基本粒子、核物理）、化学（化学反应、高分子物理）以及非线性现象（分形、逾渗）等研究领域起作非常重要的作用，成为科学研究的一种标准手段。除了用于研究数学物理问题以外，它的应用还已渗透到了经济和技术等其它各个领域，充分显示出其独到之处。

学习 Monte Carlo 模拟方法的意义，不仅在于它是科学研究的重要和普遍的手段，而且由于它的简单易操作性可以使学生更容易理解和掌握理论课程中的知识，如果再借助于计算机绘图方法，更使得一些深奥的理论概念成为直观上可以想象和理解的。

### §1.1 随机数产生器

许多数值计算都需要用随机数来构造初始构型或直接用于数值模拟。尽管人们会认为计算机可以产生随机数，但实际上在计算机中没有什么东西是随机的，计算机是按照给定的程序严格执行的，如果程序起始是一样的话，其结果也应该是一样的，即可预测的。随机数产生器其实是一伪随机数产生器。而随机数的随机性是指单个随机数之间的关联很小；当用两个不同的随机数产生器产生的随机数序列代入某一实际的应用程序后，其造成的统计输出结果值应该是一致的。如果不是的话，至少其中一个随机数产生器是不好的。有一些统计测试标准可以用来判断随机数的质量。

#### 1.1.1 均匀随机数产生器

##### 1.1.1.1 伪随机数的要求

最有用的随机数是 $[0,1]$ 区间内均匀分布的随机数。对于一个随机数产生器的

好坏, 有三个判断标准, 好的随机数产生器要求:

- 1、一个随机数序列并非是永不重复的, 在经过一个周期后, 其序列将再次形成一个循环重复。因此需要有长的循环周期。如在 32 位机上, 要求周期接近最大可取的正整数  $2^{31}-1=2147483647$  (整数的取值区间是  $[-2^{31}, 2^{31}-1]$ ), 也可设计出周期更长的随机数序列。
- 2、有良好的随机性, 也就是说, 随机数序列  $\{x_n\}$  中的所有随机数之间没有大的关联, 或自相关函数  $C(l)$  ( $l=1, 2, 3, \dots$ ) 的值很小。说明这个关联的一个办法是在  $xy$  平面上绘出点  $(x_n, x_{n+l})$ 。好的随机数产生器是, 对于任何  $l$  值, 产生的平面上数据点的分布都是非常均匀的。而差的情况是有条带、格点等非均匀分布出现。
- 3、由于 Monte Carlo 计算中需要大量使用随机数, 因此要求算法的速度非常快。为了得到好的统计结果, 随机数产生器的速度也是一个非常重要的考虑因素。

### 1.1.1.2 Lehmer 线性同余法

最简单的均匀随机数的产生法是线性同余法, 随机数序列  $\{x_n\}$  是按线性关系

$$I_{n+1} = (aI_n + b) \bmod m, \quad (1.1.1.2-1)$$

$$x_n = I_n / m, \quad (1.1.1.2-2)$$

得到。整数  $I_1, I_2, \dots$  的取值是 0 至  $m-1$  中的一个值, 故实数  $x_n$  是严格小于 1 的, 但偶尔会为 0。也因为  $x_n$  的取值个数为  $m$ , 因此希望  $m$  足够大以使得  $[0, 1]$  区间中的随机数分布足够稠密。这里的常数  $a$  是乘子,  $b$  是增量,  $m$  是模数, 如何选择它们就决定了产生器的质量。  $b=0$  时称为乘同余法,  $b \neq 0$  时为混合同余法。

线性同余法的优点是速度快, 每次调用只要几个操作步骤即可, 因此是最为常用的。不好之处是序列相关性方面还差一些, 如果用  $k$  个连续随机数在  $k$  维空间画点 (一个点的  $k$  个坐标值) 的话, 那么这些点不会均匀充满  $k$  维空间, 而是布满在  $k-1$  维平面上, 形成一种类晶体结构 ( $k=2$  时, 用连续的两个随机数在  $xy$  平面上绘出点, 则点是沿着直线段的), 最多会有  $(k!m)^{1/k}$  个平面。如果常数  $a, b, m$  选择不合适的话, 则平面数远远小于此值。

$m$  的值常取为 32 位计算机所可能取的最大值  $\sim 2^{31}$ 。因此连续使用随机数于 3 维空间中, 则空间的平面数的上界为 2344, 所以当你关注的体积中的一小块时, 平面的规则离散性对计算结果有很大影响。

例如, 以前一个有名的子程序 RANDU 在 IBM 主流机型上使用了多年, 并且移植到了多种其他系统上, 它采用的常数值是:

$$m = 2^{31} = 2147483648, \quad a = 2^{16} + 3 = 65539,$$

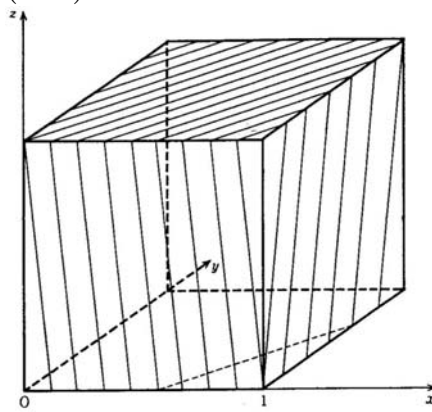


图 1.1.1.2-1 RANDU 形成的 3 维晶体结构。

$b=0$ 。如果以连续的 3 个随机数作为  $(x, y, z)$  坐标画图的话, 可以发现它产生的平面数仅为 15, 平面  $9x-6y+z=k$  ( $k=-5, -4, \dots, 8, 9$ ) 以等间隔排列的 (图 1.1.1.2-1), 这是因为随机数满足方程

$$\begin{aligned} x_{n+2}-6x_{n+1}+9x_n &= \{65539^2-6\times 65539+9\}x_n \\ &= 2\times 2^{31}x_n = 0 \pmod{2^{31}} \end{aligned} \quad (1.1.1.2-3)$$

有趣的是, 某个计算中心的顾问对此的解释是: “我们保证每一个随机数是单独随机的, 但不能同时保证其它也是随机的”。

(1.1.1.2-1) 式将自身循环, 显见周期不会超过  $m$ 。如果适当选择常数  $a, b, m$ , 可使周期有最大的长度  $m$ , 此时, 0 至  $m-1$  之间的所有整数可出现一次, 因此随意选择的任一初始值  $I_0$  与其它初始值有相同的效果。从理论和实际效果看,  $b=0$  和  $b\neq 0$  是差不多的。

对混合同余法, 可取  $m$  为最大整数  $2^{31}$ , 而  $a$  和  $b$  的选择要经过仔细研究, 如当  $a$  满足  $(a \bmod 8)=5$ ,  $m$  满足  $m/100 < a < m-\sqrt{m}$  且二进制表示上没有特别的形态,  $b$  是奇数满足  $b/m \sim 1/2 - \sqrt{3}/6 \approx 0.21132$  的情形下可以给出好的随机数。

### 1.1.1.3 16807 产生器

有一个被认为是“最低标准”的产生器是:

$$a=7^5=16807, b=0, m=2^{31}-1=2147483647. \quad (1.1.1.3-1)$$

它已经通过了许多理论测试和大量实用的考验, 被认为是 32 位机上最有效的产生器, 可以作为其他产生器与之比较的标准, 但它仍然不是十全十美的。

要把 (1.1.1.3-1) 式直接变成计算机程序并不是一件易事, 因为 32 位机的整数区间是  $[-2^{31}, 2^{31}-1]$ ,  $aI_n$  在取模之前就可能超过最大整数值。如果计算机可以自动取模的话, 则编程就很容易了, 只需给一初始值然后连续产生  $I_{n+1}=16807I_n$  即可, 但这样的话, 物理问题可能其值已超界但计算机仍能持续运行, 故实际上计算机系统不会自动取模, 而必须设计取模的方法。

Schrage 方法: 设  $m$  可表示为

$$m=aq+r, q=[m/a], r=m \bmod a, \quad (1.1.1.3-2)$$

如:  $2147483647=16807\times 127773+2836$ , ( $q=127773, r=2836$ )。则对:  $r < q$  和  $0 < z < m-1$ ,  $a(z \bmod q)$  和  $r[z/q]$  取值区间是  $[0, m-1]$ , 有

$$\begin{aligned} az \bmod m &= \left\{ \frac{z}{q}(aq+r) - \frac{rz}{q} \right\} \bmod m \\ &= \left\{ [z/q](aq+r) + \frac{z \bmod q}{q}(aq+r) - r[z/q] - r \frac{z \bmod q}{q} \right\} \bmod m \\ &= \{a(z \bmod q) - r[z/q]\} \bmod m \\ &= \begin{cases} a(z \bmod q) - r[z/q], & \text{if } \geq 0, \\ a(z \bmod q) - r[z/q] + m, & \text{otherwise} \end{cases} \end{aligned} \quad (1.1.1.3-3)$$

另外一种简便但不严格的编程方法是直接利用整数的区间。当 4 字节的整数  $aI_n$  超过最大值  $2^{31}-1$  时, 计算机将保留后 32 位有效位数, 而自动扔掉前面多余的位数 (因此, 随机数序列不等同于 Schrage 方法得到的结果), 剩下的位数中最左边的第一位决定了该整数的符号 (0 为正, 1 为负)。该位有一半的可能性为 1, 即整数值成为负的。这时可取绝对值, 或加  $2^{31}$  使它变为正的, 但  $2^{31}$  已经超过最大值, 故要加上  $2^{31}-1=2147483647$  再加 1, 最后再乘以  $2^{-31}=4.656612\times 10^{-10}$  以得到  $[0,1]$  区间中均匀分布的随机数。最后, 应取奇数种子值使整数值恒为奇数。

**[作业]:** 用 Schrage 方法编写随机数子程序, 用连续两个随机数作为点的  $xy$  坐标值绘出若干点的平面分布图; 再用下面的  $\chi^2$  统计测试其 2 维独立性 (总点数  $N$  大于  $2\times 10^7$ )。再取  $a=106, b=1288, m=6075$ , 重复以上计算, 总结和评价产生器的随机性质量。

#### 1.1.1.4 种子值

在用同余法时, 有时要求每次使用随机数序列时初始的种子值也是不一样的, 因此需要一个系统产生种子值的方法, 几乎每个计算机系统都有一个报告当前时间的整数值, 可以用它来构造一个初始值。例如: 大多数计算机上都有

年:  $0 \leq i_y \leq 99$ , 月:  $1 \leq i_m \leq 12$ , 日:  $1 \leq i_d \leq 31$

时:  $0 \leq i_h \leq 23$ , 分:  $0 \leq i_n \leq 59$ , 秒:  $0 \leq i_s \leq 59$

则可设种子值为:  $I_0 = i_y + 70(i_m + 12\{i_d + 31[i_h + 23(i_n + 59i_s)]\})$ , 它的值约在区间  $[0, 2^{31}-1]$  内, 第二部分的括号在 100 年内不会重复。

#### 1.1.1.5 Tausworthe 位移计数器法

Tausworthe 方法或称一般反馈位移计数器法可以用以消除同余法中的关联问题。人们曾经认为它要强于 16807 法, 但后来发现情况并非那么简单, 但至少该方法提供了随机数产生的另一种途径。该方法中即对整数进行位操作: 首先用其它方法产生一个随机的整数序列, 然后对两个整数进行 XOR (与或) 操作以产生一个新的随机整数 (需再除以  $m$  得到  $[0,1]$  区间的随机数):

$$I_n = I_{n-p} \oplus I_{n-q}, \quad (1.1.1.5-1)$$

其中的  $[p, q]$  是一对整数, 最佳的选择是满足条件

$$p^2 + q^2 + 1 = \text{prime}, \quad (1.1.1.5-2)$$

的 Mersine 素数, 如:  $[31, 3]$ 、 $[98, 27]$ 、 $[250, 103]$ 、 $[1279, (216,418)]$ 、 $[9689, (84,471,1836,2444,4187)]$  等, 其中的 R250 ( $p=250, q=103$ ) 是最为常用的产生器。一般来说  $[p, q]$  值越大, 产生的随机数质量越好, 而且起始的随机整数表的质量也很重要, 因此常用同余法产生有  $p$  个数的初始值表。

Fortran90 中对两个整数  $m$  和  $n$  进行“与或”位操作的函数是 IEOR( $m, n$ ), 例如:  $I_1 = 6, I_{148} = 11$ , 则  $I_{251} = I_1 \oplus I_{148} = 0110 \oplus 1011 = 1101 = 2^3 + 2^2 + 2^0 = 13$ , 程序



中可写成  $N(K)=\text{IEOR}(N(K-250),N(K-103))$ ，要求数组  $N$  中存储所有之前的 250 个随机整数。

16807 和 R250 法在不同的场合下各有利弊，统计检验也不能保证随机数的质量，实际上也根本不存在理想的伪随机数序列，一种产生器的好坏还是要在具体的应用中根据结果来判断。经验指出，一种新的随机数产生器要经过持久和广泛的应用累积才能知道它的各种隐含的缺陷。故 16807 和 R250 仍是最为广泛应用的，因为人们对它们已经有所了解。避开随机数序列各种缺陷的一个实用方法是，对计算中的各物理量不要顺序地使用连续的随机数序列，或它们各自采用分别的随机数序列并有各自的种子值。还应使循环的周期错开，或者随机淘汰序列中的任意个随机数。

### 1.1.1.6 Fibonacci 延迟产生器

位移计数器法是更一般的 Fibonacci 延迟产生器的一个特例。其思想是用序列中的两个整数进行操作得到后续的整数 (Fibonacci 数序列 1,2,3,5,8,13,21,34,... 的特点是其中的一个整数为前两位数之和，用以描述兔子后代的繁衍)，

$$I_n = I_{n-p} \otimes I_{n-q} \bmod m, \quad (1.1.1.6-1)$$

其操作符  $\otimes$  可以是：加、减、乘、XOR。整数对  $[p, q]$  表示延迟，取值并非按 Fibonacci 数序列，而是根据统计验证后确认。Fibonacci 延迟产生器较线性同余法的优势在于它的周期非常长，32 位机上的最大周期为  $(2^p - 1)2^{31}$  ( $p > q$ )。

更复杂一些的有：带载减法产生器，

$$I_n = I_{n-22} - I_{n-43} - C, \quad (1.1.1.6-2)$$

$$\begin{cases} C = 0, & \text{if } I_n \geq 0 \\ C = 1, I_n = I_n + (2^{32} - 5), & \text{if } I_n < 0 \end{cases}.$$

带载减法 Weyl 产生器：

$$J_n = J_{n-22} - J_{n-43} - C, \quad (1.1.1.6-3)$$

$$\begin{cases} C = 0, & \text{if } J_n \geq 0 \\ C = 1, J_n = J_n + (2^{32} - 5), & \text{if } J_n < 0 \end{cases}$$

$$K_n = (K_{n-1} - 362436069) \bmod 2^{32}.$$

$$I_n = (J_n - K_n) \bmod 2^{32}$$

### 1.1.1.7 Marsaglia 产生器

Marsaglia 产生器是组合产生器法之一种，组合法的原理是用两个不同的随机数产生器序列生成另一个随机数序列。该产生器的周期可达  $2^{144} \approx 2.23 \times 10^{43}$ 。Marsaglia 产生器中，第一个随机数产生器序列实际上就是减法操作的 Fibonacci 延迟产生器：

$$x_n = \begin{cases} x_{n-p} - x_{n-q}, & \text{if } \geq 0 \\ x_{n-p} - x_{n-q} + 1, & \text{otherwise} \end{cases}, \quad (1.1.1.7-1)$$

其中的  $[p, q]$  整数对的值选为  $[97, 33]$ ，因此其算法要求存储所有前面的 97 个随机数值。

第二个随机数产生器序列中采用如下定义的算术操作：

$$a \circ b = \begin{cases} a - b, & \text{if } \geq 0 \\ a - b + 16777213/16777216, & \text{otherwise} \end{cases}, \quad (1.1.1.7-2)$$

则该序列的第  $n$  个值为：

$$y_n = y_{n-1} \circ (7654321/16777216), \quad (1.1.1.7-3)$$

组合后的第  $n$  个随机数值为：

$$z_n = x_n \circ y_n, \quad (1.1.1.7-4)$$

$y$  序列只要求有一个初始种子值，而  $x$  序列需要 97 个，这些初始值之间也需是充分随机的，因此可用其它需较少种子值的产生器或组合法首先产生这些值。

## 1.1.2 伪随机数的统计检验

### 1.1.2.1 独立性与相关系数

伪随机数的好坏通常是由各种统计检验来判定，这是必要条件但不是充分条件（不能通过检验的必定不是好的产生器，但通过有限个检验指标的也不能保证它是好的）。最主要的检验有两种：均匀性检验和独立性检验。均匀性是指在  $[0,1]$  区间内等长度子区间中随机数的数量是一样的，独立性是指按先后顺序出现的随机数中，每个随机数的取值与其相距一定间隔的随机数取值之间无关。

讨论伪随机数序列独立性的第一个方法是顺序相关法，它用相邻两个随机数的自相关函数（或相关系数）来标识伪随机数序列的独立性情况，相关系数越小，独立性越好。间距为  $l$  的自相关函数是

$$C(l) = \frac{\langle x_n x_{n+l} \rangle - \langle x_n \rangle^2}{\langle x_n^2 \rangle - \langle x_n \rangle^2}, \quad (1.1.2.1-1)$$

其中，平均值的定义是  $\langle x_n \rangle = \sum_{n=1}^N x_n / N$ 。当两个随机数序列  $x_n$  与  $x_{n+l}$  不相关时， $\langle x_n x_{n+l} \rangle = \langle x_{n+l} \rangle \langle x_n \rangle = \langle x_n \rangle^2$ ，故相关系数为 0。这时，在  $xy$  平面上用坐标点  $(x_n, x_{n+l})$  作图的话，点的分布是无规的（图 1.1.2.1-1）。但要注意到，相关函数很小只能说明  $x_n$  与  $x_{n+l}$  之间的线性关系很弱，不能保证它们之间没有其他的函数关系。

### 1.1.2.2 均匀性检验 - 频率检验

将区间  $[0,1]$  分为  $K$  个子区间，统计随机数落在第  $k$  个子区间的实际频数  $n_k$ ，它应当趋近于理论频数  $m_k = N/K$ ，( $k=1, \dots, K$ )，注意此处的  $n_k$  是整数而  $m_k$  可以是小数。令统计量

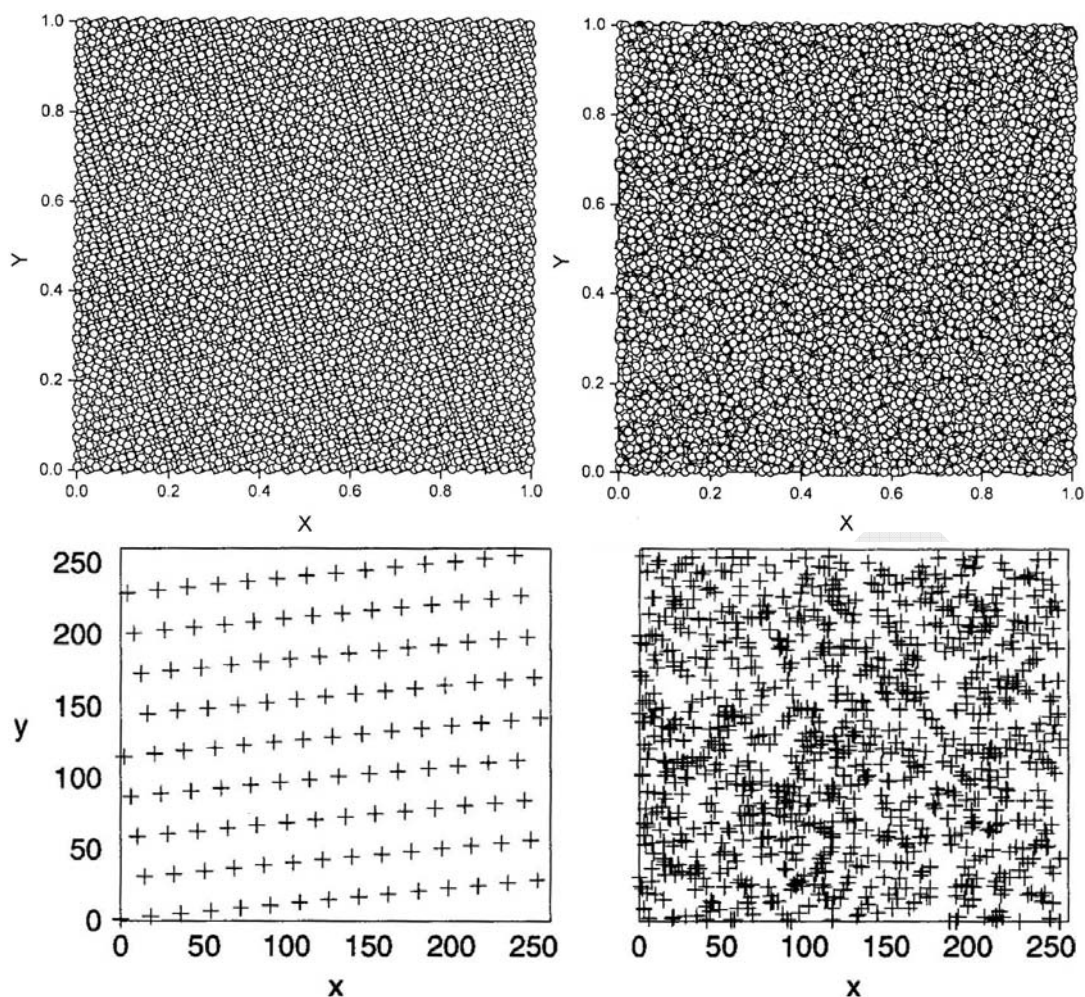


图 1.1.2.1-1 用连续的两个随机数作为点 $(x, y)$ 的坐标作图可以直观看出随机数之间的关联性。显然左边是不好的随机数。左上显示出条带结构，左下则是规则网格结构。

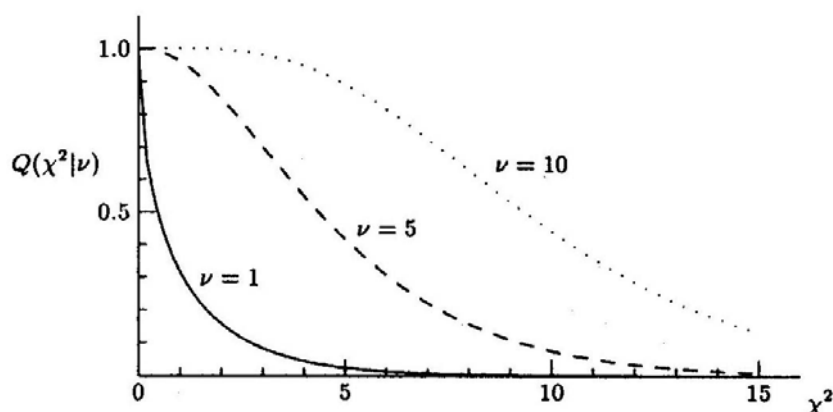
$$\chi^2 = \sum_{k=1}^K \frac{(n_k - m_k)^2}{m_k}, \quad (1.1.2.2-1)$$

式中取平方是为了消除负号，另外从相对误差来说， $m_k$ 越大，允许 $|n_k - m_k|$ 越大，故取比值 $(n_k - m_k)^2 / m_k$ 。如果 $\chi^2$ 值很大，表示远远偏离理想值，因此要求 $\chi^2$ 值尽可能小，但如果它趋于0则有可能 $N$ 已进入循环。通常求和中的每一项的大小约为1，因此 $\chi^2$ 的值约为 $K$ 。

概率论中的 Pearson 定理说明，(1.1.2.2-1) 式的极限概率分布是 $\chi^2$ 分布，

$$P(\chi^2 \leq x | \nu) = \frac{1}{2^{\nu/2} \Gamma(\nu/2)} \int_0^x t^{(\nu-2)/2} e^{-t/2} dt, \quad (1.1.2.2-2)$$

它给出了 (1.1.2.2-1) 式中的其中 $\chi^2 \leq x$ 的概率。整数 $\nu$ 是系统的自由度，表示独立测量的次数。由于有一个约束条件存在， $\sum_{k=1}^K m_k = N$ ，故自由度 $\nu = K - 1$ 。余函数 $Q(x | \nu) = 1 - P(x | \nu)$ 则给出了 $\chi^2 > x$ 的概率 (图1.1.2.2-1)。因此，当给定显著水平 $\alpha$ 后(或置信度 $1 - \alpha$ )，由方程 $Q(\chi_\alpha^2 | \nu) = \alpha$ 或 $P(\chi_\alpha^2 | \nu) = 1 - \alpha$ 解出 $\chi_\alpha$ 值，或从已有的 $\chi^2$ 表中查得 $\chi_\alpha$ 值，如果由 (1.1.2.2-1) 式计算出来的 $\chi$ 小于 $\chi_\alpha$ ，则

图 1.1.2.2-1 几个自由度下的  $\chi^2$  分布。

认为在此置信度下，伪随机数序列在  $[0,1]$  中是均匀分布的。

### 1.1.2.3 独立性检验 - 多维频率检验

将伪随机数序列用任意一种办法进行组合，每  $S$  个随机数作为  $S$  维空间中的一个点的坐标值，于是可以构成一个点序列。把  $S$  维空间中的单位方体分成为  $K$  个子方体， $K_0 = K^{-1/S}$  是方体边长，统计落在第  $k$  个子方体中的实际频数  $n_k$ ，它应当趋近于理论频数  $m_k = NK_0^S = N/K$ ，( $k=1, \dots, K$ )。同一维频率检验一样，统计量 (1.1.2.2-1) 式应有自由度为  $K-1$  的  $\chi^2$  分布 (1.1.2.2-2) 式。

例如，将  $2N$  个随机数序列分为两组： $\{x_1, x_3, \dots\}$  和  $\{x_2, x_4, \dots\}$ ，分别作为平面中  $N$  个点的  $x$  和  $y$  坐标值。在  $xy$  平面中作  $K_0 \times K_0$  个小正方形网格区域 ( $K_0 = \sqrt{K}$ )，落在第  $(i, j)$  个网格区域中的实际频数为  $n_{ij}$ ，则

$$\chi^2 = \sum_{ij=1}^{K_0} \frac{(n_{ij} - N/K_0^2)^2}{N/K_0^2} \quad (1.1.2.3-1)$$

### 参考文献

- [1] D.P. Landau and K. Binder, A Guide to Monte Carlo Simulations in Statistical Physics (Cambridge University Press, Cambridge, 2000) (该书有世界图书出版公司中国重印版) (第二章中列出了随机数的若干种产生方法)
- [2] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, Numerical Recipes in FORTRAN-The Art of Scientific Computing, Second Edition (Cambridge University Press, Cambridge, 1992) (第七章中给出了随机数的算法)
- [3] 伏见正则,《概率的方法とシミュレーション》(岩波讲座:应用数学,方法[10]) (日文), (岩波书店, 1994 年) (第一章讨论了随机数的产生方法和特性)
- [4] 裴鹿成、张孝泽,《蒙特卡罗方法及其在粒子输运问题中的应用》(科学出版社, 1980 年) (第四章和第五章分别叙述随机数的检验和产生方法)
- [5] G. Marsaglia, A. Zaman and W.W. Tsang, Stat. & Probab. Lett. **9** (1990) 35 (Marsaglia 随机数产生器方法)



## §1.2 由已知分布的随机抽样

大多数实际物理问题中，随机变量并不是均匀分布的，它们有一定的几率分布密度函数，如微分散射截面中的散射角的几率分布即为微分散射截面。通常对一个具有分布几率密度函数  $p(x)$  的伪随机变量的抽样步骤是：先产生  $[0,1]$  区间均匀分布的随机数序列  $\{\xi_n\}$ ，然后再从这个伪随机数总体中抽取一个子样  $\{x(\xi_n)\}$ ，使其满足分布密度函数  $p(x)$ 。只要随机数总体有好的独立性，则抽得的子样也有好的随机性。这个子样  $\{x_n\}$  就是满足特定分布的随机数。

### 1.2.1 直接抽样法

#### 1.2.1.1 离散型变量分布

设变量  $x$  是离散型的，取值为  $x_1, x_2, \dots$ ，相应值出现的几率为  $p_1, p_2, \dots$ 。对于一个物理量常常给出的不是归一化的几率值  $p$ ，而是其截面值  $\sigma$ ，则可将其归一化为几率，

$$p_i = \sigma_i / \sum_i \sigma_i. \quad (1.2.1.1-1)$$

这里的  $p_i$  是无量纲的。如果从  $[0,1]$  区间中均匀抽样得到的随机数  $\xi$  满足下式时，

$$\sum_{i=1}^{n-1} p_i < \xi \leq \sum_{i=1}^n p_i, \quad (1.2.1.1-2)$$

则物理量  $x$  取值为  $x_n$ 。

例如， $x$  可取 3 个值， $x_1, x_2, x_3$ ，它们出现的几率分别为  $\frac{2}{8}, \frac{5}{8}, \frac{1}{8}$ ，则随机数小于  $\frac{2}{8}$  时实现  $x_1$ ，在区间  $[\frac{2}{8}, \frac{7}{8}]$  中时实现  $x_2$ ，大于  $\frac{7}{8}$  时实现  $x_3$ 。

例如，Poisson 分布是离散型分布，

$$p_n = e^{-\lambda} \frac{\lambda^n}{n!}, \quad (1.2.1.1-3)$$

对此分布进行抽样得到第  $n$  个事件发生的条件为，

$$\sum_{i=0}^{n-1} \frac{\lambda^i}{i!} < e^\lambda \xi \leq \sum_{i=0}^n \frac{\lambda^i}{i!}. \quad (1.2.1.1-4)$$

#### 1.2.1.2 连续型变量分布

设连续型变量  $x$  在区间  $[a, b]$  中取值，可视为将上述的离散情形取连续极限，即  $\lim_{N \rightarrow \infty} \Delta x_n = \lim_{N \rightarrow \infty} (b-a)/N \rightarrow 0$ ， $N$  为离散值的总数，因此 (1.2.1.1-2) 式左右两边相等，求和变为积分

$$\xi(x) = \int_a^x p(x') dx', \quad (1.2.1.1-5)$$

这里  $p(x)$  是已归一化的几率密度分布函数，其定义是：

$$\int_a^b p(x) dx = 1, \quad p(x) \geq 0, \quad (1.2.1.1-6)$$

$$dP(x \rightarrow x+dx) = p(x) dx,$$

式中  $P$  是无量纲几率，因此几率密度函数  $p(x)$  的量纲为自变量量纲的倒数： $[p(x)] = 1/[x]$ ，这与离散情况下不同。如  $[a, b]$  区间上均匀分布的几率密度函数为  $p(x) = 1/(b-a)$ 。对于未归一化的物理量如截面  $\sigma(x)$ ，则可定义其几率密度函数为  $p(x) = \sigma(x) / \int_a^b \sigma(x) dx$ 。

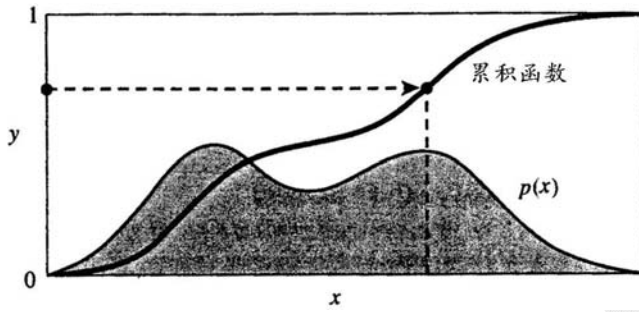


图 1.2.1.1-1 概率密度分布函数与累积函数。

$\xi(x)$  称为累积函数， $\xi(a)=0, \xi(b)=1$  且是单调增（图 1.2.1.1-1）。（1.2.1.1-5）式也可以数值计算，即对每一  $x$  值求相应的  $\xi$ ，然后列表，当给定某一随机数后从表中插值得到  $x$ 。但此法很繁，最好是由上式解析反解出  $x(\xi)$  的函数表达式，即求反函数，这对一些简单的几率密度函数解析表达式是可以做到的。

例如，粒子随机运动的自由程分布为指数分布，

$$p(x) = \begin{cases} \lambda^{-1} e^{-x/\lambda}, & x > 0 \\ 0, & \text{otherwise} \end{cases}, \quad (1.2.1.1-7)$$

其中的  $\lambda$  为运动平均自由程，则由（1.2.1.1-5）式，

$$\xi = \int_0^x \lambda^{-1} e^{-t/\lambda} dt = 1 - e^{-x/\lambda}, \quad (1.2.1.1-8)$$

则求反函数后得

$$x = -\lambda \ln(1 - \xi) = -\lambda \ln \xi, \quad (1.2.1.1-9)$$

因为随机数  $1 - \xi$  与  $\xi$  是等价的。

例如，粒子输运问题中散射方位角余弦分布

$$p(x) = \pi^{-1} (1 - x^2)^{-1/2}, \quad -1 \leq x \leq 1, \quad (1.2.1.1-10)$$

的抽样是： $x = \sin 2\pi\xi$  或  $x = \cos 2\pi\xi$ （为什么？）。

**[作业]:** 在球坐标系  $(\rho, \theta, \varphi)$  下产生球面上均匀分布的随机坐标点，给出其直接抽样方法。