



实验物理中的统计方法

第三章：蒙特卡罗方法

杨振伟

回顾

概率的基本概念

随机变量与概率密度函数

随机变量的均值与方差

常见的概率分布

本章要点

- 蒙特卡罗方法
- 随机数产生子
- 任意分布抽样的函数变换法和舍选法
- 蒙特卡罗模拟在粒子物理与核物理中的应用

蒙特卡罗方法简介

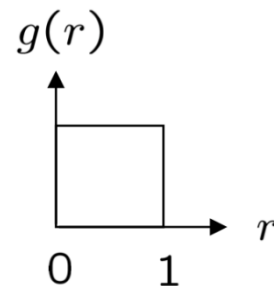
蒙特卡罗方法 (Monte Carlo Method) 是一种数值分析技术, 有时也称为统计实验法。它利用计算机模拟研究随机性问题, 或者被映射为随机性问题的确定性问题。

常用于近似求解物理或数学问题的一种方法, 在科学与工程研究中应用广泛。

蒙特卡罗方法要求按照给定的概率分布产生一系列随机数。

蒙特卡罗方法简介（续）

通常的步骤为：



- 1) 产生随机数序列 $r_1, r_2, \dots, r_m \sim U(0,1)$
- 2) 利用 r_1, r_2, \dots, r_m , 按我们感兴趣的概率密度 $f(x)$ 生成另一个随机数序列 x_1, x_2, \dots, x_n 【注： $f(x)$ 中的 x 可以是矢量】
- 3) 利用 x_1, x_2, \dots, x_n 估计 $f(x)$ 的某种特性，例如： $x \in [a, b]$ 占有所有 x 值的比例 $\rightarrow \int_a^b f(x) dx$

第一层面上的应用： 蒙特卡罗计算 = 积分

第二层面上的应用： 蒙特卡罗变量 = “模拟的数据”

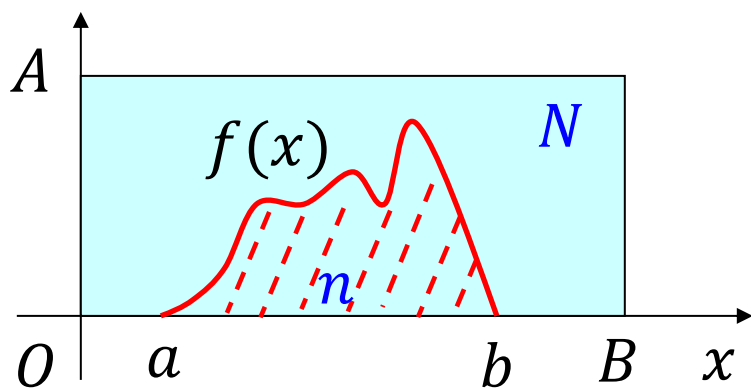
蒙特卡罗法计算积分

计算积分值 $\int_a^b f(x)dx$

解析解: $\int_a^b f(x)dx = F(x)|_{x=b} - F(x)|_{x=a}$ 函数必须解析可积

数值解: $\int_a^b f(x)dx \simeq \sum_{i=1}^n f(x_i) \frac{b-a}{n}$ 自变量不能太多

蒙特卡罗方法:



在AB区间均匀投点, 总数为 N 。

$$\int_a^b f(x)dx \simeq \overline{OA} \cdot \overline{OB} \cdot \frac{n}{N}$$

其中 n 为 $f(x)$ 曲线下的投点数。

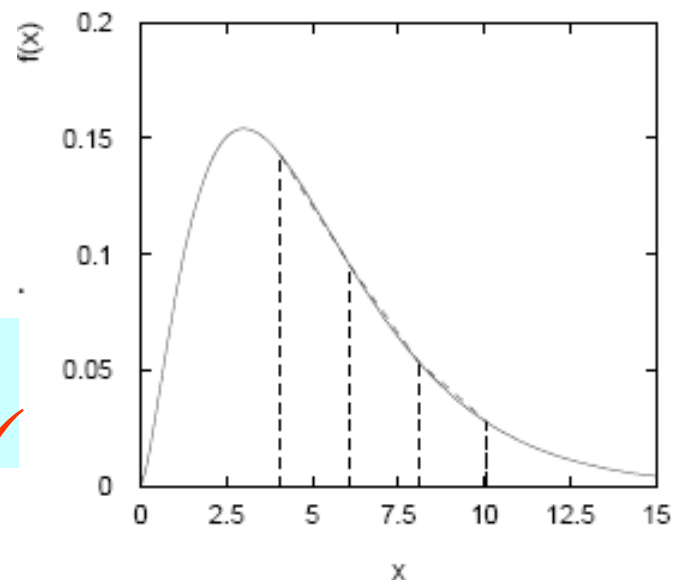
对函数是否解析可积、
是否太多自变量无要求

蒙特卡罗方法中的精度问题

采用蒙特卡罗方法(MC)计算积分
与传统的梯形法相比有如下特点

一维积分:

MC精度: $\propto 1/\sqrt{n}$ (n : 随机数的个数)
梯形法精度: $\propto 1/n^2$ (n : 子区间的数目) ✓



多维积分:

MC精度: $\propto 1/\sqrt{n}$ (与积分维度无关) ✓
梯形法精度: $\propto 1/n^{2/d}$ (d : 积分维度)

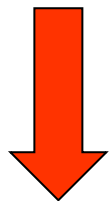
对于维数大于4的积分, 用蒙特卡罗方法计算积分总是最好。

本章要点

- 蒙特卡罗方法
- 随机数产生子
- 任意分布抽样的函数变换法和舍选法
- 蒙特卡罗模拟在粒子物理与核物理中的应用

随机数的产生

用物理方法产生
真随机数



- 不可重复
- 产生速度慢

用数学方法产生
伪随机数



- 可重复
- 产生的速度快

可重复既是缺点，也是优点。

随机数产生子

目标：产生 $[0,1]$ 范围内均匀分布的数。



随机数产生子

实际上，用计算机算法产生 r_1, r_2, \dots, r_n 。

例如：线性乘同余法 (MLCG)

$$n_{i+1} = (a n_i) \bmod m$$

n_i : 整数

n_0 : 种子 (初值)

a : 乘子 (multiplier)

m : 模量 (modulus)

mod: 除余算子

参数 a, m, n_0 给定，可以得到一个确定的随机数序列 n_0, n_1, \dots

随机数产生子：周期性

这个随机数序列是周期性的！

例如： $a = 3, m = 7, n_0 = 1$

$$n_1 = (3 \cdot 1) \bmod 7 = 3$$

$$n_2 = (3 \cdot 3) \bmod 7 = 2$$

$$n_3 = (3 \cdot 2) \bmod 7 = 6$$

$$n_4 = (3 \cdot 6) \bmod 7 = 4$$

$$n_5 = (3 \cdot 4) \bmod 7 = 5$$

$$n_6 = (3 \cdot 5) \bmod 7 = 1$$



m $= 2^K$	a $= 5^{2q+1}$	n_0	周期 $= 2^{K-2}$
2^{32}	5^{13}	1	$2^{30} \approx 10^9$
2^{36}	5^{13}	1	$2^{34} \approx 2 \cdot 10^{10}$
2^{42}	5^{17}	1	$2^{40} \approx 10^{12}$

随机数序列开始重复

- 选择合适的 a, m → 长周期
 - 最大的周期是多少？
- 只使用一个周期内的随机数序列的子集

随机数产生子：随机性

如何保证这个随机数序列确实是随机的？

→ $r_i = n_i/m$ 在 $[0,1]$ 之间，但它们是随机出现的吗？

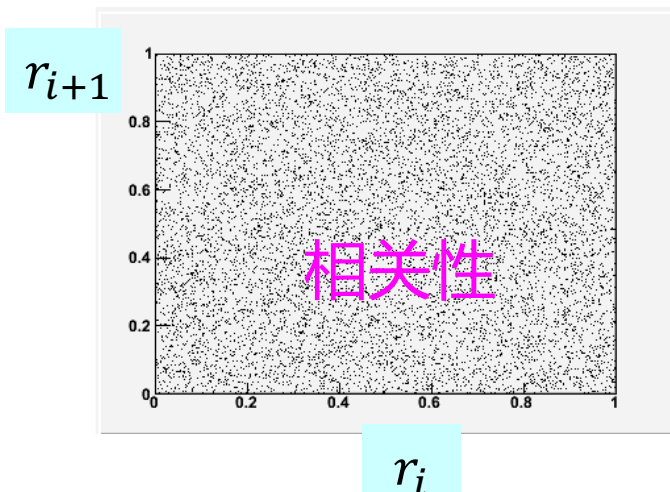
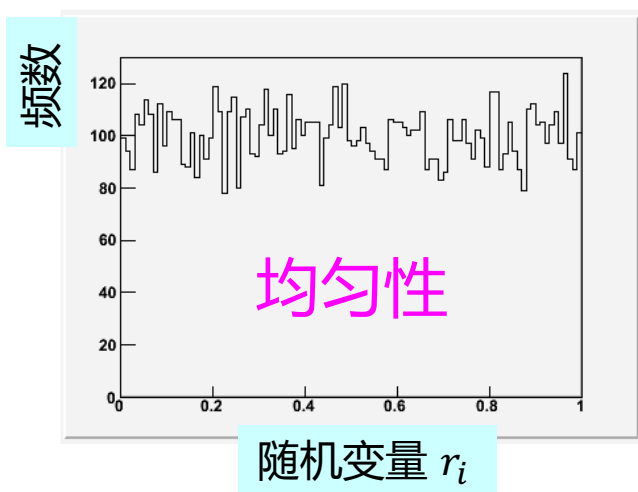
选择 a, m ，使得 r_i 能通过各种随机性检验：

- 均匀性：在 $[0,1]$ 之间均匀分布
- 独立性：两两之间不相关

例如，取

$a = 40692$,

$m = 2147483399$



对随机数序列的性质检验应当按照问题的性质有所侧重。

随机性检验

```
// random.C
```

```
void random() {
```

```
    UInt_t a, m, x0;
```

```
    TH1F *h1 = new TH1F("h1","",100,0,1);
```

```
    TH2F *h2 = new TH2F("h2","",100,0,1,100,0,1);
```

```
    a = 1220703125; //5^13
```

```
    m = 4294967296; //2^32
```

```
    x0 = 1;
```

```
    double y, yold;
```

```
    for (int i=0; i<10000; i++) {
```

```
        x = (lambda*x0)%m;
```

```
        y = (double)x/m;
```

```
        h1->Fill(y);
```

```
        if (i>1) h2->Fill(yold, y);
```

```
        x0=x;
```

```
        yold=y;
```

```
    }
```

```
}
```

```
root[0].x random.C  
root[1]h1-> Draw()  
root[2]h2-> Draw()
```

粒子物理与核物理研究中，大都采用
CERN程序库提供的随机数产生子。

计算程序中的随机数产生子

大部分计算程序都提供了随机数产生子，可以用不同算法产生随机数。

如果要处理的问题对随机数的统计性质要求很高，需要特别注意随机数产生子的统计性质。大部分同余算法（尤其是周期小于 2^{32} ）无法通过比较严格的检验，对结果会造成影响。

用于大型数据统计分析的ROOT软件包（CERN开发）提供的随机数产生子采用了一些特殊的算法，周期最长为 $2^{19937} - 1$ 。其推荐的算法比较好地兼顾了随机数的统计性质和产生速度。

参见：

<http://pdg.lbl.gov/2018/reviews/rpp2018-rev-monte-carlo-techniques.pdf>

ROOT中一些随机数产生子的说明

This class defines the **ROOT** Random number interface and it should not be instantiated directly but used via its derived classes. The generator provided in **TRandom** itself is a LCG (Linear Congruential Generator), the **BSD rand generator**, that it should not be used because its period is only 2^{31} , i.e. approximately 2 billion events, that can be generated in just few seconds.

To generate random numbers, one should use the derived class, which are :

- **TRandom3**: it is based on the "Mersenne Twister generator", it is fast and a very long period of about 10^{6000} . However it fails some of the most stringent tests of the **TestU01 suite**. In addition this generator provide only numbers with 32 random bits, which might be not sufficient for some application based on double or extended precision. This generator is however used in **ROOT** used to instantiate the global pointer to the **ROOT** generator, *gRandom*.
- **TRandomRanluxpp** : New implementation of the Ranlux generator algorithm based on a fast modular multiplication of 576 bits. This new implementation is built on the idea and the original code of Alexei Sibidanov,

The following table shows some timings (in nanoseconds/call) for the random numr

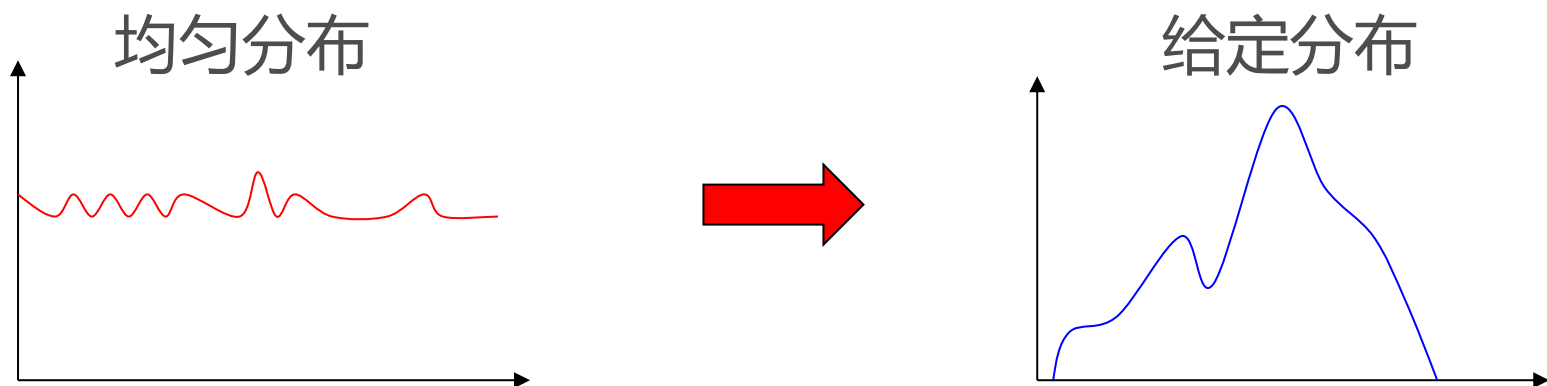
- **TRandom** 3 ns/call (but this is a very BAD Generator, not to be used)
- **TRandom2** 5 ns/call
- **TRandom3** 5 ns/call
- **TRandomMixMax** 6 ns/call
- **TRandomMixMax17** 6 ns/call
- **TRandomMT64** 9 ns/call
- **TRandomMixMax256** 10 ns/call
- **TRandomRanluxpp** 14 ns/call
- **TRandom1** 80 ns/call
- **TRandomRanlux48** 250 ns/call

<https://root.cern.ch/doc/master/classTRandom.html>

本章要点

- 蒙特卡罗方法
- 随机数产生子
- 任意分布抽样的函数变换法和舍选法
- 蒙特卡罗模拟在粒子物理与核物理中的应用

从均匀分布到任意分布的随机数



变换法

寻找某个函数，当函数的自变量取均匀分布值时，对应的函数值自动满足给定分布。

舍选法

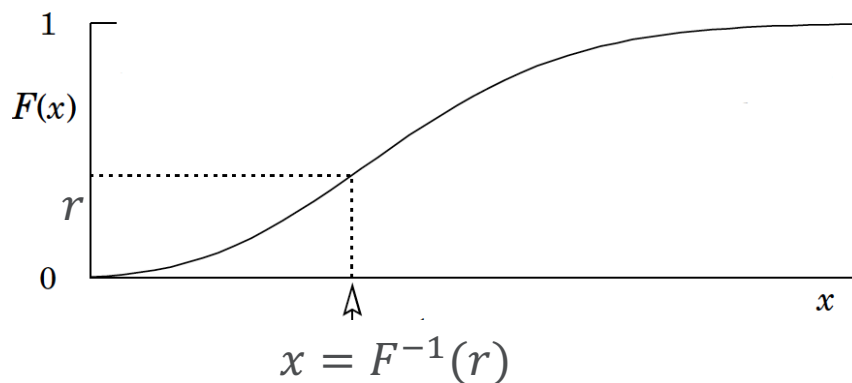
从一个随机变量与对应概率密度函数最大值构成的二维均匀分布中，按概率密度函数与自变量关系曲线切割得到。

变换法：连续分布

问题：如果我们能够产生 $[0,1]$ 区间均匀分布的随机数序列 $r_i (i = 1, 2, \dots)$ ，如何据此得到服从给定密度函数 $f(x)$ 的随机数序列 $x_i (i = 1, 2, \dots)$ ？

分布函数的重要性质： $f(x)$ 的分布函数 $F(x) = \int_{-\infty}^x f(x')dx'$ 也是一个随机变量，服从 $[0,1]$ 区间均匀分布。

假设 r 是根据 $[0,1]$ 内均匀分布产生的随机数，且 $F(x)$ 可逆，则 $r = F(x)$ 可唯一确定一个 $x = F^{-1}(r)$ ，服从分布 $f(x)$ 。



亦称反函数变换法。
适用条件：反函数可求。

变换法：证明

证明：假设随机变量 $r \sim U(0,1)$ ，概率密度 $g(r)$ ，变换 $x = h(r)$ 也是一个随机变量，设其概率密度为 $f(x)$ ，有

$$\begin{aligned} f(x) &= \int g(r) \delta(x - h(r)) dr = \int_0^1 \delta(x - h(r)) dr \\ &= \frac{1}{h'(r)} \Big|_{r=h^{-1}(x)} = \frac{1}{h'(h^{-1}(x))} \end{aligned}$$

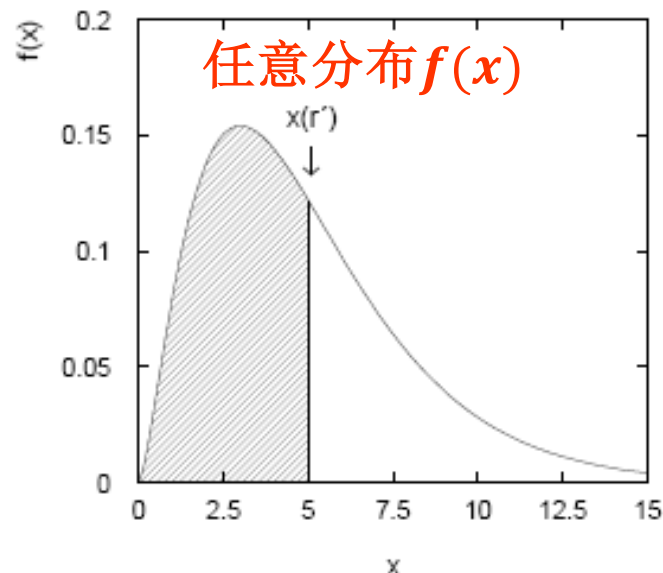
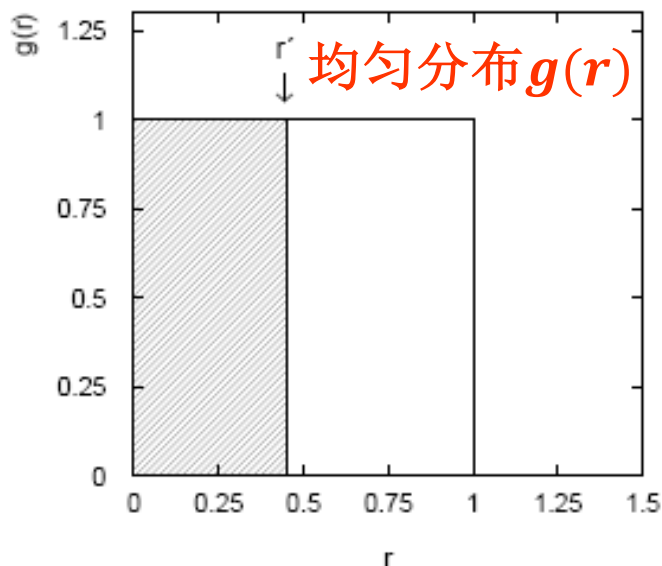
由恒等式 $h(h^{-1}(x)) = x$ ，等式两边对 x 微分：

$$h'(h^{-1}(x)) \cdot (h^{-1}(x))' = 1 \Rightarrow f(x) \frac{1}{h'(h^{-1}(x))} = (h^{-1}(x))'$$

于是

$$h^{-1}(x) = \int_{x_{\min}}^x f(x) dx = F(x) \Rightarrow x = h(r) = F^{-1}(r)$$

变换法：证明



假设将每个 r 都变换成 $x(r)$, 使得 $x(r) \sim f(x)$, 那么

$$P(r \leq r') = P(x \leq x(r'))$$

$$\int_{-\infty}^{r'} g(r) dr = r' = \int_{-\infty}^{x(r')} f(x') dx' = F(x(r')) \quad \Rightarrow \quad F(x) = r$$

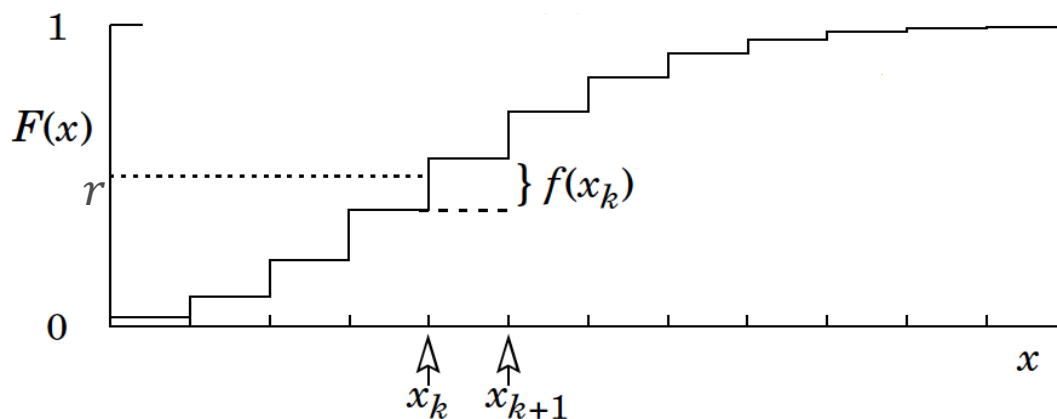
$$\text{解得 } x(r) = F^{-1}(r)$$

变换法：离散分布

对离散分布， $F(x)$ 不连续，在每个可取值 x_k ($k = 1, 2, \dots$)处的跳跃幅度为 $f(x_k)$ 。

假设 r 是根据 $[0,1]$ 区间均匀分布产生的随机数，则对应的 x_k 由下式给出：

$$F(x_{k-1}) < r \leq F(x_k) \equiv P(x \leq x_k) = \sum_{i=1}^k f(x_i)$$



例：指数分布抽样

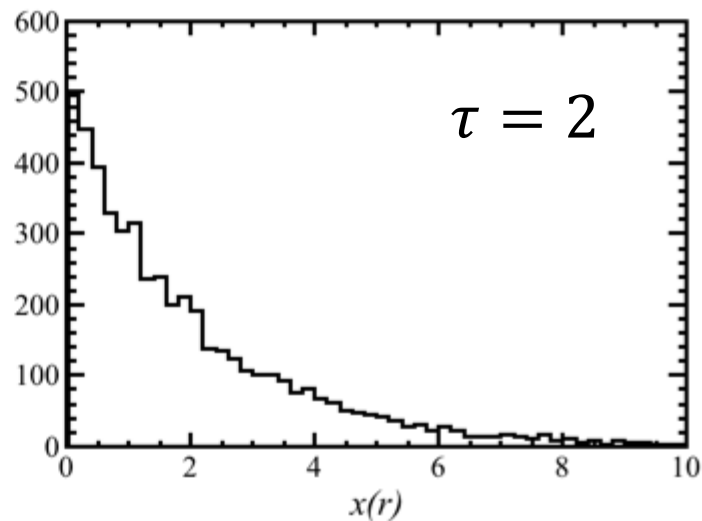
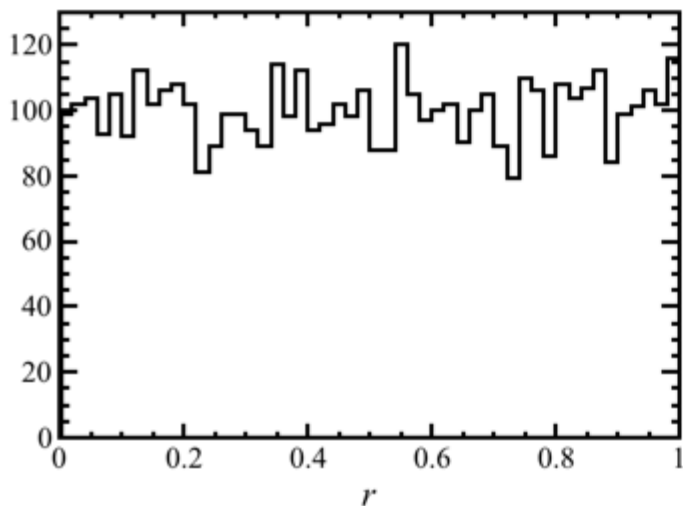
指数概率密度函数： $f(x; \tau) = \frac{1}{\tau} e^{-\frac{x}{\tau}} \quad (x \geq 0)$

令 $r = \int_0^x f(x'; \tau) dx'$, 求解 $x = x(r)$:

$$r = 1 - e^{-\frac{x}{\tau}} \Rightarrow x(r) = -\tau \log(1 - r)$$

等价于

$$x(r) = -\tau \log(r)$$

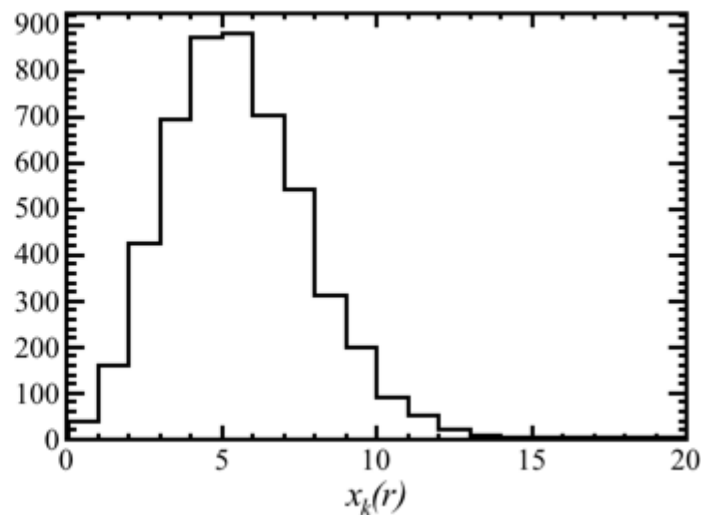
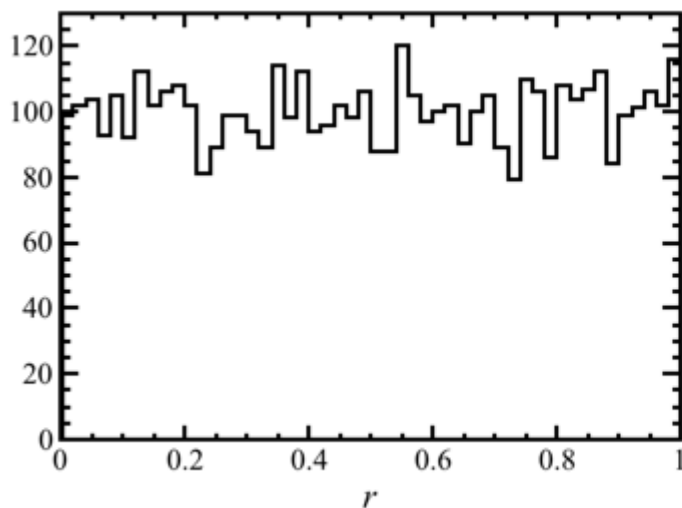


例: 泊松分布抽样

泊松分布: $P(x = k; \nu) = \frac{\nu^k e^{-\nu}}{k!} \quad (k = 0, 1, 2, \dots)$

令 $F(x_{k-1}) < r \leq F(x_k) = \sum_{n=0}^k \frac{\nu^n e^{-\nu}}{n!}$, 求解 $k = x_k(r)$:

程序中利用循环很容易求解。

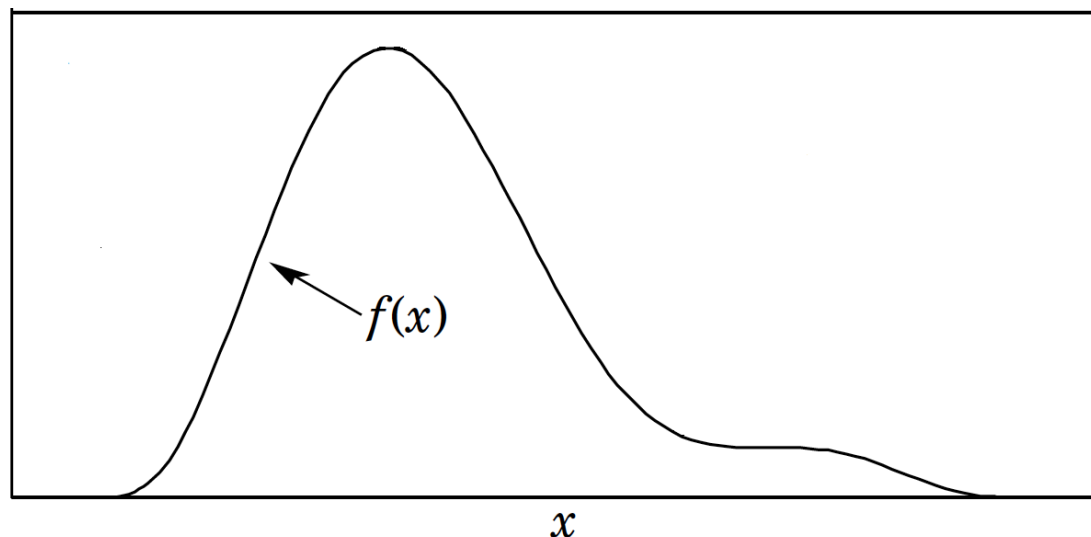


舍选法

实际应用中经常出现的情况是，概率密度 $f(x)$ 的累积分布 $F(x)$ 无法求出，或其计算过于复杂。有时候 $f(x)$ 甚至没有解析表达式。

假设在定义域内任何 x 点的概率密度 $f(x)$ 可以计算， $f(x)$ 的形状可以确定。

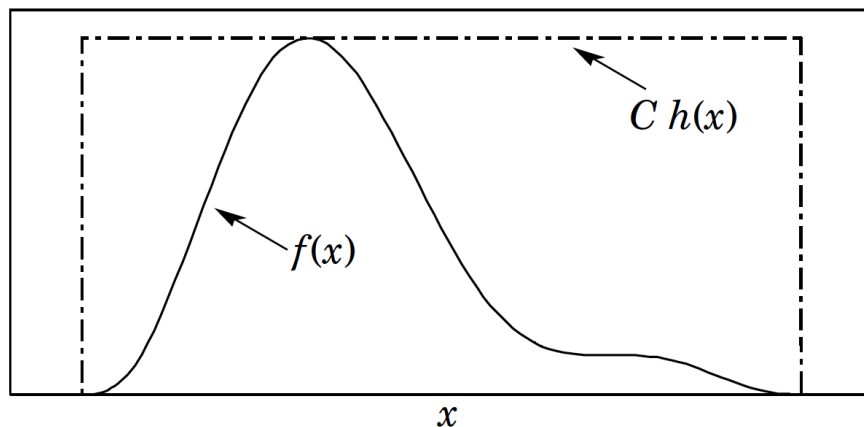
此时适合采用
舍选法。



舍选法：步骤

假设希望按照分布 $f(x)$ ($a < x < b$) 产生随机数：

- 1) 选取一个容易抽样的分布 $h(x)$ (通常为均匀分布) , 使得曲线 $Ch(x)$ 可以覆盖待抽样的分布 $f(x)$ 。
- 2) 根据 $[0,1]$ 区间均匀分布产生 r_1 , 做变换 $x = a + (b - a)r_1$ 并计算 $f(x)$ 和 $Ch(x)$ 。
- 3) 根据 $[0,1]$ 区间均匀分布产生 r_2 , 并计算 $r_2Ch(x)$ 。
- 4) 若 $r_2Ch(x) \leq f(x)$, 则保留 x ; 否则舍弃之。
- 5) 回到2) 进行重复, 直到保留的 x 达到指定数目。



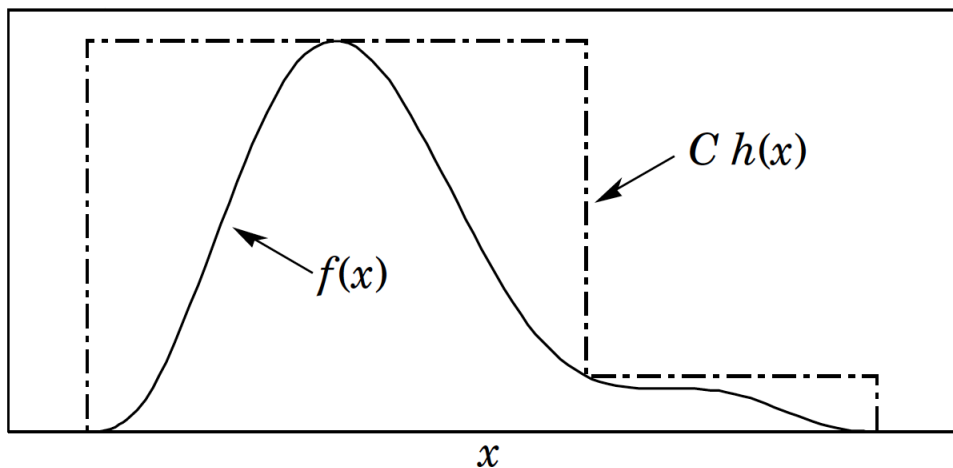
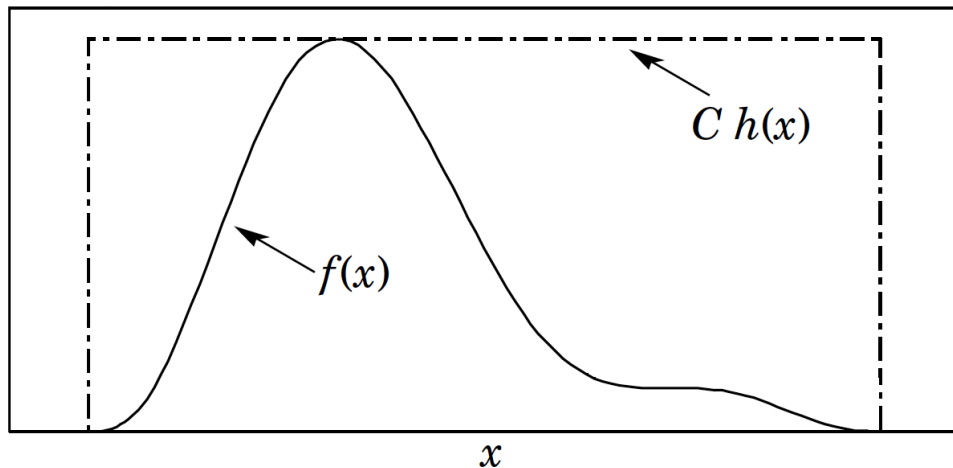
所有保留的 x 服从分布 $f(x)$ 。

舍选法：抽样效率

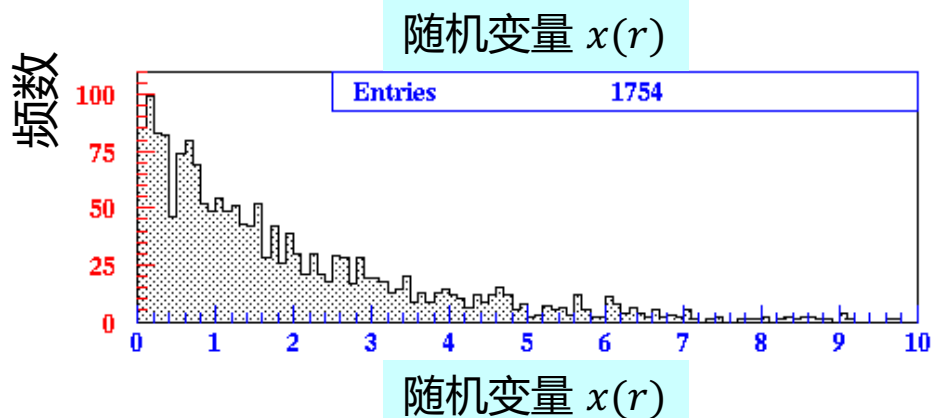
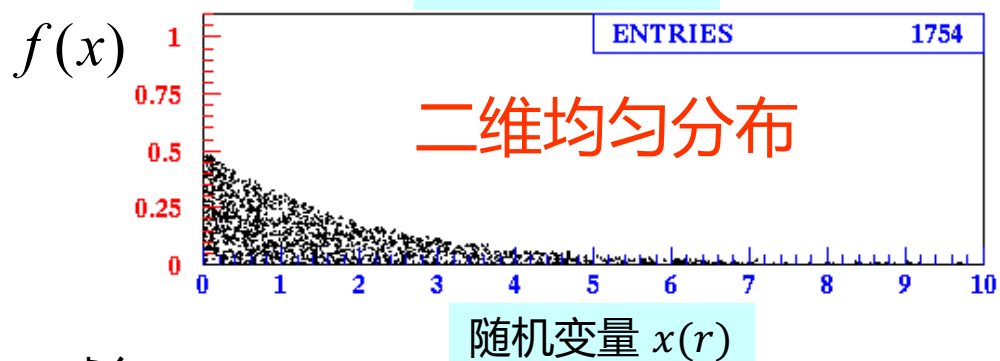
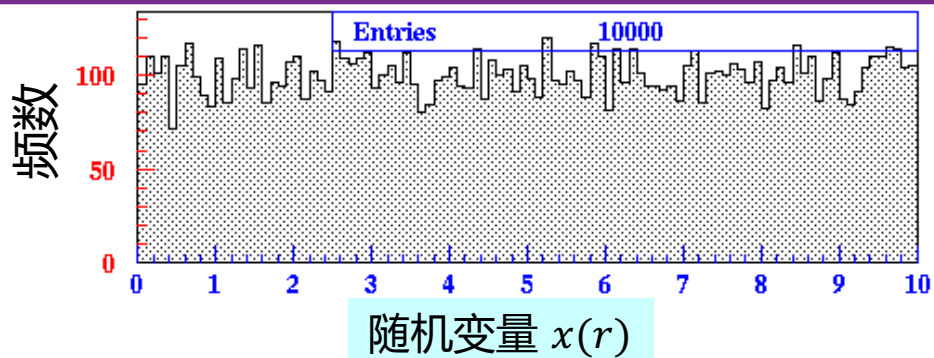
舍选法存在抽样效率问题，即，产生的 x 只有一部分被保留下来。

抽样效率等于 $1/C$ ，因为 $f(x)$ 和 $h(x)$ 都是归一化的。

为了提高抽样效率，应当使得 $Ch(x)$ 尽量靠近 $f(x)$ 的曲线。



舍选法举例



舍选法存在效率问题。

边缘分布

变换法与舍选法对比

变换法

优点：100%的抽样效率

缺点：函数须可积

舍选法

优点：方法简单，可用于非常复杂的函数

缺点：抽样效率可能会很低

常用概率密度函数建议采用的方法

(<https://pdg.lbl.gov/2022/reviews/rpp2022-rev-monte-carlo-techniques.pdf>)。

除此之外，舍选法最为常用。

- 指数分布
- 三维各向同性分布
- 二维随机角度的正余弦分布
- 高斯分布
- 自由度为 n 的 χ^2 分布
- 伽马分布
- 二项分布
- 泊松分布
- 学生氏分布

常用概率分布的抽样：ROOT

高斯分布

```
void histgaus()  
{  
  TH1F *hx = new TH1F("hx","x dis.", 100,-10,10);  
  gRandom->SetSeed();  
  Double_t x;  
  const Double_t sigma=2.0;  
  const Double_t mean=1.0;  
  const Int_t kUPDATE = 1000;  
  for ( Int_t i=0; i<kUPDATE; i++) {  
    x=gRandom->Gaus(mean,sigma);  
    hx->Fill(x); }  
}
```

可以换为

```
x = gRandom->Rndm(i);  
x = gRandom->Uniform(xup);  
x = gRandom->Integer(Imax);  
x = gRandom->Landau(mean,sigma);  
x = gRandom->Binomial(ntot,prob);  
x = gRandom->Poisson(mean);  
x = gRandom->PoissonD(mean);  
x = gRandom->Exp(tau);  
x = gRandom->BreitWigner(me,sig);
```

产生平均值为mean
标准偏差为sigma的
高斯分布。

在ROOT环境下采用已有的分布，可以轻松完成布置的练习。

初学者常犯的错误

我们在做蒙特卡罗模拟用到的几乎都是伪随机数，它的源头是数学递推公式。对于伪随机数，随机数“种子”（即随机数序列的初始化）非常重要。

给定了随机数种子，某个随机数产生子产生出来的随机数序列一定完全相同。

在需要海量随机数的时候，我们往往会将程序复制多份，利用多CPU核同时进行蒙特卡罗模拟。

初学者往往容易忘记修改复制的程序中随机数种子，从而导致每一程序模拟出来的每一套数据实际上完全相同。这种情况下，一定要为每一份程序设置不同的随机数种子。

本章要点

- 蒙特卡罗方法
- 随机数产生子
- 任意分布抽样的函数变换法和舍选法
- 蒙特卡罗模拟在粒子物理与核物理中的应用

粒子与核物理中模拟的应用

- 实验初期的设计阶段建模分析
- 了解实验可能遇到物理过程的基本特征
- 了解实验仪器自身所受到的各种影响因素与所影响的大小
- 数据分析阶段的系统分析
- ...

蒙特卡罗物理产生子

目的:

将理论用于某种物理过程的事例产生

输出量:

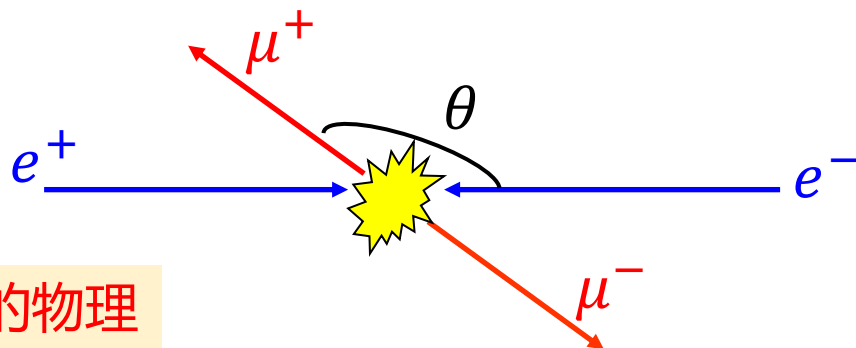
为对应某一物理过程的事例。对于每个事例，给出过程产生的末态粒子和对应的动量

在粒子物理与核物理实验数据分析中，为了验证某一理论或模型，常常需要理论家提供蒙特卡罗物理产生子。

蒙特卡罗物理产生子 (续)

简单例子：正负电子对撞机

$$e^+ e^- \rightarrow \mu^+ \mu^-$$



产生极角 θ 与方位角 ϕ 已知的物理

$$f(\cos \theta; A_{\text{FB}}) \propto \left(1 + \frac{8}{3} A_{\text{FB}} \cos \theta + \cos^2 \theta\right); \quad g(\phi) = \frac{1}{2\pi}$$

粒子物理与核物理中常用的产生子程序包

$e^+ e^- \rightarrow$ 强子

JETSET(PYTHIA)
HERWIG
ARIADNE

$pp \rightarrow$ 强子

ISAJET
PYTHIA
HERWIG

$e^+ e^- \rightarrow W^+ W^-$

KORALW
EXCALIBUR
ERATO

产生子的输出是“事例”，即每个事例包含产生的粒子列表以及每个粒子的四动量、类型等等

蒙特卡罗探测器模拟

以产生子得到的事例为输入，模拟粒子的输运过程

模拟
探测器响应

多重库仑散射 (产生散射角)
粒子衰变 (产生寿命)
电离能损 (产生能损 Δ)
电磁簇射、强子簇射
产生电子学响应 ...

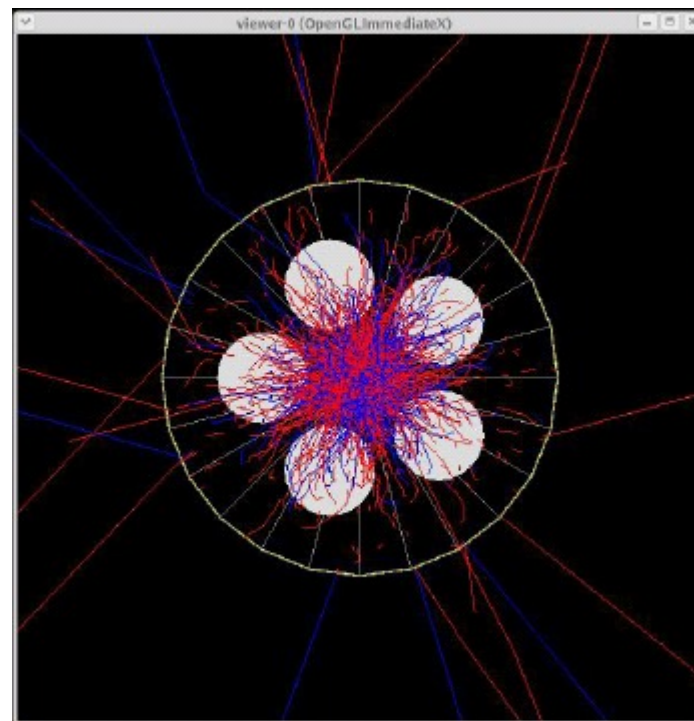
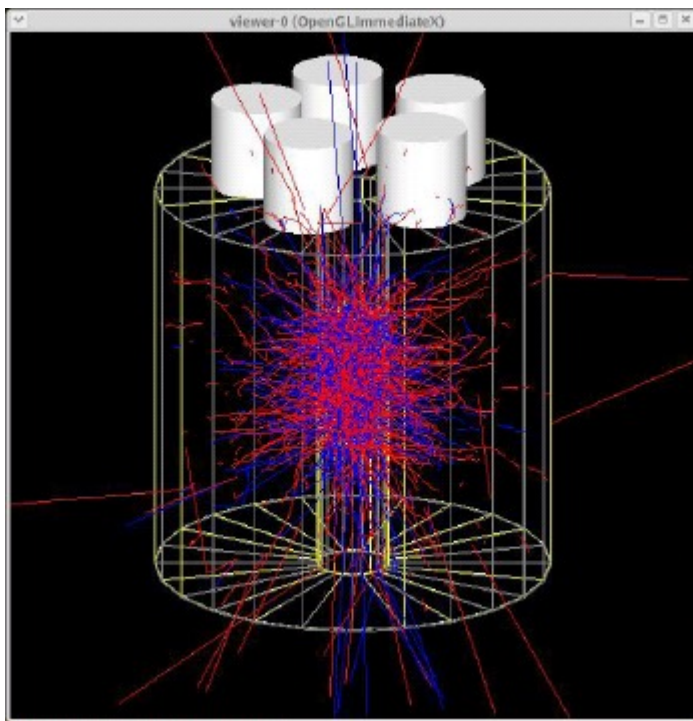
输出量 = 模拟的数据  重建分析软件的输入

用途：在给定“物理产生子层面”的某种假设下，预测在“探测器层面”应该得到的结果。

例如，估计探测效率 $\varepsilon = n_{\text{observed}}/n_{\text{generated}}$

通用软件包：GEANT4 <http://geant4.web.cern.ch/geant4>

探测器模拟的物理过程



这种模拟可以提供对探测器效率与预期性能的很好估计。

蒙特卡罗快速模拟(fast simulation)

例如：一质量为 m_0 宽度为 Γ_0 的共振态在实验上观测到的概率分布是什么形式？

贝叶斯定理：

$$P(\text{理论}|\text{实验}) = \frac{P(\text{理论}|\text{实验})}{P(\text{实验})} P(\text{理论})$$

卷积过程：

布莱特-魏格纳
分布

$$BW(M; m_0, \Gamma_0)$$

⊗

探测器
分辨率

$$R(M'|M)$$

⊗

探测
效率

$$P(\varepsilon = 100\%|M)$$

$$f(M') = BW(M; m_0, \Gamma_0) \otimes R(M'|M) \otimes P(\varepsilon = 100\%|M)$$

蒙特卡罗快速模拟(fast simulation)

对应于真实的 M ，实际观测的 M' 应该是怎样一个分布

$$f(M') = \frac{\int_{-\infty}^{+\infty} BW(M; m_0, \Gamma_0) R(M'|M) P(\varepsilon = 100\%|M) dM}{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} BW(M; m_0, \Gamma_0) R(M'|M) P(\varepsilon = 100\%|M) dM dM'}$$

假设

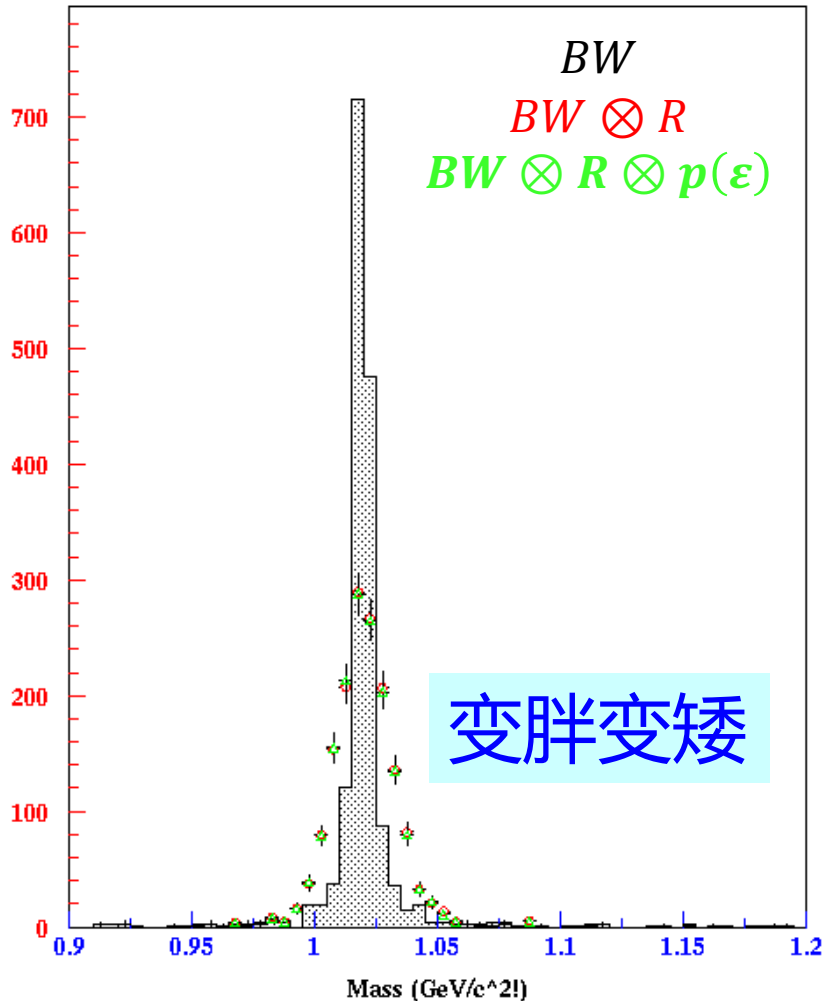
$$m_0 = 1.19456 \text{ GeV}/c^2$$

$$\Gamma_0 = 0.00426 \text{ GeV}/c^2$$

$$R(M'|M) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(M'-M)^2}{2\sigma^2}}$$
$$(\sigma = 0.01 \text{ GeV}/c^2)$$

$$P(\varepsilon = 100\%|M) = 0.9(1 - 0.1(M[\text{GeV}/c^2])^2)$$

蒙特卡罗快速模拟(fast simulation)



真实物理的图像在实验观测中会发生变化。

如果探测器的影响可以用函数来表达，有时积分可积。

但大多数情况下，不能用函数表示，蒙特卡罗方法可以给出最好的近似。

小结

➤ 蒙特卡罗方法

利用随机数对概率或与概率有关的数值计算
精度与积分维度无关，总是正比于 $1/\sqrt{n}$

➤ 随机数产生子

$[0,1]$ 均匀分布 r ，相互独立，长周期(伪随机数)

➤ 函数变换法

$$r = \int_{-\infty}^x f(x')dx' = F(x) \Rightarrow x = F^{-1}(r)$$

➤ 舍选法

$r_1 \rightarrow x; y = r_2 f_{\max}; f(x) < y$ 则保留 $x \Rightarrow x$ 服从 $f(x)$ 分布

➤ 在粒子与核物理中的应用

物理产生子与探测器模拟等

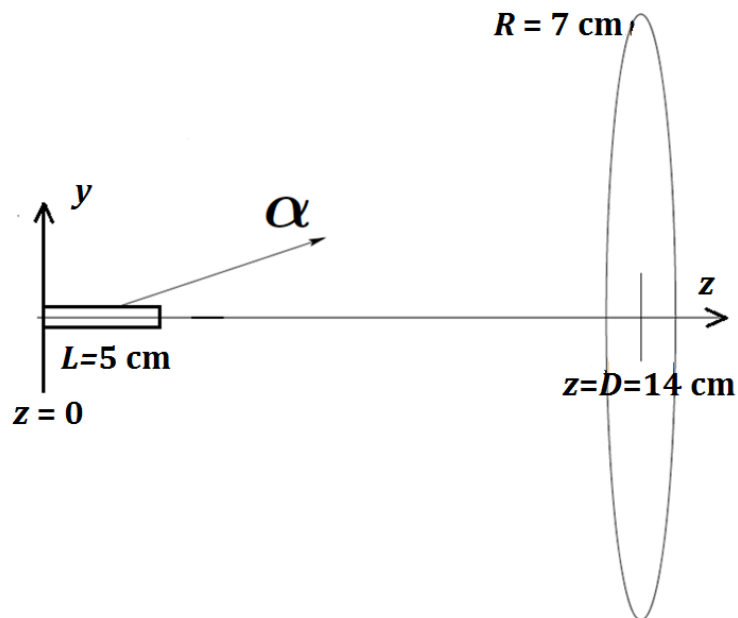
案例：探测效率的估计

有一个长度为 $L = 5 \text{ cm}$ 的细长棒状的 α 放射源，棒的左端位于原点，棒体与 z 轴平行。放射源在棒中均匀分布，发射的 α 粒子在空间中的角分布各向同性。

在 z 轴正向距离原点 $D = 14 \text{ cm}$ 处放置了一个圆盘状的探测器用来记录 α 粒子，圆盘半径 $R = 7 \text{ cm}$ ，轴线与 z 轴重合。

求探测器的几何接受效率，
即 α 粒子有多大概率可以
入射到探测器中。

(忽略细棒的半径。)



案例：探测效率的估计（分析）

- 1) 放射棒均匀意味着在 $z \in (0, 5)$ cm 范围内任意位置发射 α 粒子的概率相同。
- 2) 发射的 α 粒子空间角分布均匀，意味着 α 粒子的运动方向在三维空间均匀分布，即 $\cos \theta \sim U(-1, 1)$, $\phi \sim U(0, 2\pi)$ 。

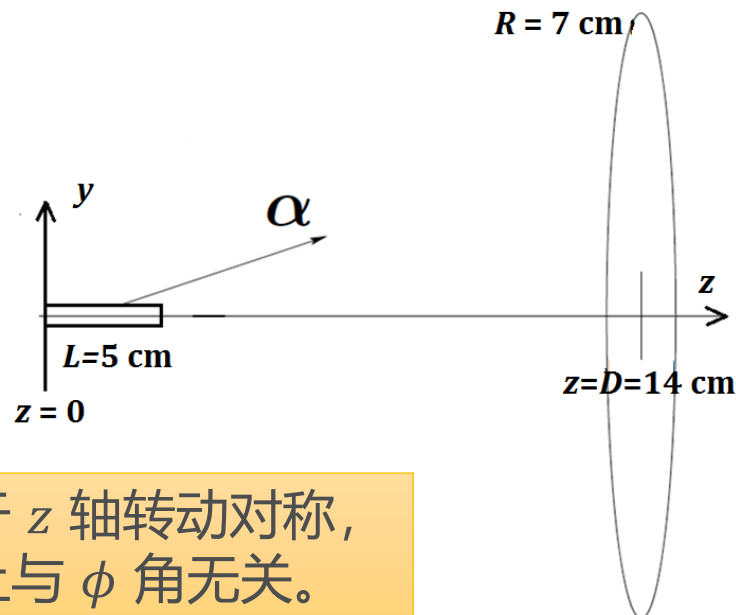
极角 θ : α 粒子运动方向与 z 轴的夹角

方位角 ϕ : 运动方向在 $x - y$ 平面上的投影与 x 轴的夹角

- 3) α 粒子能否入射到探测器中与 ϕ 无关，只与发射位置 z 和极角 θ 有关。

需要满足

$$\cos \theta > \frac{D - z}{\sqrt{R^2 + (D - z)^2}}$$



该问题关于 z 轴转动对称，因此实际上与 ϕ 角无关。

案例：探测效率的估计（分析）

1) 放射棒均匀意味着 α 粒子的发射位置 $z \sim U(0, 5 \text{ cm})$:

$$f(z) = 1/5, \quad 0 < z < 5 \text{ cm}$$

2) α 粒子的运动方向可用角度 (θ, ϕ) 表示, 角分布均匀意味着:

$$g(C) = 1/2, \quad -1 < C < 1, \quad C \equiv \cos\theta$$

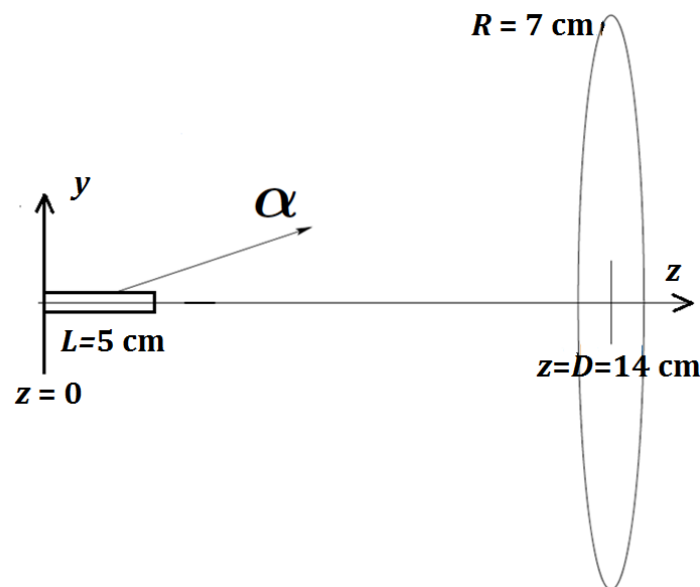
$$h(\phi) = 1/(2\pi), \quad 0 < \phi < 2\pi$$

3) α 粒子能否入射到探测器中只与 z 和 θ 有关, 要求:

$$C > \frac{D-z}{\sqrt{R^2+(D-z)^2}} \equiv C_{\min}$$

4) 解析解

$$\begin{aligned} P_A &= \int_0^{2\pi} d\phi \int_0^5 dz \int_{C_{\min}}^1 dC f(z)g(C)h(\phi) \\ &= \frac{1}{10} \int_0^5 dz \left(1 - \frac{D-z}{\sqrt{R^2+(D-z)^2}} \right) = 0.0749278 \end{aligned}$$



案例：探测效率的估计（蒙特卡罗求解）

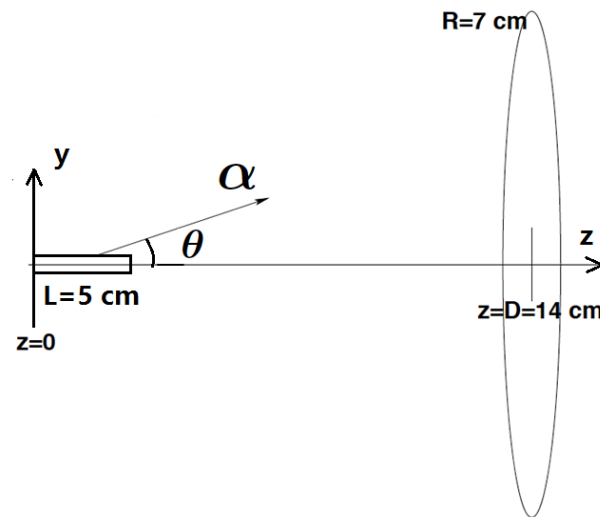
1) 根据分布 $f(z)$ 产生 α 粒子的发射位置 z ;

2) 根据分布 $g(C)$ 产生 $C \equiv \cos\theta$

3) 判断 C 和 z 是否满足

$$C > \frac{D-z}{\sqrt{R^2+(D-z)^2}}$$

如果满足，记录为“成功”；否则记录为“失败”。



模拟 N 个事例，假设“成功”事例数为 n ，接受度效率近似为

$$P_A = \frac{n}{N} \pm \frac{\sigma(n)}{N}$$

其中 n 服从 $p \approx \frac{n}{N}$ 的二项分布，标准差为 $\sigma(n) = \sqrt{Np(1-p)}$

模拟 $N = 10^6$ 个事例，结果为 $P_A = 0.07496 \pm 0.00026$
(对比解析结果 0.0749278)

更真实（复杂）的案例

假设利用加速器产生了从原点出发沿 z 轴正向运动的单能 K_S 粒子，能量 $E_{K_S} = M_{K_S}^2 c^2 / 2m_\pi$ 。 K_S 粒子平均寿命为 τ ，在实验室系飞行一段距离后衰变成 $\pi^+ \pi^-$ 粒子对。在 K_S 质心系中， π^\pm 的角分布各向同性。粒子束流前放置了一个圆盘状的探测器以记录末态粒子 π^\pm ，圆盘半径 $R = 7\text{ cm}$ ，轴线与 z 轴重合，距离原点 $D = 14\text{ cm}$ 。

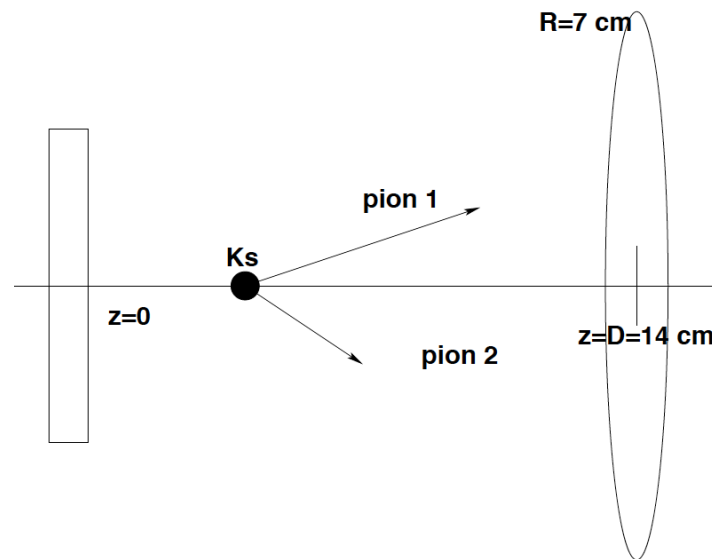
末态粒子对 $\pi^+ \pi^-$ 同时击中探测器则表明探测到了 K_S 粒子的衰变。求探测器的接受效率。

(已知质量 $M_{K_S} = 0.498\text{ GeV}/c^2$

$M_{\pi^\pm} = 0.140\text{ GeV}/c^2$

寿命 $\tau = 8.954 \times 10^{-11}\text{ s}$

光速 $c = 3 \times 10^8\text{ m/s}$)



更真实（复杂）的案例：分析

这个案例由于特殊设定的能量 $E_{K_S} = M_{K_S}^2 c^2 / 2m_\pi$ ，使得其可以解析求解。
结果为 $P_A = 0.203574$ （过程从略）。

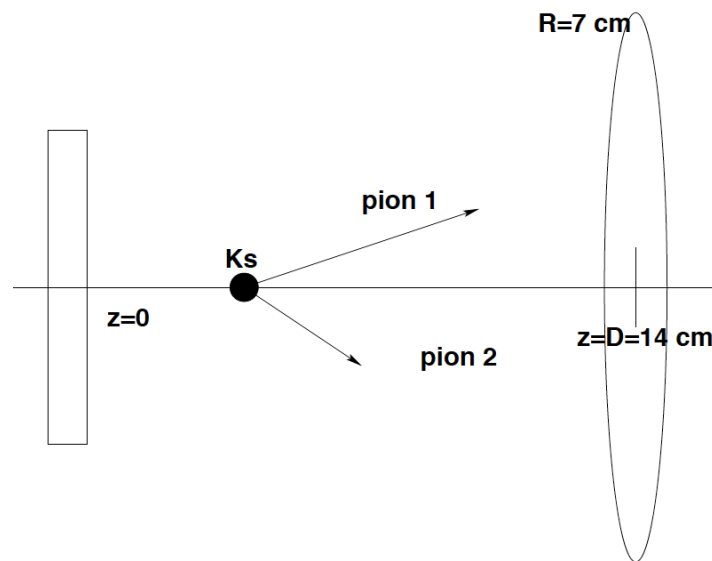
用蒙特卡罗直接模拟求解要简单得多：

- 1) 按照指数分布 $f(z)$ 产生衰变位置 z 。
- 2) 在 K_S 静止系中产生 π^\pm 的运动方向： $\cos\theta$ 均匀分布，且 π^+ 与 π^- 方向背对背。
- 3) 在 z 方向做洛伦兹变换得到实验室系下运动方向。
- 4) 判断 π^\pm 是否能入射到探测器内，统计“成功”的次数。

$N = 10^6$ 次模拟， $n = 203573$ 次成功。

$$P_A = 0.20357 \pm 0.00040$$

（对比解析结果 0.203574）。



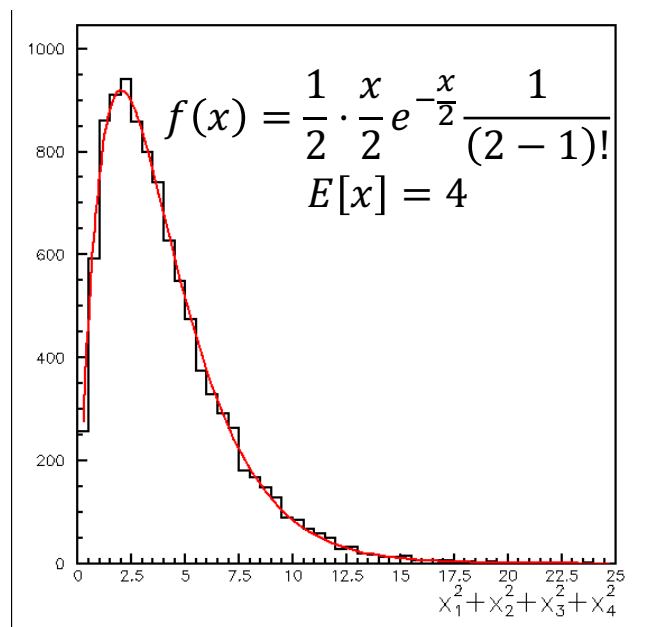
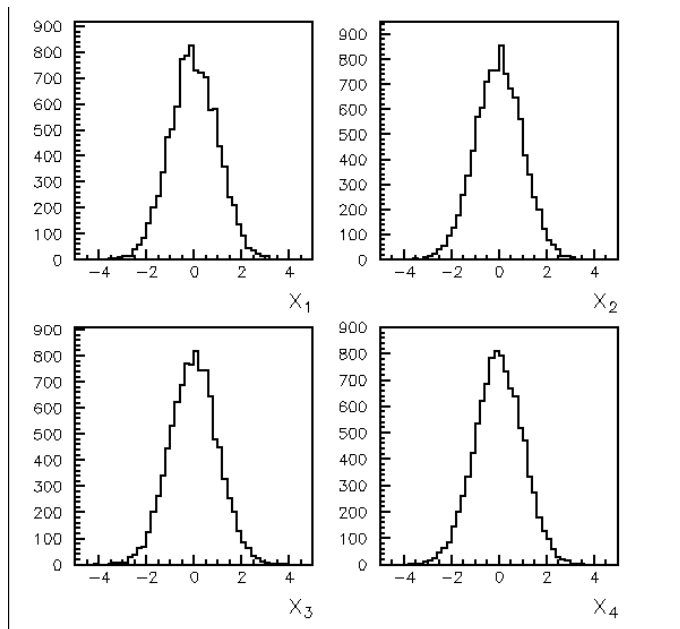
蒙特卡罗统计检验

例如，检验理论与实验符合好坏的 χ^2 分布。

4个服从标准正态分布且相互独立的随机变量的平方和



服从 $\chi^2(4)$ 分布



思考：如果出现不符合的情况，该如何解释？