

粒子物理与核物理实验中的数据分析

杨振伟
清华大学

第四讲:蒙特卡罗方法

上一讲回顾

概率的基本概念

随机变量与概率密度函数

随机变量的平均值与方差

能不通过实验对随机变量进行研究吗？

本讲要点

- 蒙特卡罗方法
- 随机数产生子
- 任意分布抽样之函数变换法与舍选法
- 蒙特卡罗方法中的精度问题
- 在粒子物理与核物理中的应用

蒙特卡罗方法简介

蒙特卡罗方法就是利用一系列随机数来计算各种概率大小和随机变量均值等等的数值分析技术。通常的步骤为：

- 1) 产生一系列在 $[0, 1]$ 之间均匀分布的随机数 r_1, r_2, \dots, r_m
- 2) 利用这些随机数按某些概率密度函数 $f(x)$ 抽样生成我们感兴趣的另一随机序列 x_1, x_2, \dots, x_n 。
- 3) 利用这些 x 值来估计 $f(x)$ 的一些特性，例如：通过找到在区间 $a \leq x \leq b$ 的 x 比例，给出积分值 $\int_a^b f(x)dx$ 。

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”

随机数的产生

用物理方法产生
真正的随机数



- 不可重复
- 产生速度慢

用数学方法产生
伪随机数



- 可以重复
- 产生的速度快

真随机数与伪随机数

美国兰德（**RAND**）公司在**1950**年代，利用真空管中产生的噪音制作了一个含十万个真正的随机数表，并运用于其开展的所有模拟研究中。

真正的随机数与伪随机数之间的区别在于：数据串是否具有可压缩性，即能否用更短的形式来表示。

真正的随机数是不可压缩的，非常不规则，以至于无法用更短的形式来表示它。

在粒子物理与核物理研究中，随机数的可重复性经常也是非常有用的，尤其是程序的调试(**debugging**)。

随机数产生子

目的是使在 $[0, 1]$ 范围内产生的伪随机数满足：

均匀性；相互独立性；长周期性

乘同余法

$\xi_i \equiv \text{余数}(\lambda \xi_{i-1} / M) \Rightarrow$ 随机数 $r_i = \xi_i M^{-1}$
 λ 与 M 为选定常数, ξ_0 为随机数种子。

友情推荐

$M=2^K$	$\lambda=5^{2q+1}$	ξ_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	$2^{40} \approx 10^{12}$

CERN库的随机数产生子

PAW用户

...

Real random(1)

Call Rmarin(ISEED,0,0)

...

Call Ranmar(random,1)

...

Root 用户

...

gRandom->SetSeed();

...

Float_t random = gRandom->Rndm(1);

...

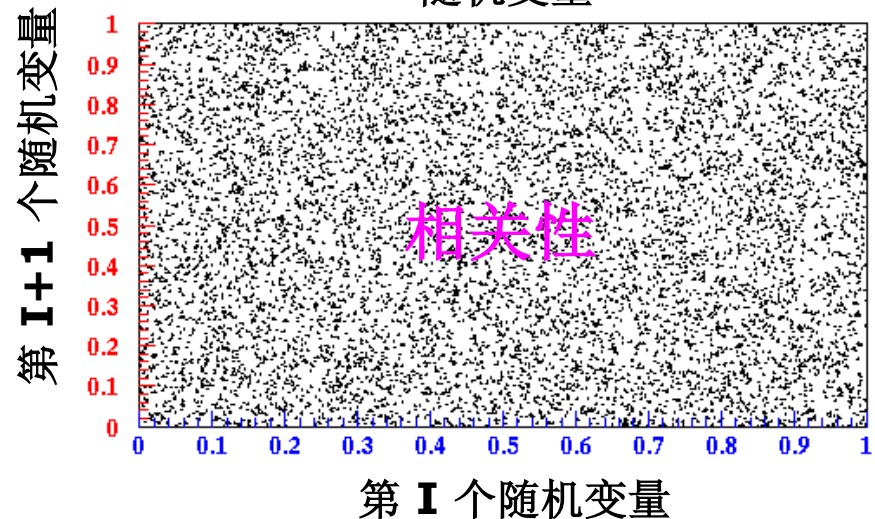
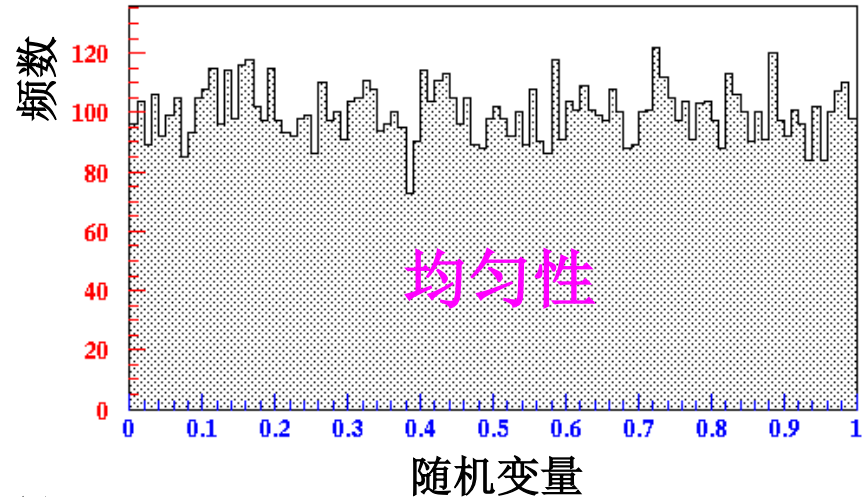
注意：用于产生子的随机数种子还可以用来保证后续进程的随机数不重复。

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。

随机数均匀性与相关性检验

```
subroutine mc
double precision lamda,M,x,x0,y
call hbook1(10,'r',100,0.,1.,0.)
call hbook2(20,'r(i+1) vs. r(i)',
&100,0.,1.,100,0.,1.,0.)
x0=1.
lamda=1220703125 ! 5**13
M=4294967296.    ! 2**32
do i=1,10000
  x=Mod(lamda*x0,M)
  y=x/M
  call hfill(10,real(y),0.,1.0)
  if(i.gt.1)call
& hfill(20,real(y_old),real(y),1.0)
  x0=x
  y_old=y
end do
return
end
```

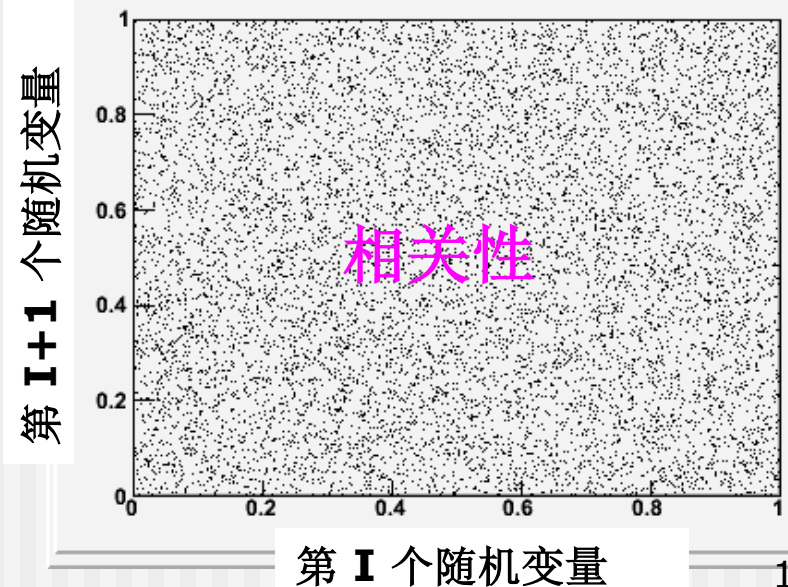
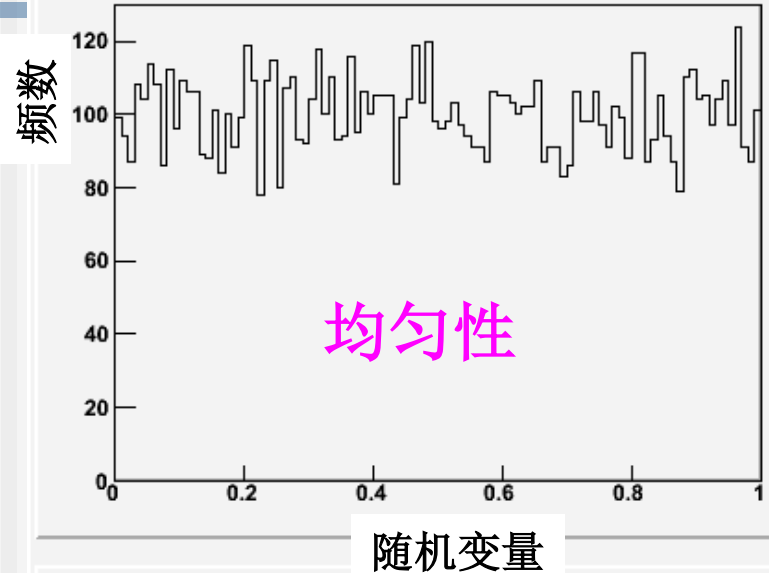
PAW > call mc.f
PAW > zone 1 2; h/pl 10; h/pl 20



随机数均匀性与相关性检验

```
void random() {  
    UInt_t lambda,M,x0;  
    TH1F *h1 = new TH1F("h1","",100,0,1);  
    TH2F *h2 = new TH2F("h2","",100,0,1,100,0,1);  
  
    lambda=1220703125; //5^13  
    M      =4294967296; //2^32  
    x0      =1;  
    double y, y_old;  
    for (int i=0; i<10000; i++) {  
        x = (lambda*x0)%M;  
        y = (double)x/M;  
        h1->Fill(y);  
        if (i>1) h2->Fill(y_old,y);  
        x0=x;  
        y_old=y;  
    }  
}
```

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



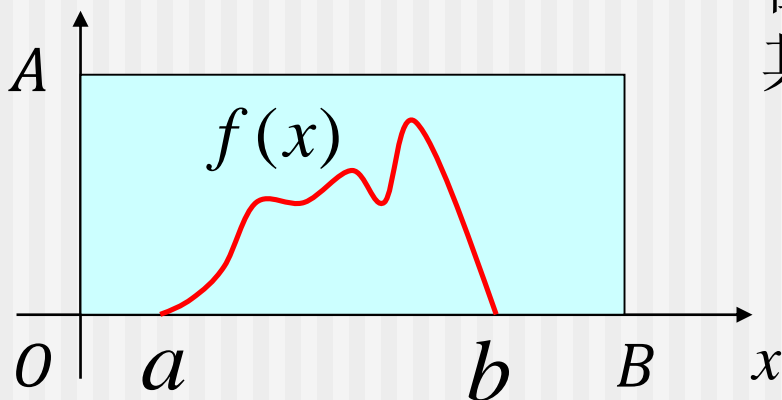
用蒙特卡罗法计算积分

对于计算积分值 $\int_a^b f(x)dx$

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

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

蒙特卡罗方法:



在AB区间均匀投总数为 N 个点，其中 n 个位于 $f(x)$ 与 x 轴所围区域。

$$\int_a^b f(x)dx \approx (AO) \cdot (BO) \cdot \frac{n}{N}$$

对函数是否解析可积和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:

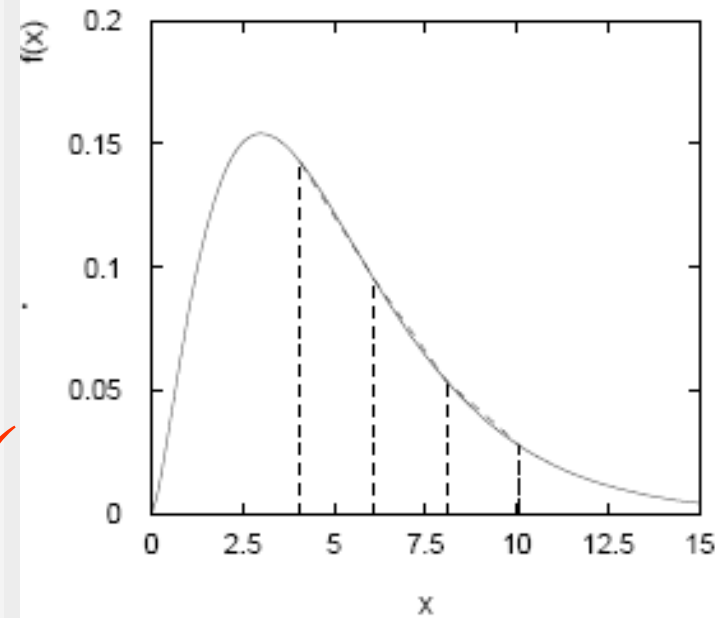
MC 精度: $\propto 1/\sqrt{n}$, (n 为产生的随机数)

梯形法精度: $\propto 1/n^2$, (n 为子区间的数目) ✓

多维积分:

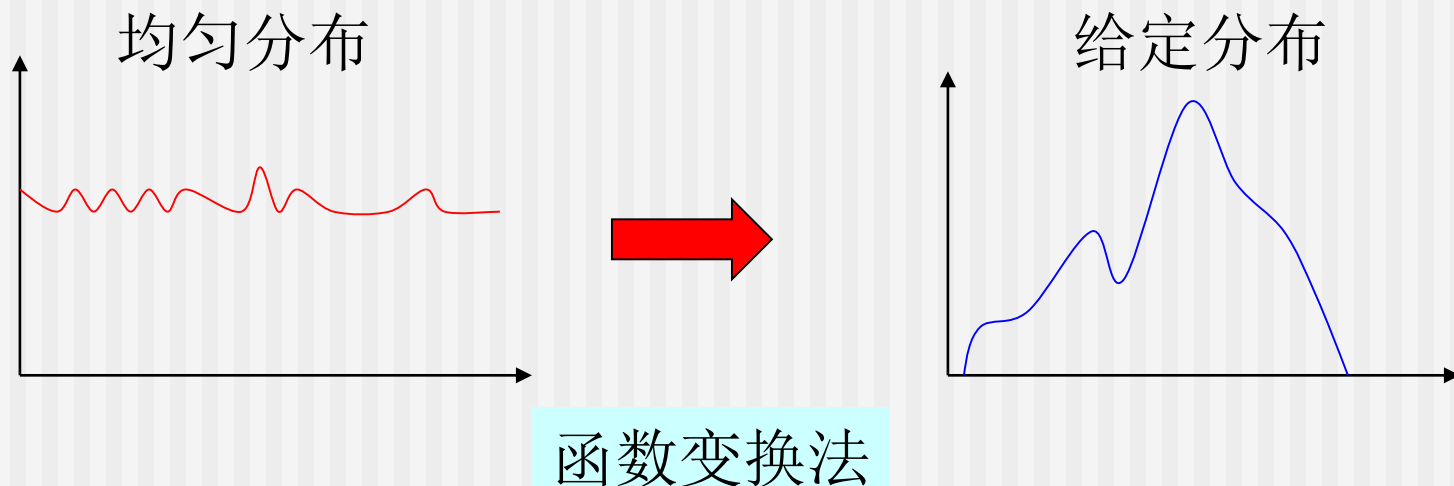
MC 精度: $\propto 1/\sqrt{n}$ 且与维数无关, (n 为产生的随机数) ✓

梯形法精度: $\propto 1/n^{2/\text{维数}}$, (n 为子区间的数目)



对于维数大于4的积分, 用蒙特卡罗方计算积分总是最好。

从均匀分布到任意分布的随机数



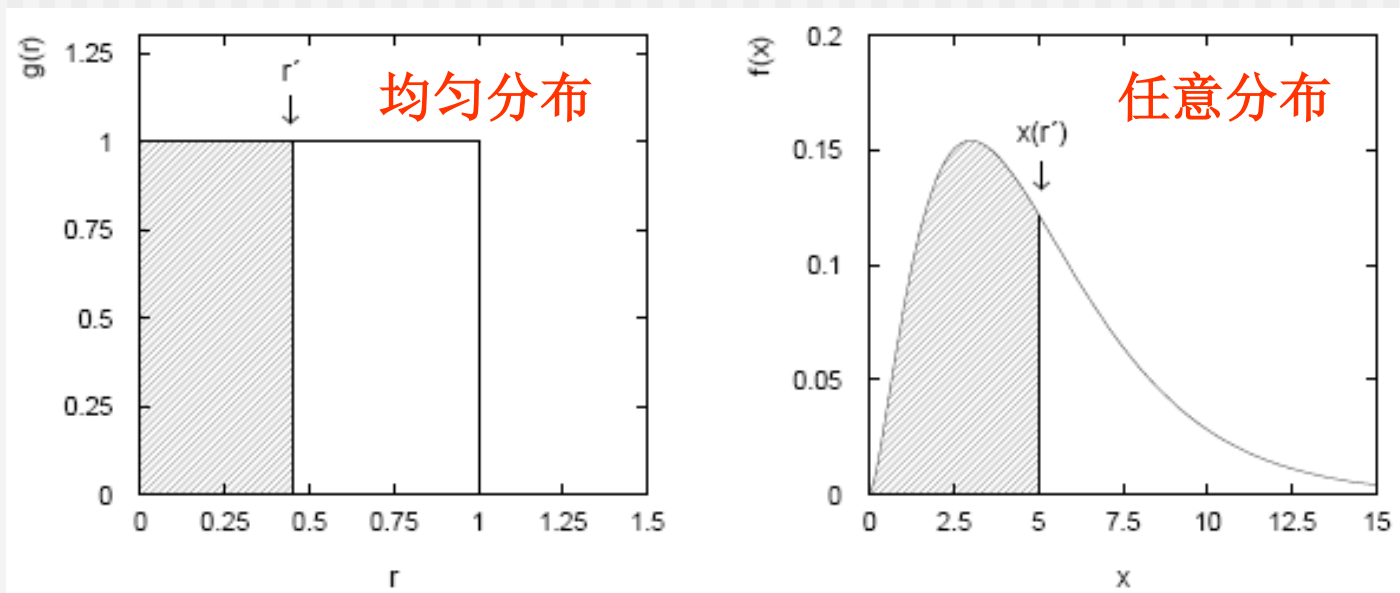
寻找某个函数，当函数的自变量取均匀分布值时，对应的函数值自动满足给定分布。

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。

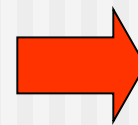
函数变换法

从在 $[0, 1]$ 均匀分布的随机数, 通过适当的变换 $x(r)$ 找出服从 $f(x)$ 分布的随机数 x_1, x_2, \dots, x_n



要求: $P(r \leq r') = P(x \leq x(r'))$,

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



令 $F(x) = r$

解出 $x = F^{-1}(r)$

例子:指数分布抽样

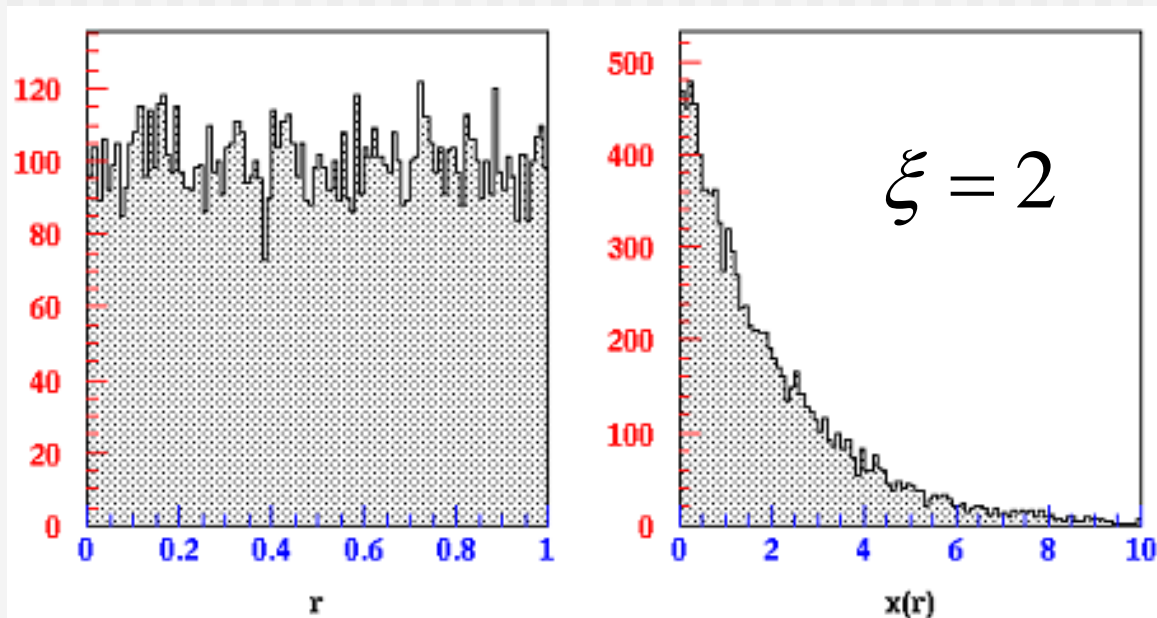
指数概率密度函数: $f(x, \xi) = \frac{1}{\xi} e^{-x/\xi} \quad (x \geq 0)$

令 $\int_0^{x(r)} \frac{1}{\xi} e^{-x'/\xi} dx' = r$ 并解出 $x(r)$, 得到

$$x(r) = -\xi \log(1-r)$$

随机变量 r 与 $x(r)$
呈一一对应关系

**抽样效率
为100%。**



可采用函数变换法抽样的分布

- 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
- Student 分布

(<http://pdg.lbl.gov/2017/reviews/rpp2017-rev-monte-carlo-techniques.pdf>)

舍选法

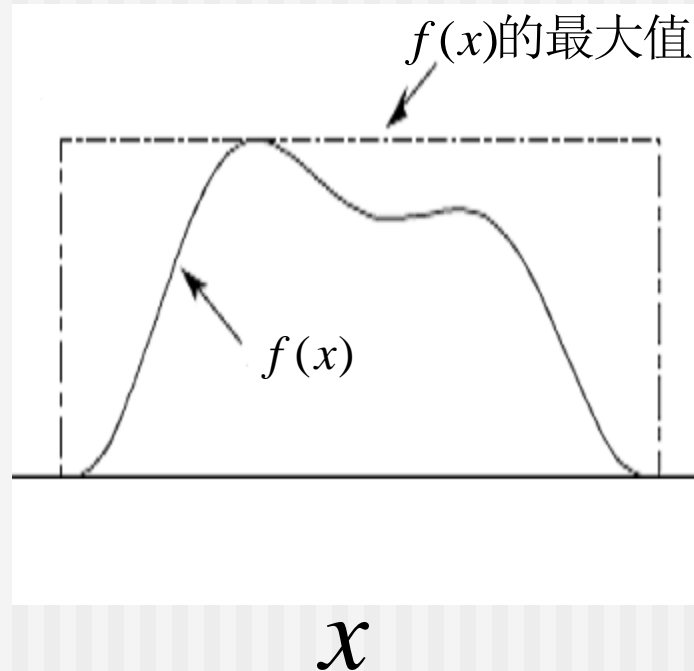
根据概率密度函数 $f(x)$ 的自变量 x 取值范围,由第一个随机变量产生均匀分布的 x ,即: $x = x_{\min} + r_1(x_{\max} - x_{\min})$



产生第二个独立的随机变量,均匀分布在 0 与 f_{\max} 范围内,即 $u = r_2 f_{\max}$



如果 $u < f(x)$,则接受该 x 值,否则,拒绝该 x 值, 从新进行抽样。



问题: 如何找到函数的最大值?

舍选法举例

```
subroutine acc_rej
real rvec(1)
call hbook1(10,'x(r)',100,0.,10.,0.)
call hbook1(20,'x(r)',100,0.,10.,0.)
call hbook2(30,'f(x) vs. x(r)',100,0.,10.,100,0.,1.1,0.)
fmax=-999.
do i=1,100
    call ranmar(rvec,1)
    r=0+rvec(1)*(10.-0.)
    f=0.5*exp(-r/2.)
    if(fmax.lt.f)fmax=f
end do
fmax=1.2*fmax
ntot=0
do i=1,10000
    call ranmar(rvec,1)
    r=0+rvec(1)*(10.-0.)
    z=0.5*exp(-r/2.)
```

找 $f(x)$ 的最大值

PAW > call acc_rej.f

PAW > zone 1 3; h/pl 10; h/pl 20; h/pl 30

```
if(z.gt.fmax)then
    fmax=z*1.2
    write(6,*)'z greater than fmax'
end if
call hfill(10,r,0.,1.0)
call ranmar(rvec,1)
u=rvec(1)*fmax
if(u.lt.z)then
    call hfill(20,r,0.,1.0)
    call hfill(30,r,u,1.0)
    ntot=ntot+1
end if
end do
write(6,*)'ntot=',ntot
return
end
```

舍选法举例

```
void acc_rej() {  
    TH1F *h11 = new TH1F("h11","",100,0,10);  
    TH1F *h12 = new TH1F("h12","",100,0,10);  
    TH2F *h2 = new TH2F("h2","",100,0,10,100,0,1);  
    double fmax=-999.;  
    for (int i=0;i<100;i++) {  
        double r = gRandom->Uniform(0,10);  
        double f = 0.5*exp(-r/2.);  
        if (fmax < f) fmax=f;  
    }  
    fmax *= 1.2;  
    cout << "fmax=" << fmax << endl;
```

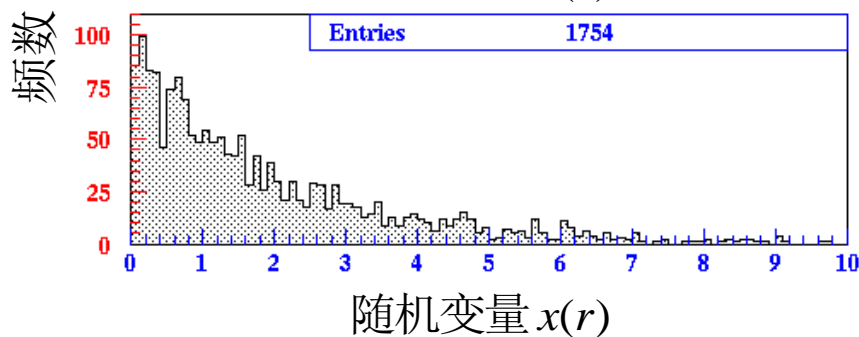
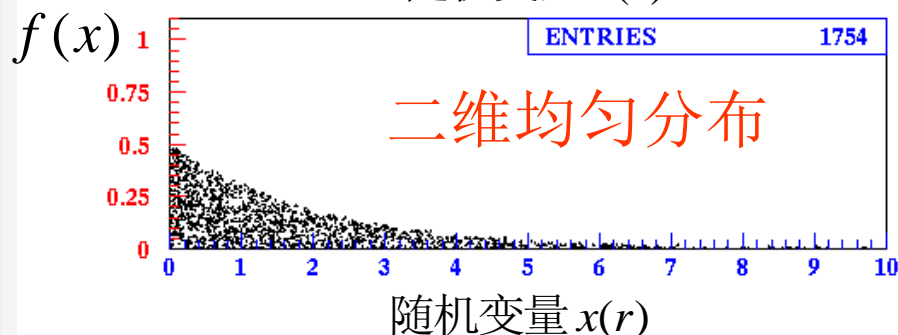
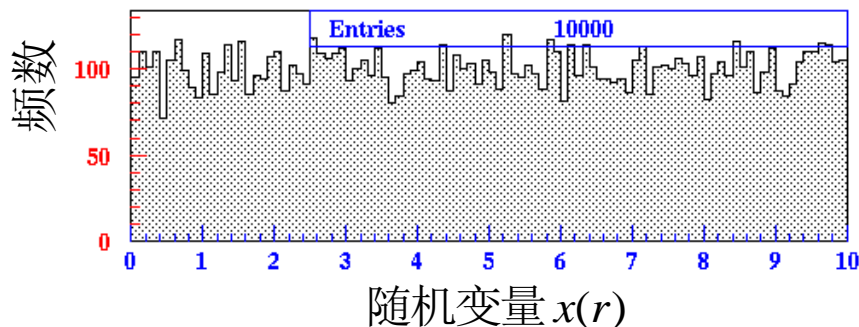
找 $f(x)$ 的最大值

```
root[0] .x acc_rej.C  
root[1] h11- > Draw();  
root[2] h12- > Draw();  
root[3] h13- > Draw();
```

ROOT脚本

```
int ntot=0;  
for (int i=0;i<10000;i++) {  
    double r=gRandom->Uniform(0,10);  
    double z=0.5*exp(-r/2.);  
    if (z>fmax) fmax=1.2*z;  
    h11->Fill(r);  
    double u=gRandom->Uniform(0,fmax);  
    if (u<z) {  
        h12->Fill(r);  
        h2->Fill(r,u);  
        ntot += 1;  
    }  
}  
cout << "ntot=" << ntot << endl;  
}
```

舍选法举例(续)



舍选法存在
效率问题。

边缘分布

函数变换法与舍选法

函数变换法

优点: **100%**的抽样效率

缺点: 函数须解析可积

舍选法

优点: 方法简单, 可用于非常复杂的函数

缺点: 需要估计函数最大值, 而且抽样效率低

粒子物理与核物理中, 对常用的概率密度函数有各种建议采用的方法(见<http://pdg.lbl.gov/2008/reviews/monterpp.pdf>)。除此之外, 舍选法最为常用。

初学者常犯的错误: 对同一个过程做计算机模拟批处理, 没有考虑在批处理结果中存在随机数的重复性。

常用概率密度分布函数的抽样

高斯(正态)分布

```
{ gROOT->Reset();  
  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(lmax);  
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。

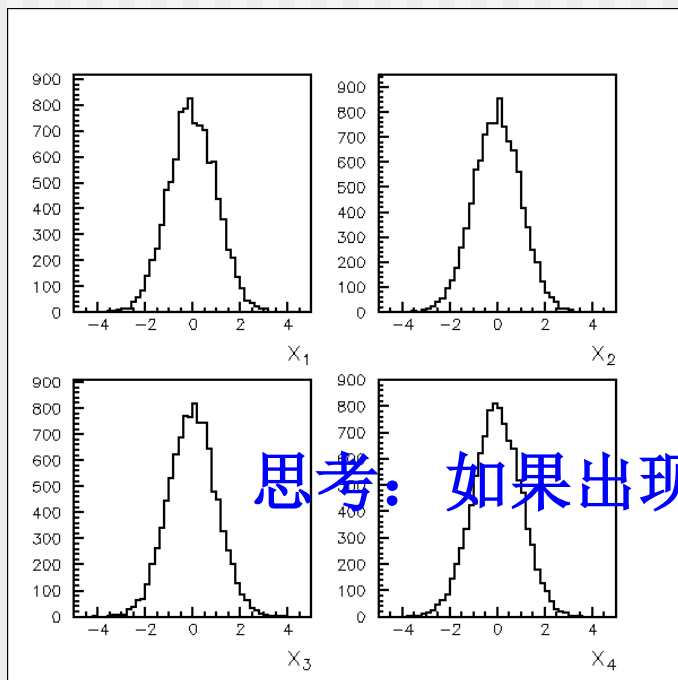
在ROOT环境下采用已有的分布，可以容易完成布置的练习。

蒙特卡罗统计检验

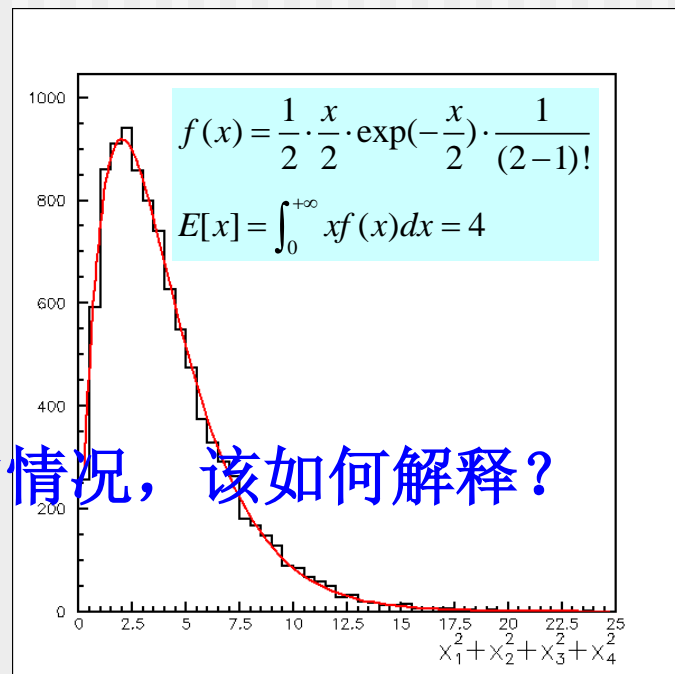
例如常用来检验理论与实验符合好坏的 χ^2 分布。

四个服从 $N(0,1)$ 正态分布的且相互独立的随机变量平方和

一定符合自由度为 4 的 χ^2 分布

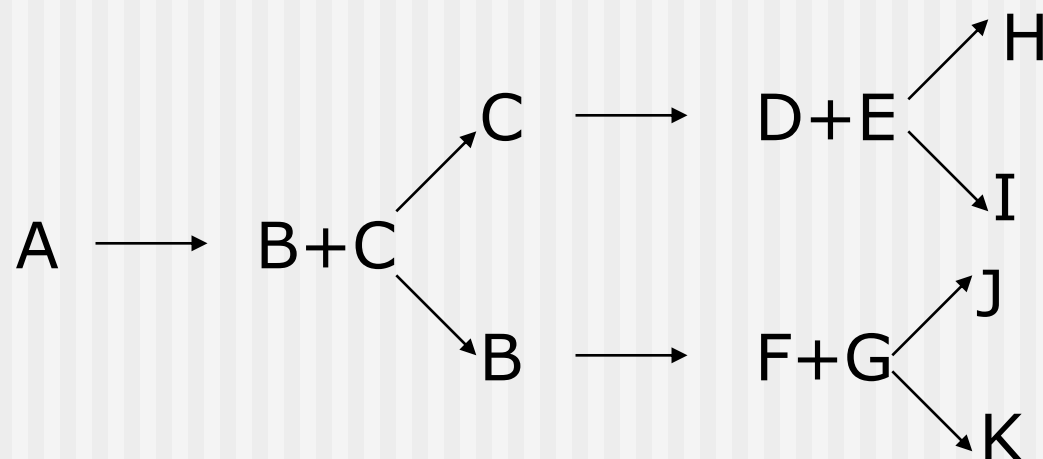


思考：如果出现不符合的情况，该如何解释？



Toy 蒙特卡罗方法

粒子物理与核物理在实验的早期设计阶段，通常利用Toy蒙特卡罗来估计可达到的测量精度（也称**黑盒子方法**）。



末态有D, F, H, I, J, K。研究测量**E**质量时实验可以达到的分辨率。

在不做探测器模拟的情况下，可以对稳定的末态粒子动量各分量进行含高斯分辨率的抽样，能损大小进行朗道分布抽样，寿命进行指数分布抽样，等等，然后在所有末态中寻找中间不稳定态**E**，根据能动量关系计算其对应的质量，得到的质量分布称为**Toy 蒙特卡罗结果**。

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。

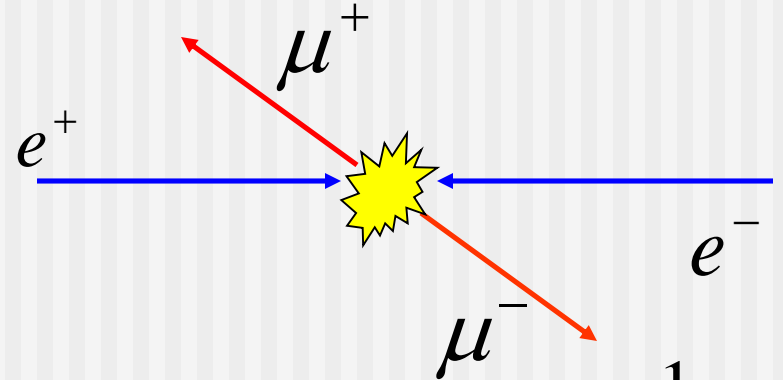
蒙特卡罗物理产生子(续)

简单情形

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

产生 θ 与 ϕ

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



粒子物理与核物理中常用的产生子程序包

$$e^+ e^- \rightarrow \text{强子}$$

JETSET(PYTHIA)
HERWIG
ARIADNE

$$pp \rightarrow \text{强子}$$

ISAJET
PYTHIA
HERWIG

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

KORALW
EXCALIBUR
ERATO

蒙特卡罗探测器模拟

从产生子中输入粒子种类与动量，然后模拟粒子的输运过程

模拟探测器响应 { 多重散射(产生散射角)
粒子衰变(产生寿命)
电离能损(产生能损)
电磁与强子簇射
产生信号，电子学响应
...

输出量 = 模拟的数据  输入重建分析软件

用途: 预测 “物理产生子层面上的” 给定假设在 “探测器层面上” 应该观测到的响应。

通用软件包: **GEANT3(FORTRAN)**, **GEANT4(C++)**

粒子与核物理中模拟的应用

- 用于实验初期的设计阶段建模分析
- 用于了解实验可能遇到物理过程的基本特征
- 用于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
- 用于数据分析阶段的系统分析
- ...

带电粒子在水中的输运过程模拟

给定带电粒子的四动量



单位厘米产生多少光子？



从均匀分布中产生满足一定波长分布的光子



沿朗伯-科夫光锥方向均匀给所有光子动量



每个光子开始在水中传播

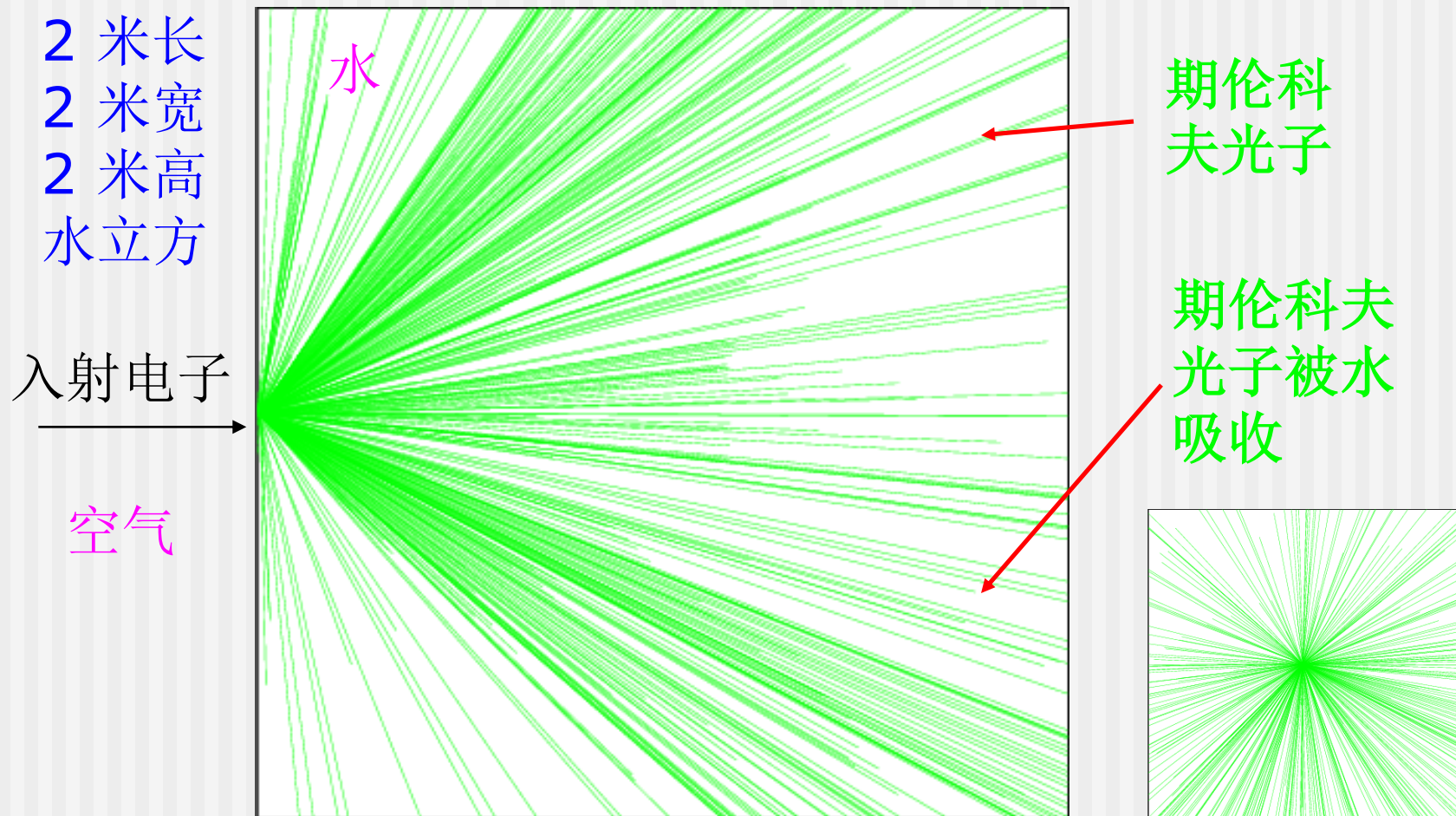


按光与水分子发生作用的概率抽样该光子是否被吸收或散射



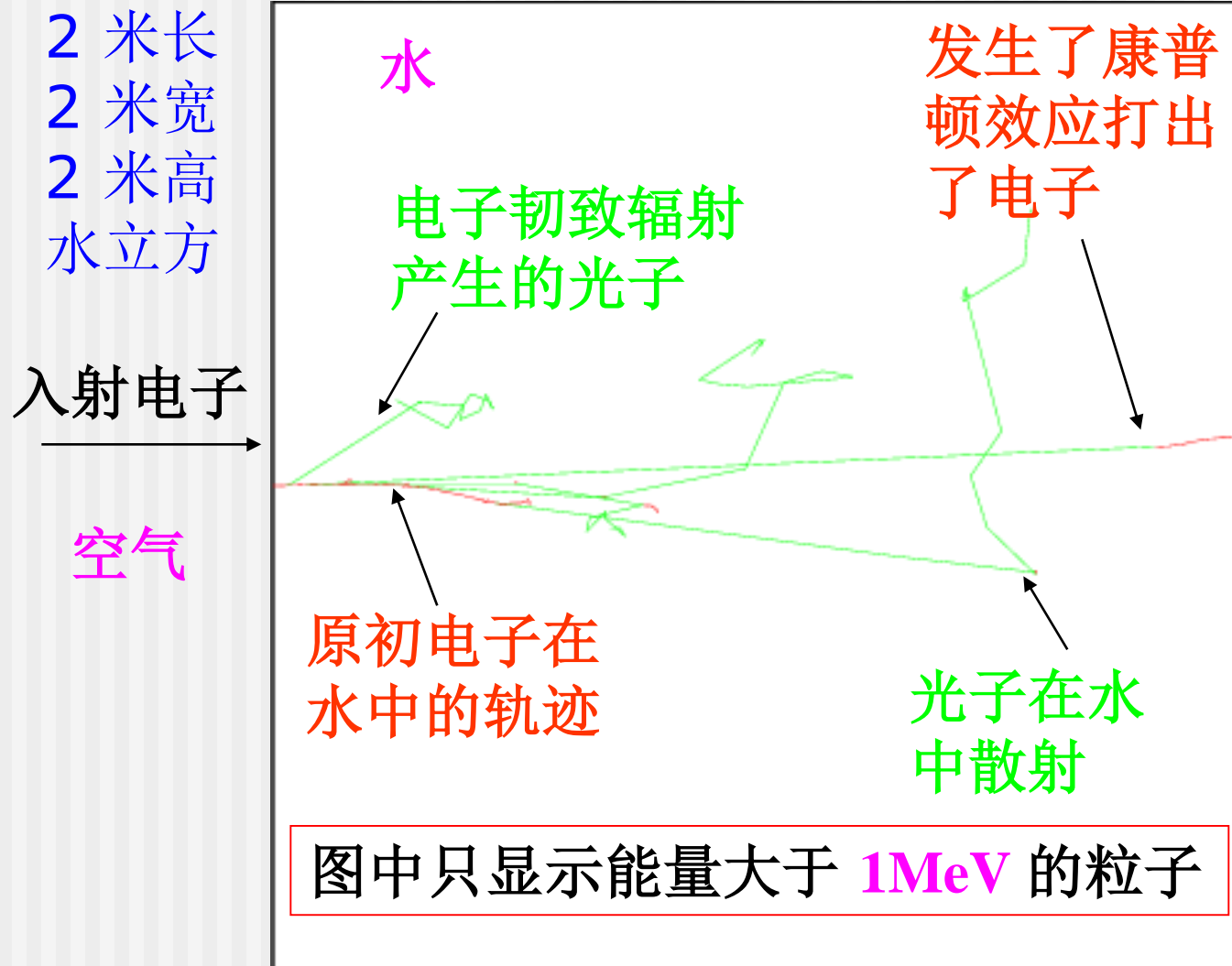
...

2 MeV 电子在水中的输运过程

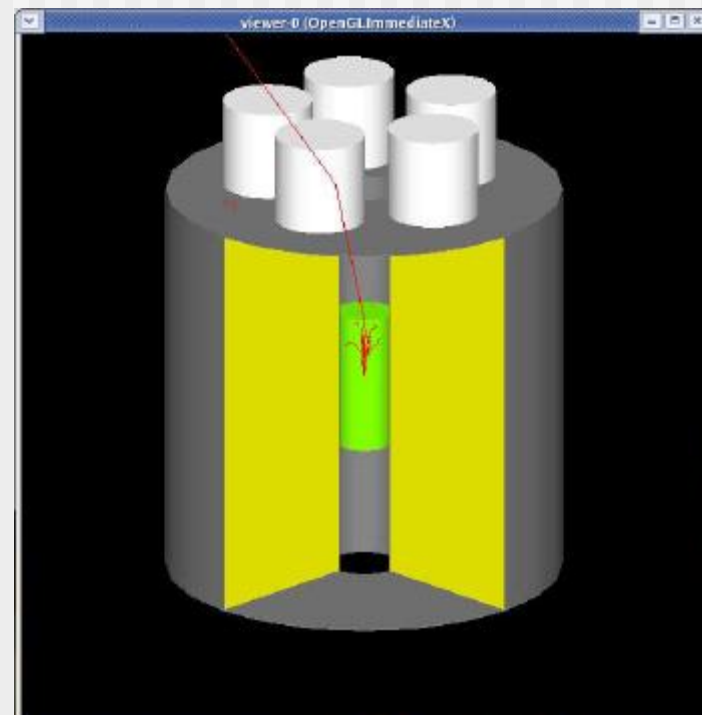


模拟结果显示了电子在水中发出期伦科夫光，损失能量直至被停止在水中的过程。

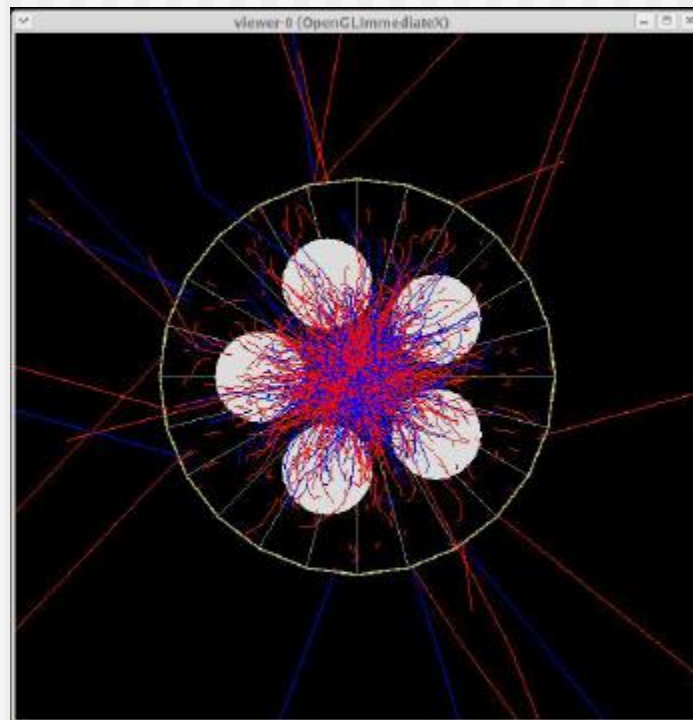
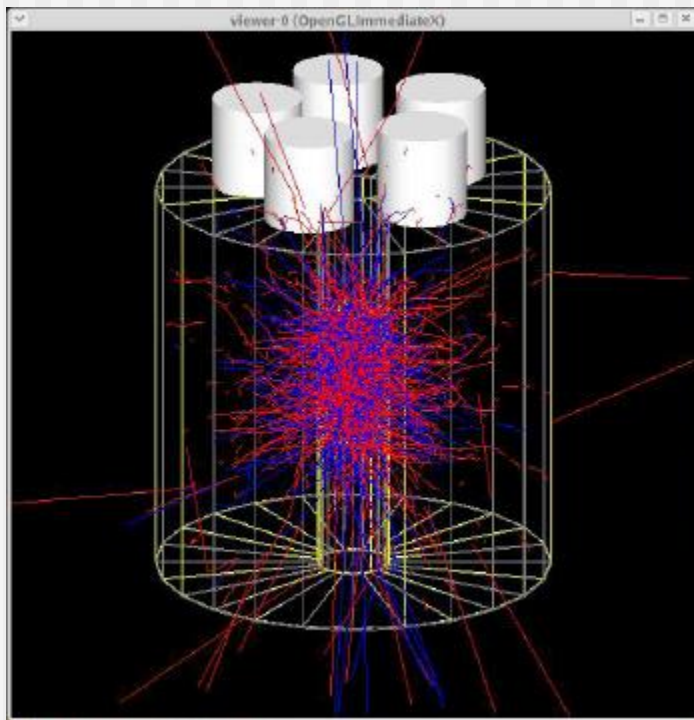
200 MeV 电子在水中的输运过程



探测器模拟（几何设置）



探测器模拟（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。

CERN的蒙特卡罗模拟程序包

GEANT4 是模拟粒子经过物质时所发生的相互作用的一个软件包。 它的应用范围包括：

粒子物理，核物理和加速器物理

空间科学



医学物理

<http://geant4.web.cern.ch/geant4>

蒙特卡罗方法应用举例

如何确定在实验条件下，理论的概率密度函数

贝叶斯定理：
$$P(\text{理论} | \text{实验}) = \frac{P(\text{实验} | \text{理论})}{P(\text{实验})} P(\text{理论})$$

例如：一质量为 m 共振宽度为 Γ 的共振态在实验上观察到的概率密度函数是什么形式？

布莱特-魏格纳
分布



探测器
分辨率



探测
效率

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

$$R(M' | M)$$

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

应用举例(续一)

也就是说, 对应于真实的 M , 实际的 M' 应该是怎样一个分布

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

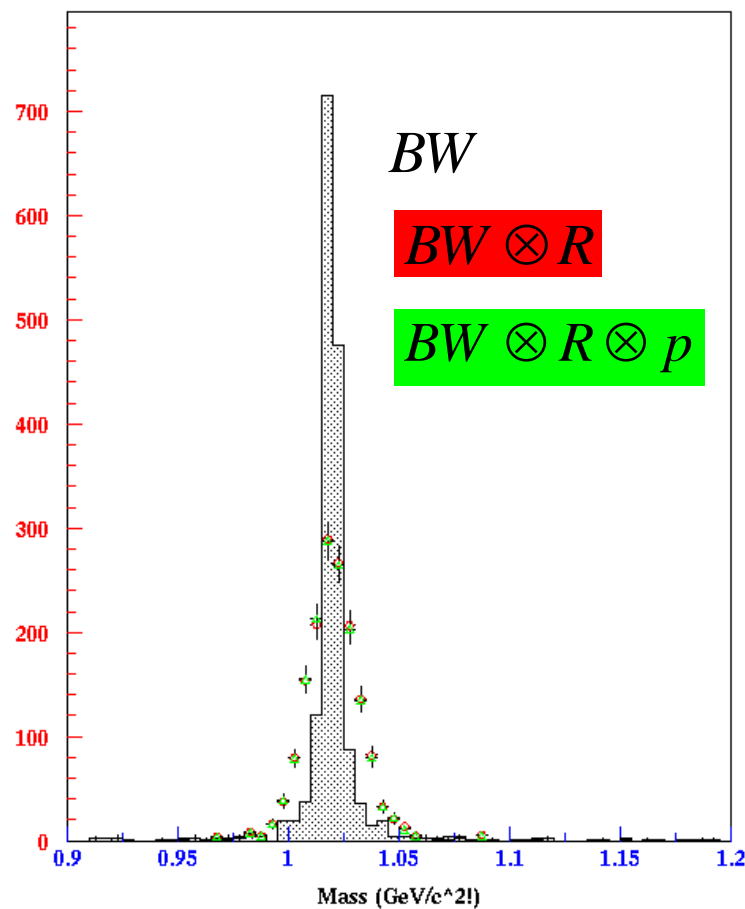
如果假设

$$m_0 = 1.19456 \text{ GeV}/c^2; \Gamma_0 = 0.00426 \text{ GeV}/c^2$$

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

$$p(\varepsilon = 100\% | M) = 0.9(1 - 0.1M^2)$$

应用举例(续二)



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，不能用函数表示时，蒙特卡罗方法可以给出最好的近似。

应用举例(续三)

应用蒙特卡罗方法的步骤:

步骤一: 写出布莱特-魏格纳产生子

输出末态粒子的四动量

步骤二: 输入末态粒子四动量, 模拟粒子在探测器的响应

输出各子探测器响应

步骤三: 输入各子探测器响应, 重建探测粒子的四动量

输出探测粒子四动量

步骤四: 输入探测粒子四动量, 计算不变质量分布



实验条件下预期的布莱特-魏格纳分布

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

■ 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

■ 蒙特卡罗方法的精度问题

精度与积分维度无关, 总是正比于 $1/\sqrt{n}$

■ 在粒子与核物理中的应用

事例产生子与探测器模拟