TFY4235/FY8904 Computational Physics, Spring 2013

Solution Set 7

Problem 1. Here are fortran programs that generate gaussian distributed random numbers using (a) the Box-Müller algorithm (`boxm`) and (b) the Metropolis algorithm (`gmetro`):

```
      program boxm
c Box-Mueller algorithm generating gaussian random numbers
      parameter(nmax=250000,nh=100,nm=-nh,hmax=8.,nc=500)
      dimension nhist(nm:nh),corr(nc),xold(nc)
      hinv=1./hmax
      nbin=nh-nm+1
      nc2=nc/2
      do i=nm,nh
      nhist(i)=0
      enddo
      rinv=0.5/(2.**31-1.)
      ibm=1955
      do i=1,1000
      ibm=ibm*16807
      enddo
      pi2=8*atan(1.)
      do i=1,nmax
      ibm=ibm*16807
      y1=rinv*float(ibm)+0.5
      ibm=ibm*16807
      y2=rinv*float(ibm)+0.5
      x1=sqrt(-2.*log(y1))*cos(pi2*y2)
      x2=sqrt(-2.*log(y1))*sin(pi2*y2)
c Histogram
      nh1=(x1*hinv)*nbin
      nhist(nh1)=nhist(nh1)+1
      nh2=(x2*hinv)*nbin
      nhist(nh2)=nhist(nh2)+1
      do j=2,nc
      xold(j-1)=xold(j)
      enddo
      xold(nc)=x1
c Correlaton function calculation
      if(i.gt.nc2) then
      do j=1,nc
      corr(j)=corr(j)+xold(j)*xold(nc)
      enddo
```

```
      endif
      do j=2,nc
      xold(j-1)=xold(j)
      enddo
      xold(nc)=x2
      if(i.gt.nc2) then
      do j=1,nc
      corr(j)=corr(j)+xold(j)*xold(nc)
      enddo
      endif
      enddo
      nhist(0)=nhist(0)/2
      open(unit=1,file='boxm1.dat',status='unknown')
      open(unit=2,file='boxm2.dat',status='unknown')
      do i=nm,nh
      write(1,*) float(i)*hmax/nbin,nhist(i)
      enddo
      close(1)
      do j=nc,1,-1
      write(2,*) nc-j,corr(j)/corr(nc)
      enddo
      close(2)
      end
```

and

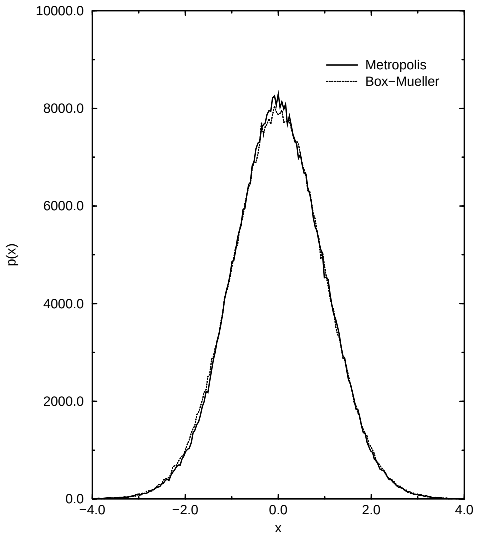
```
      program gmetro
c Metropolis algorithm generating gaussian random numbers
      parameter(nmax=500000,dx=0.5,nh=100,nm=-nh,hmax=8.,nc=500)
      dimension nhist(nm:nh),corr(nc),xold(nc)
      hinv=1./hmax
      nbin=nh-nm+1
      do i=nm,nh
      nhist(i)=0
      enddo
      do i=1,nc
      corr(i)=0.
      enddo
      rinv=0.5/(2.**31-1.)
      ridx=rinv*dx
      ibm=1955
      do i=1,1000
      ibm=ibm*16807
      enddo
      xo=0.
      po=exp(-xo**2*0.5)
```

```
      do i=1,nmax
      ibm=ibm*16807
      xn=xo+ridx*float(ibm)
      pn=exp(-xn**2*0.5)
c Histogram
      if(pn.ge.po) then
      nhn=(xn*hinv)*nbin
      nhist(nhn)=nhist(nhn)+1
      xo=xn
      po=pn
      else
      ibm=ibm*16807
      ran=rinv*float(ibm)+0.5
      if(ran.lt.pn/po) then
      nhn=(xn*hinv)*nbin
      nhist(nhn)=nhist(nhn)+1
      xo=xn
      po=pn
      else
      nhn=(xo*hinv)*nbin
      nhist(nhn)=nhist(nhn)+1
      endif
      endif
      do j=2,nc
      xold(j-1)=xold(j)
      enddo
      xold(nc)=xo
      if(i.gt.nc) then
c Correlaton function calculation
      do j=1,nc
      corr(j)=corr(j)+xold(j)*xold(nc)
      enddo
      endif
      enddo
      nhist(0)=nhist(0)/2
      open(unit=1,file='gmetro1.dat',status='unknown')
      open(unit=2,file='gmetro2.dat',status='unknown')
      do i=nm,nh
      write(1,*) float(i)*hmax/nbin,nhist(i)
      enddo
      close(1)
      do j=nc,1,-1
      write(2,*) nc-j,corr(j)/corr(nc)
      enddo
```

```
close(2)
end
```

A small remark: Note the way that I get the constant $\pi = 3.14...$: It is $4\arctan(1)$. By this construction, one ensures that π is represented with maximum precision in the machine.

In the figure below I plot the histograms that results from the two programs:



The correlation function $\langle x(t)x(t+\tau)\rangle$ for the random number sequences that result from the two algorithms is shown below. There are no visible correlations between the gaussian random numbers that result from the Box-Müller algorithm, while the correlations fall off exponentially in the Metropolis data. This is what was expected.