计算物理第一次作业

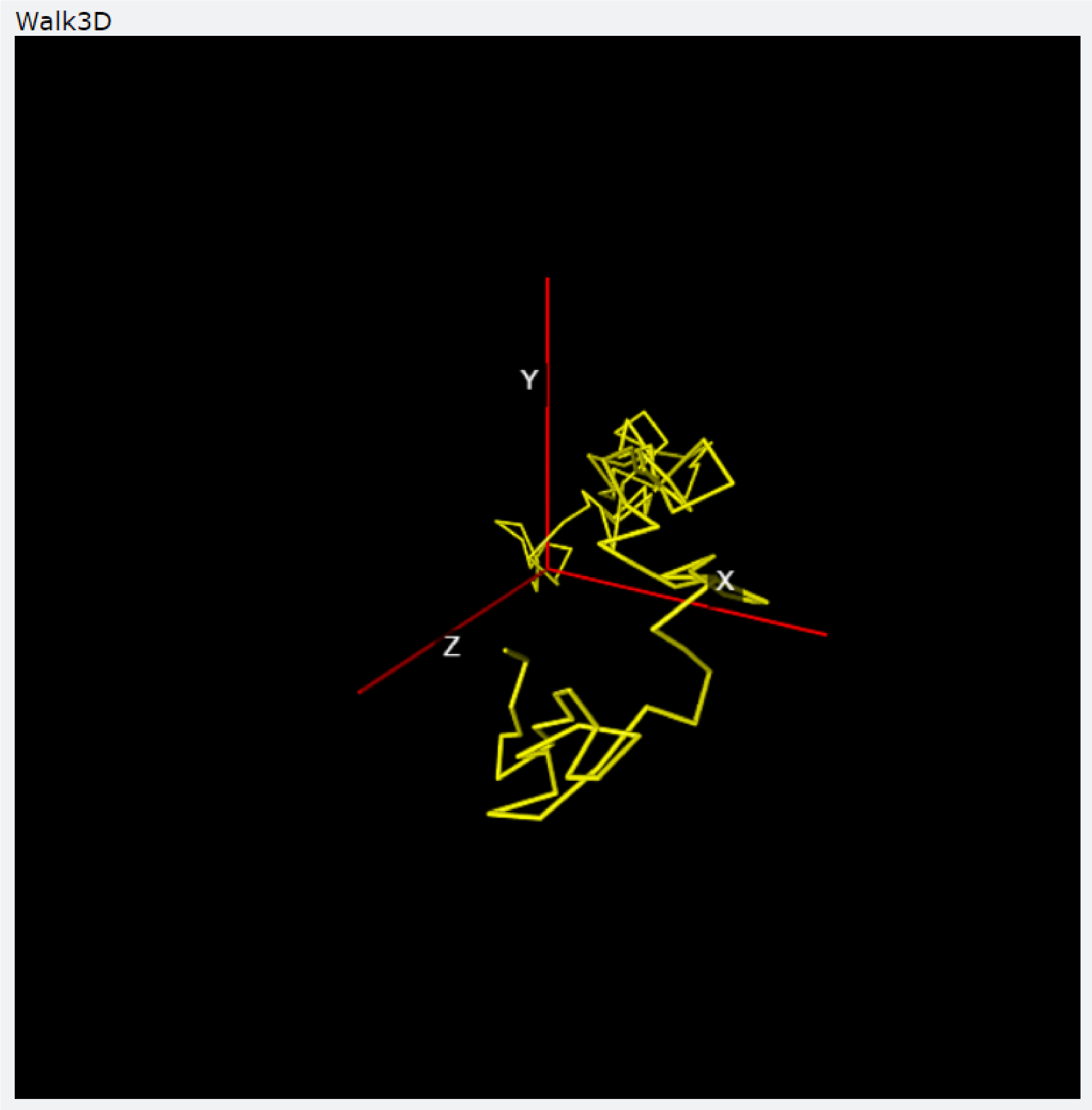
吴远清 2018300001031

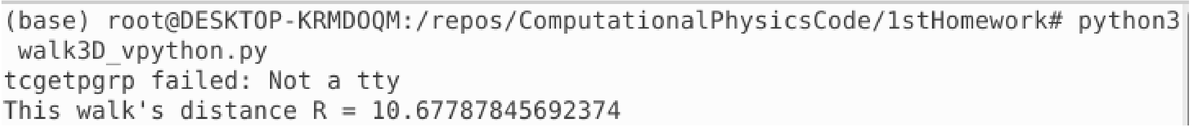
随机行走路径绘制

vpython版本代码如下:

```
from vpython import *
import random

random.seed(None)
jmax = 1000
x0 = y0 = z0 = 0.0

scene = canvas(x=0, y=0, width=600, height=600, title='Walk3D',
forward=vector(-0.6,-0.5,-1))
pts   = curve(x=list(range(0, 100)), radius=10.0,color=color.yellow)
xax   = curve(x=list(range(0,1500)), color=color.red, pos=
[vector(0,0,0),vector(1500,0,0)], radius=10.)
yax   = curve(x=list(range(0,1500)), color=color.red, pos=
[vector(0,0,0),vector(0,1500,0)], radius=10.)
zax   = curve(x=list(range(0,1500)), color=color.red, pos=
[vector(0,0,0),vector(0,0,1500)], radius=10.)
xname = label( text = "X", pos = vector(1000, 150,0), box=0)
yname = label( text = "Y", pos = vector(-100,1000,0), box=0)
zname = label( text = "Z", pos = vector(100, 0,1000), box=0)

pts.append(vector(0,0,0))           # Starting point
for i in range(1, 100):
    x0 += (random.random() - 0.5)*2.           # -1 =< x =< 1
    y0 += (random.random() - 0.5)*2.           # -1 =< y =< 1
    z0 += (random.random() - 0.5)*2.           # -1 =< z =< 1
    pts.append(vector(200*x0 - 100,200*y0 - 100,200*z0 - 100))
    rate(100)
print("This walk's distance R =", sqrt(x0*x0 + y0*y0+z0*z0))
```

运行结果:





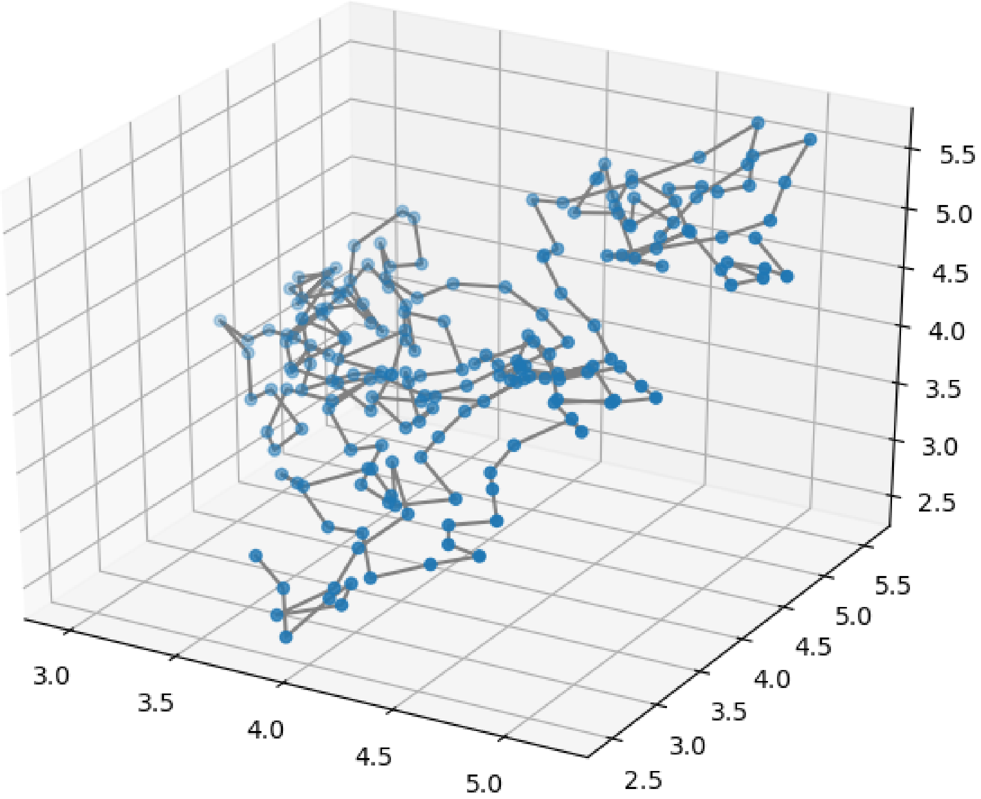
```
(base) root@DESKTOP-KRMDOQM:/repos/ComputationalPhysicsCode/1stHomework# python3
 walk3D_vpython.py
tcgetpgrp failed: Not a tty
This walk's distance R = 10.67787845692374
```

同时给出使用Matplotlib版本的代码和结果:

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from matplotlib import colors
from matplotlib.ticker import PercentFormatter

def randomWalk3D(steps, stepLeng, x0, y0, z0):
    x = np.ones(1) * x0
    y = np.ones(1) * y0
    z = np.ones(1) * z0
    for i in range(steps):
        x = np.append(x, x[-1] + (np.random.random()-0.5)*stepLeng)
        y = np.append(z, z[-1] + (np.random.random()-0.5)*stepLeng)
        z = np.append(z, z[-1] + (np.random.random()-0.5)*stepLeng)
    return x, y, z

totalNum = 5000000
```

```
steps = 200
stepLeng = 0.5
x0 = y0 = z0 = 5
fig = plt.figure()
ax = Axes3D(fig)
xr = np.zeros(totalNum)
yr = np.zeros(totalNum)
zr = np.zeros(totalNum)
r = np.zeros(totalNum)

x, y, z = randomWalk3D(steps, stepLeng, x0, y0, z0)

for i in range(totalNum):
    x, y, z = randomWalk3D(steps, stepLeng, x0, y0, z0)
    dist = np.linalg.norm([x0-x[-1], y0-y[-1], z0-z[-1]])
    r[i] = dist
    xr[i] = x[-1]
    yr[i] = y[-1]
    zr[i] = z[-1]
    if i % int(totalNum/100) == 0:
        print("{}/{}".format(i,totalNum))
plt.title('N = {}'.format(totalNum))
ax.scatter3D(xr,yr,zr,cmap = 'Blues')
plt.savefig("./{}p.png".format(totalNum))
fig, ax = plt.subplots(1,1)
plt.subplot(1,1,1)
plt.hist(r,bins = 20)
plt.title('N={}'.format(totalNum))
fig.savefig('./{}h.png'.format(totalNum))
```

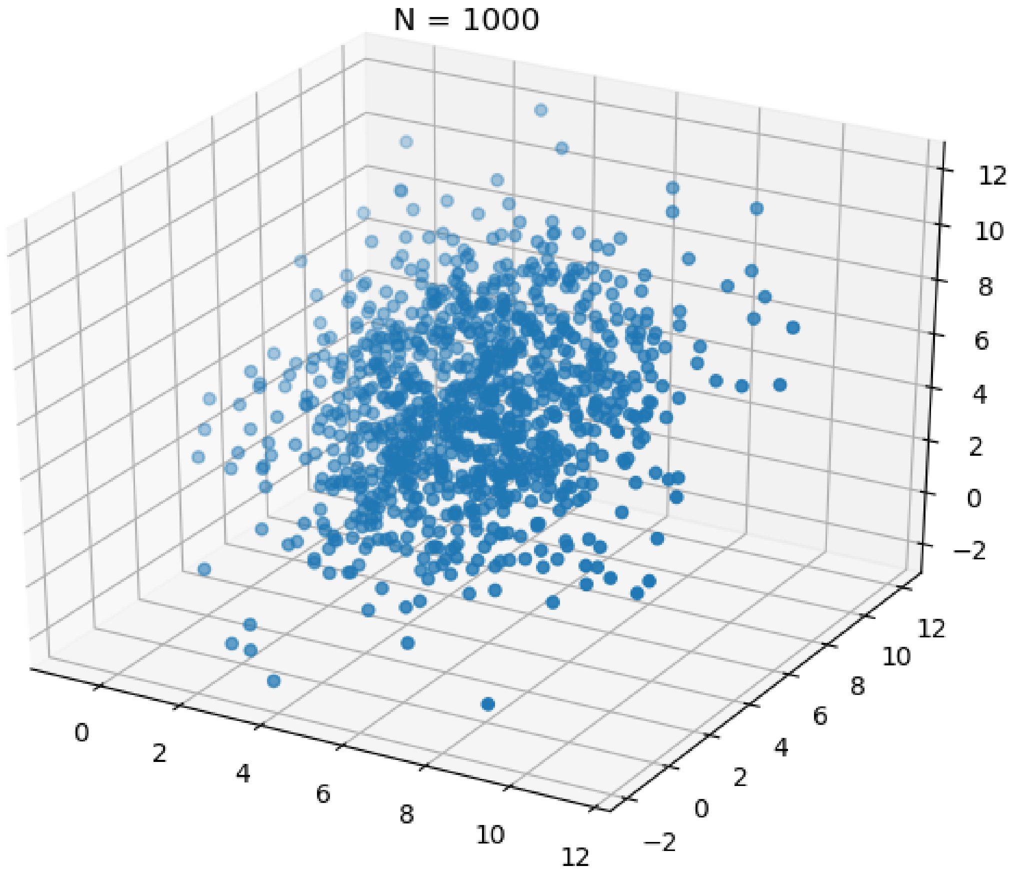
单次随机行走路径结果:

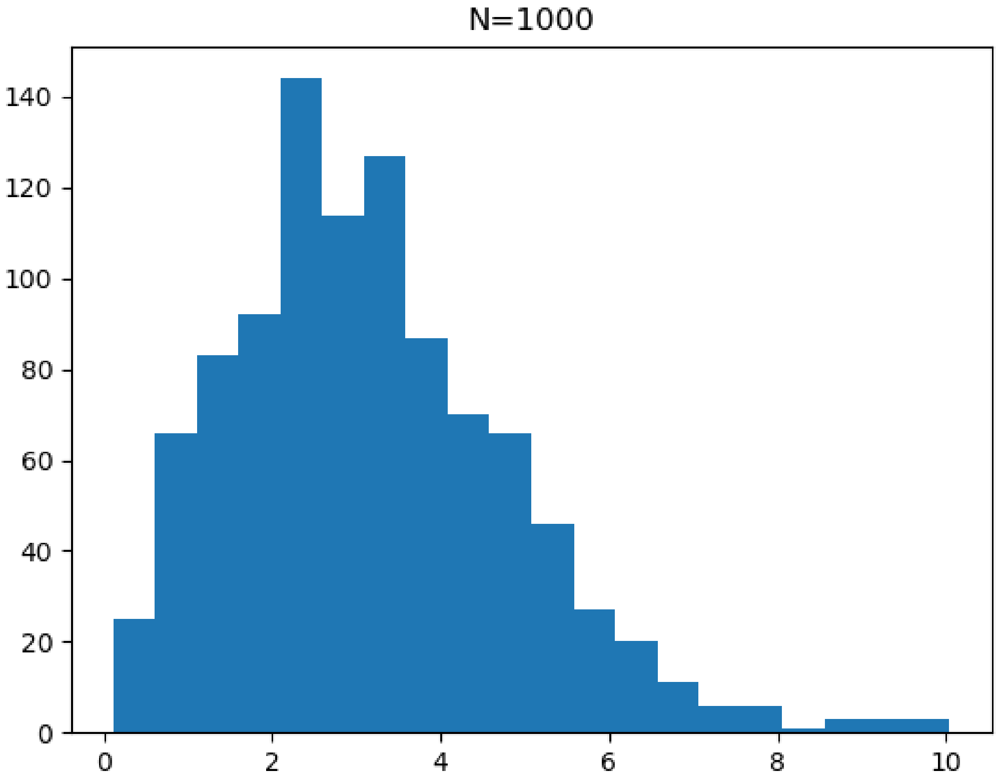


随机行走距离分布统计

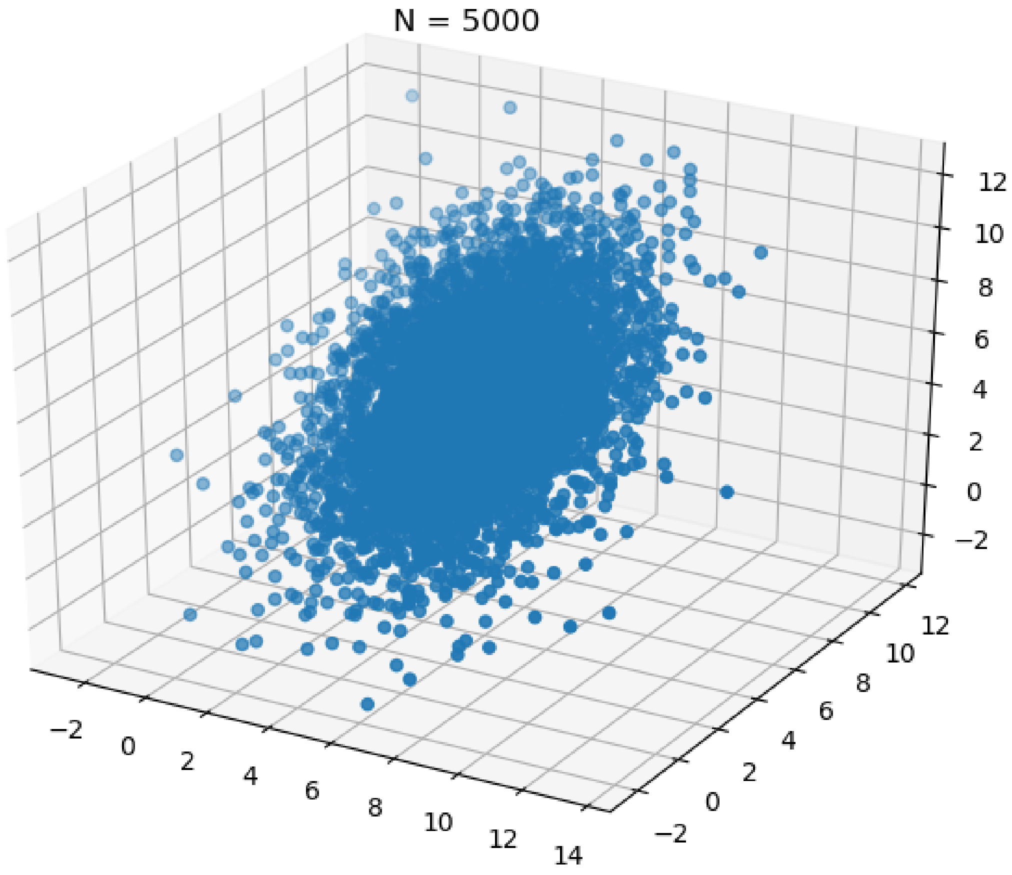
我们使用Matplotlib,结果展示上,我们通过两种方法,第一种是将每次随机行走的终点绘制在3D图中,另一种是绘制不同距离分布的直方图,下面是结果:

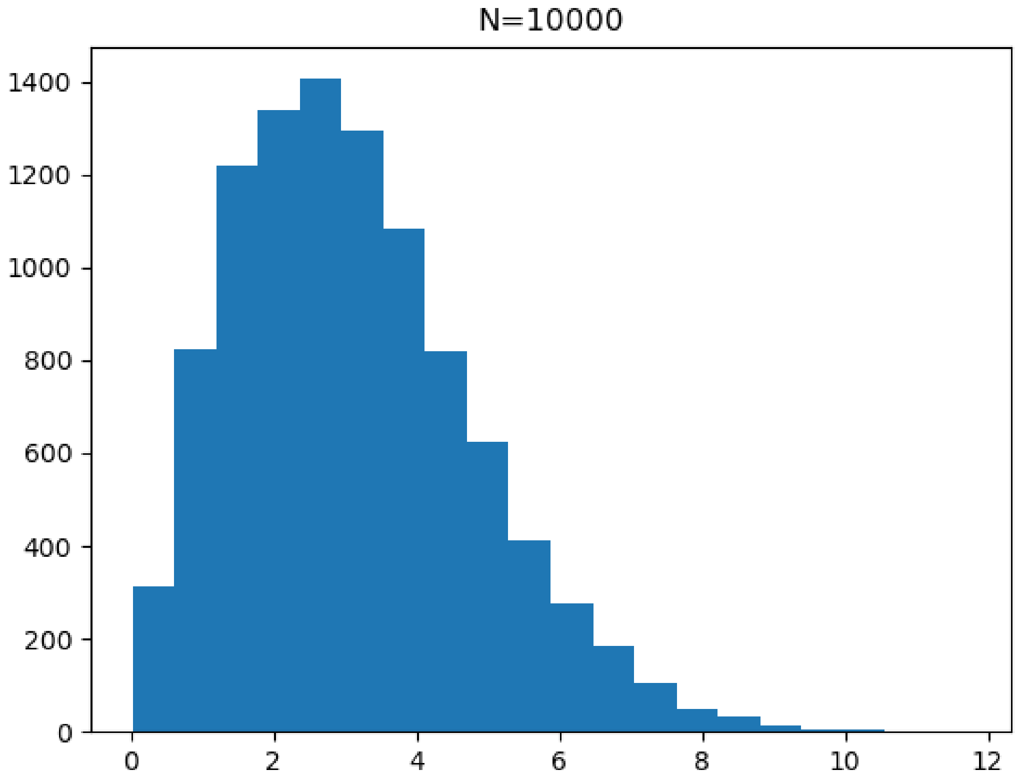
1000次:



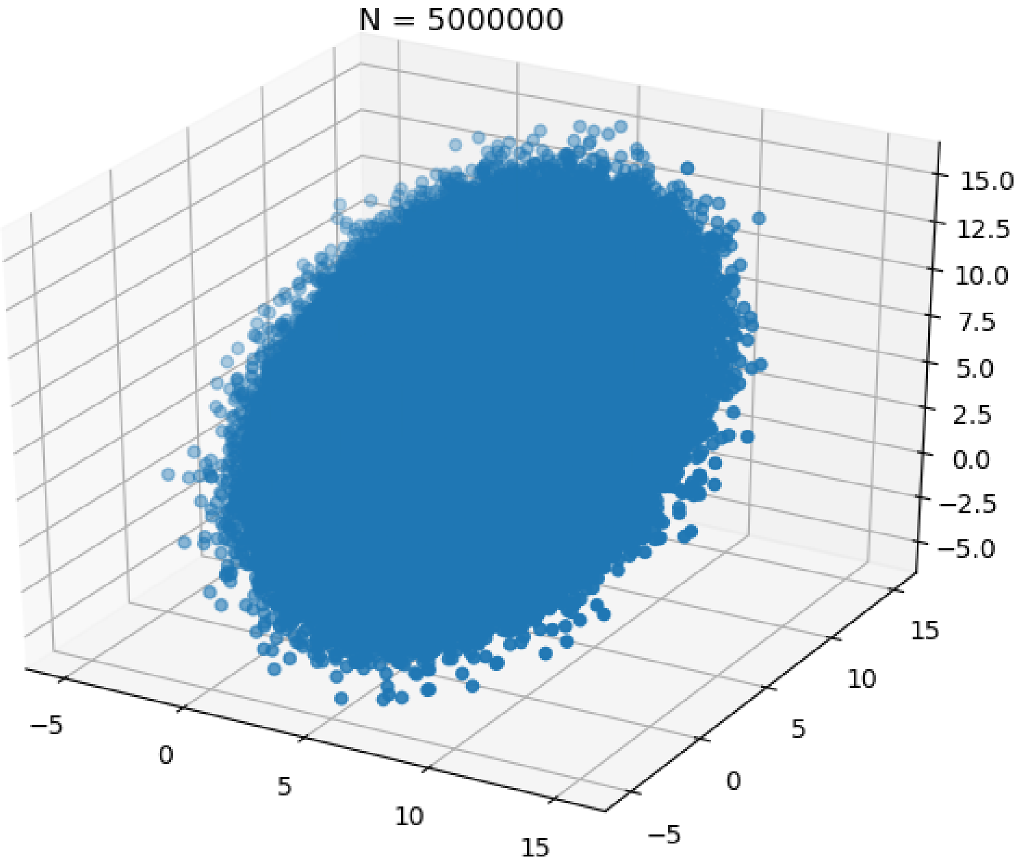


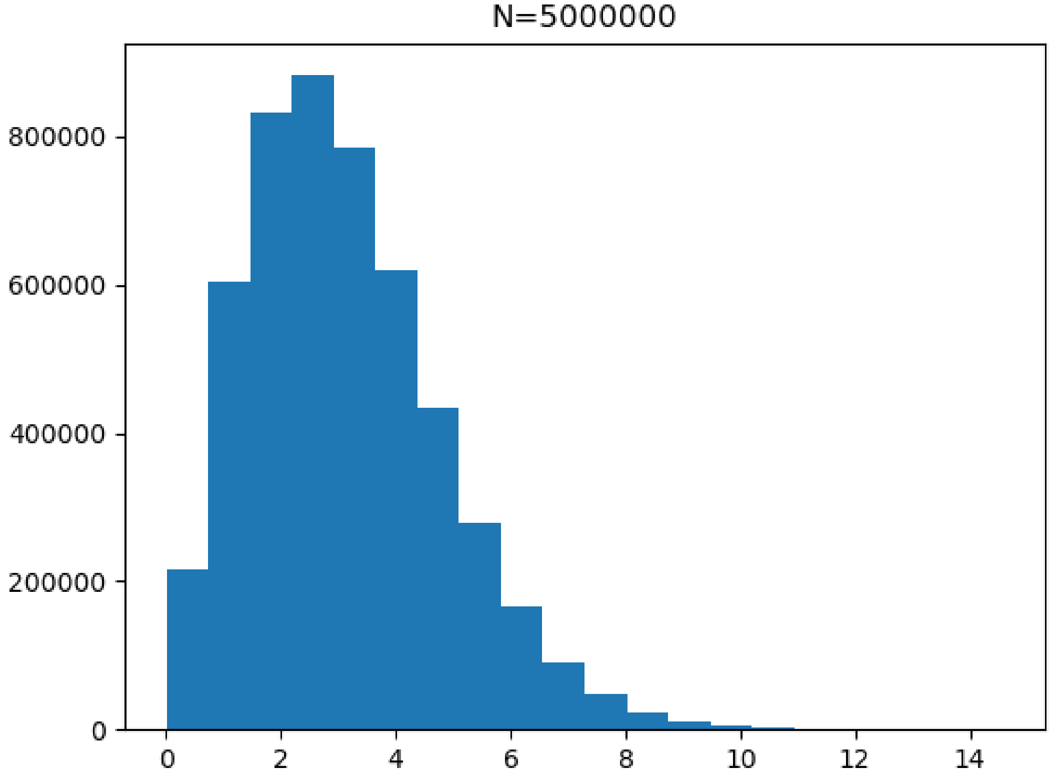
5000次:





5000000次:


N = 5000000
15.0
12.5
10.0
7.5
5.0
2.5
0.0
−2.5
−5.0
−5
0
5
10
15
15
10
5
0
−5


N=5000000
800000
600000
400000
200000
0
0
2
4
6
8
10
12
14