Fluctuation theorem for a single particle in a moving billiard

Hao-Ren Jheng, Jyun-Kai Yang, Che-Hao Chang

National Central University, Chung-Li, Taiwan

June 11, 2015

Abstract

We place a sphere in our self-made Sinai billiard, which is driven periodically by a D.C motor. In our experiment, we set two different precession frequencies f=0.3 and 0.6Hz. Both the experiment results and simulation show that the FT is hold under the frequency f=0.6Hz but failed under f=0.3Hz. Further,

1 Introduction

Over the past few years, physicists developed the concepts and formulations of the Fluctuation Theorem (FT), which describes deviations from the Second Law of Thermodynamics for small systems at small time scales. The FT quantifies the probability of the decrease of the entropy in an isolated system. We consider the formula here,

$$\frac{p(J_\tau)}{p(-J_\tau)} = e^{\frac{J_\tau \tau}{\beta_\tau}} \quad (1)$$

$p(J_\tau)$ is the probability of the mean value of J_τ (of work, momentum, heat, etc.) during a time interval τ.

Our system can be analogous to the system, which contains gas confined to a cylinder with insulating wall and with a movable piston, undergoing isothermal process. The two systems are shown in Fig. 1. The D.C motor is similar to the reservoir with constant temperature, and the sphere (for long time) is corresponding to the gas.

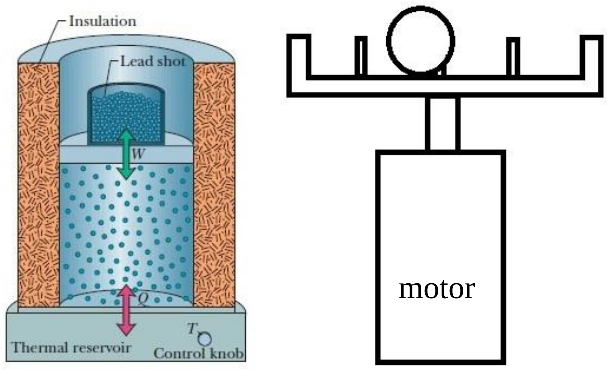


Figure 1: A gas is confined to a cylinder with a movable piston. Heat Q can be added to the gas by the reservoir and work W can be done by the piston.

The First Law of Thermodynamics states that

$$\Delta U = \Delta W_{in} + \Delta Q_{in} \quad (2)$$

where ΔU is the change of the internal energy of a system, ΔW_{in} is the work done by the surroundings on the system, and ΔQ_{in} is the heat supplied to the system. In an isothermal process, the internal energy of the system remain unchanged.

$$\Delta U = 0 = \Delta W_{in} + \Delta Q_{in}$$

$$\Delta W_{in} = -\Delta Q_{in} \quad (3)$$

The definition of the entropy in a system is

$$\Delta S = \int \frac{dQ_{in}}{T} \quad (4)$$

We make the following substitution

$$\Delta S = \int \frac{dQ_{in}}{T} = \int \frac{-dW_{in}}{T} = \int \frac{dW_{out}}{T} = \frac{1}{T}\int dW_{out} \quad (5)$$

where dW_{out} is the work done by the system, and $dW_{out} = -dW_{in}$. The entropy is related to the work. We consider the mean value of the power $J_\tau = W_\tau/\tau$, where W_τ is the work done by the system within a time interval τ. The mean value of the power is determined

$$J_\tau = \frac{1}{\tau}\int_t^{t+\tau} \vec{F}\cdot\vec{v}\,dt \quad (6)$$

for a given time interval τ. $\vec{v}$ is the velocity of the sphere. $\vec{F}$ is the gravitational force

$$\vec{F} = mg\sin(\alpha)\,[\cos(\varphi)\,\hat{x} + \sin(\varphi)\,\hat{y}] \quad (7)$$

where $\varphi = 2\pi f t + \varphi_0$ is the angl9e of the line connecting the lowest point of the box with its center.

2 Experiment setup

Figure 2 shows our device and the dimensions. The box, which is made of acrylic sheet, contains 16 aluminum pins (diameter 1.5cm) stuck equidistance on the bottom of the box (distance between pins is about 3.75cm; distance between pins and wall is about 1.88cm). We constructed our own gyratory mixer, which makes the upper box (Fig. 2⑤) precess around an imaginary vertical line (frequency f, angle α) at the center of the box. The motor (Fig. 2①) joins a square acrylic sheet (Fig. 2②) with three aluminum pins (Fig. 2③) on it. One of the pins is a little higher than the other two to make the column (Fig. 2④) tilt. The column is connected with the upper box with a bearing to make these two objects would move independently. The upper box is further connected with the stick (Fig. 2⑥) to fix the box's motion only in up-down direction. An acrylic (Fig. 2⑦) on the square revolving around the vertical line synchronously with the motor points the direction the box tilt. We put a LED on ⑦, then φ is just the angle of line connecting the LED with its center. Hence, the direction of the gravitational force is determined (Fig. 3).

A sphere (diameter 1.6 cm, mass m=2.21 g) is placed in the box. We recorded the motion of the sphere and LED with high-speed camera with 240 $frames/s$ for about 2 hours, and used "Tracker" program to determine the velocity of the sphere and the position of the LED. After getting the raw data, we wrote C programs to process those data (calculate J_τ and the probability). The logic of the program is simply the equation (6). The time step dt here is $\frac{1}{240}$s. Since dt is not infinitesimal, the integration is just summation.

$$J_\tau = \frac{1}{\tau} \sum_{(frame)_t}^{(frame)_{t+\tau}} (\vec{F} \cdot \vec{v}) \Delta t$$

For $\tau = 0.1$s, we need to sum up 24 frames; for $\tau = 0.2$s, we need to sum up 48 frames, and so on. The code is shown in Appendix A.

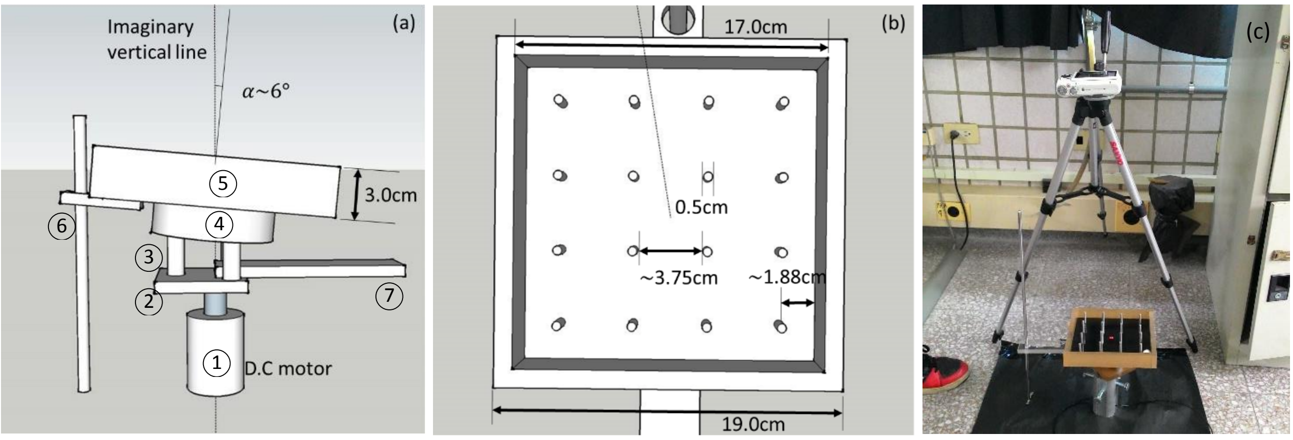


Fig. 2: Experimental setup with dimensions (a) Side view (b) Top view (c) the actual device.

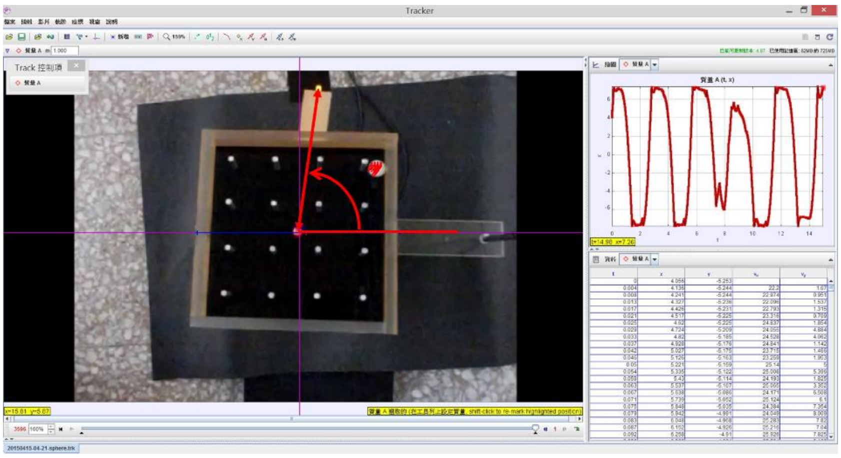


Fig. 3: The interface of the Tracker program. The purple lines are the coordinate axes. The program would automatically assume that the horizontal one is x axis and the vertical one is y axis. The blue line is the scale. φ is the angle between the line connecting the LED with the box's center and the x axis.

3 Experimental and simulation results

We set the frequency f=0.3 and 0.6 Hz in our experiment. Figure 4 show the probability distribution $p\,(J_\tau)$ of our result (left one for f=0.3Hz, right one for f=0.6Hz). Different icons (colors) stands for different time interval τ

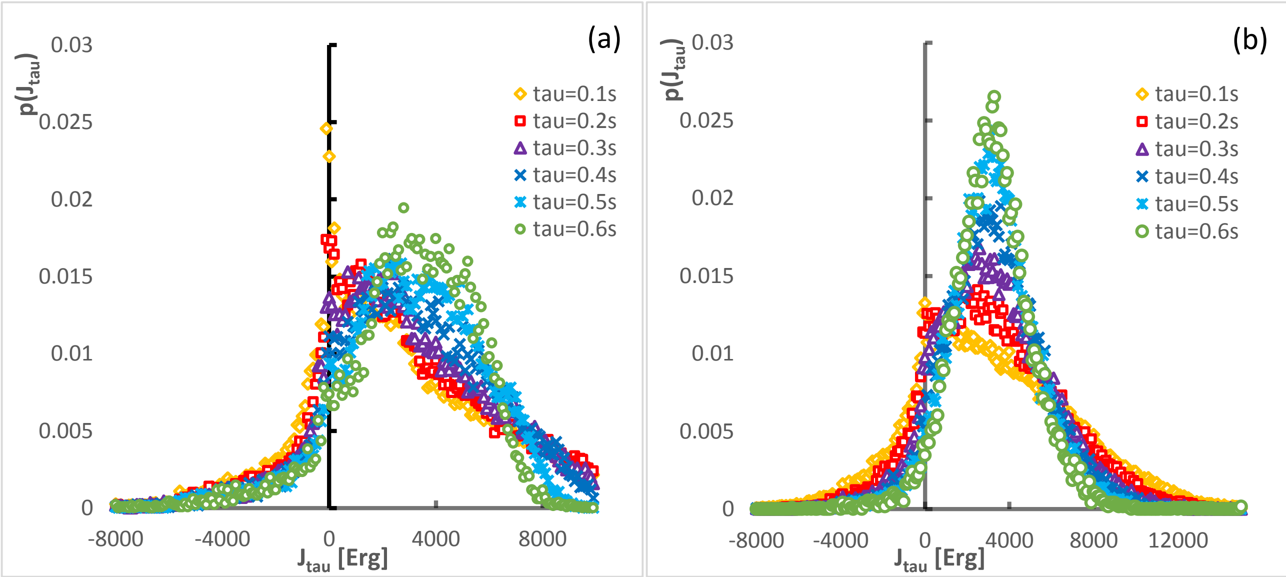
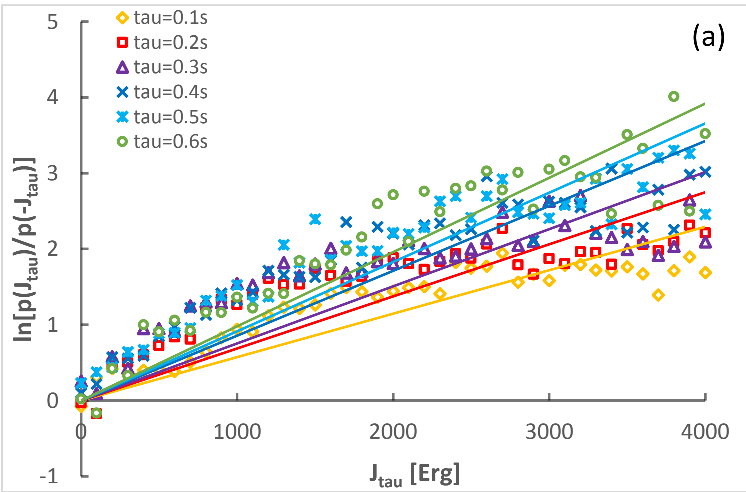


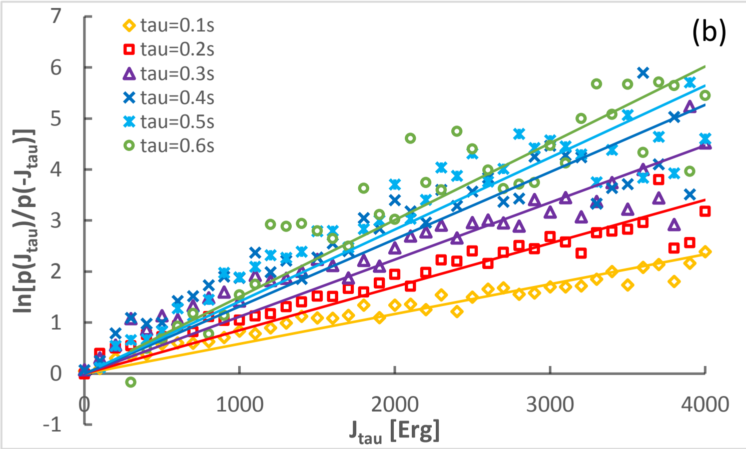
Fig 4: Experimental results. Different icons (colors) stands for different time interval τ (a) Probability distributions of the mean power J_τ for f=0.3Hz. (b) Probability distributions of the mean power J_τ for f=0.6Hz.

In the above plots, both figures show that the distribution become narrower as the time interval τ increases. The parts of negative values of J_τ decrease when τ become larger. This phenomenon is just what Second Law of Thermodynamics expect. In order to confirm the FT, we take the nature log of Eq.1 and plot $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ in figure 5 (left one for f=0.3Hz, right one for f=0.6Hz). The tables beside the figures display the coefficients of determination of the best-fit line.

$$\ln\left(\frac{p(J_\tau)}{p(-J_\tau)}\right)=\left(\frac{\tau}{\beta_\tau}\right)J_\tau=S\,J_\tau \quad (8)$$



tau (s)	R^2
0.1	0.6300
0.2	0.4589
0.3	0.2747
0.4	0.6268
0.5	0.7038
0.6	0.8027



tau (s)	R^2
0.1	0.9122
0.2	0.8854
0.3	0.8346
0.4	0.7927
0.5	0.8468
0.6	0.8407

Fig 4: Plot of $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ based on experimental results. Different icons (colors) stands for different time interval τ. (a) For $f = 0.3Hz$ (b) For $f = 0.3Hz$. The tables beside the figures display the coefficients of determination of the best-fit line.

We could not fit the data well in linear relationship in Fig (a), while there is a relatively good linearity in Fig (b). We made the following explanation. The higher frequencies the precession undergoes, the more homogeneous the sphere's spatial distribution will be. If the frequencies are too low, the sphere will just move around the borders, causes an inhomogeneous distribution and the failure of the FT. Figure 5 show the spatial distribution of the sphere in a short period of time (15 seconds). The left one is of f=0.3Hz, while right one is of f=0.6Hz.

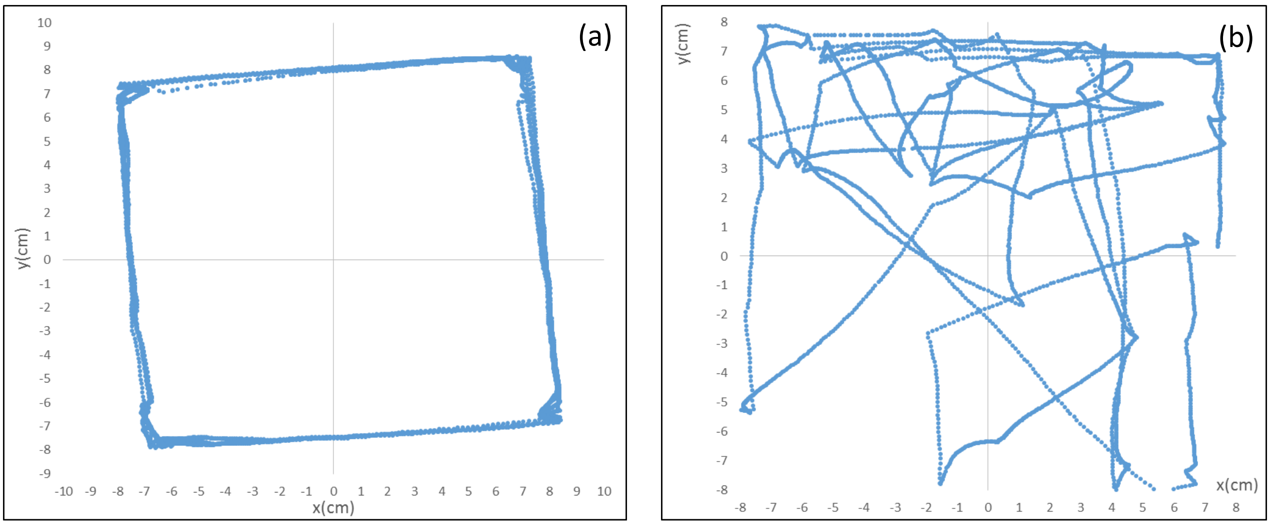


Fig 5: The spatial distribution of the sphere in 15 seconds under (a) f=0.3 Hz and (b) 0.6 Hz

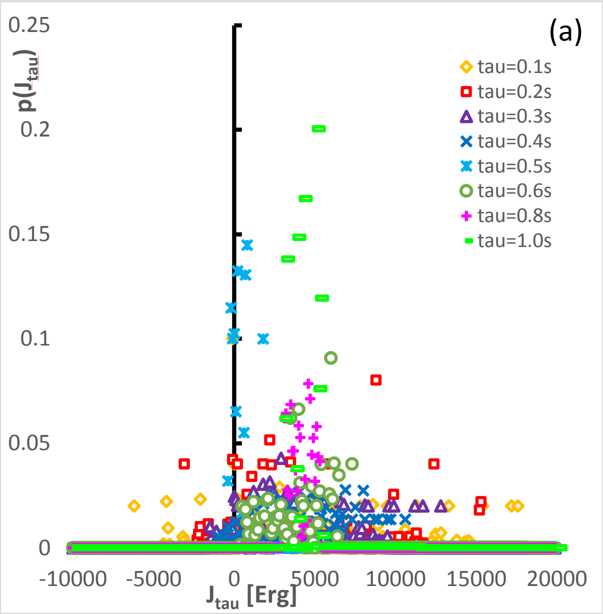
We can see in Fig. 5(a) that the sphere almost moved around the borders, and did not collide with pins under the frequency f=0.3 Hz. The probability of the appearance near the walls is much higher than that of other places, which means that this condition is not a disordered state. In thermodynamics, the randomness is one of the essential factors that should be met in systems we are observing. Such an ordered state as showing in Fig. 5(a) does not agree with the assumptions about randomness. On the other hand, this practically periodical movement of the sphere is not what we expect the system to behave. The purpose of placing pins in the box is to let sphere move chaotically. Colliding with pins in different incident angles, even tiny deviation between them, causes two distinct paths. The divergence will be magnified with time. This is a significant characteristic of chaotic system. Collision mostly with wall would not result in "enough" chaotic motion. It is the reason that causes the breakdown of linearity of $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ. If the frequencies are too low, the sphere will just move around the borders, causes an inhomogeneous distribution. The FT fails to describe this kind of systems. In higher revolving frequency, the sphere could appear anywhere. This yields uniform spatial distribution and relatively chaotic motion, which makes the FT valid.

Apart from the experiment, we also write a C program to simulate the motion of the sphere. Since the tilt angle α is small, we do the approximation by taking the calculation in two dimensions. There are several assumptions when carrying out the simulation. We assume that there is no friction between the surface of the box and the sphere. The sphere move without friction and without rotating. The sphere is hard enough that its momentum changes instantaneously when it collides with pins and the wall. Parameters input to our program are all the same as those in experiment. They are listed in Table 1. The collision of the sphere and pins or wall is not elastic, the component of the velocity of the sphere perpendicular to the pin's or wall's surface must multiplied by restitution coefficients (e_p/e_w for pins / wall). Detail discussion about restitution coefficients are described in next section. The simulation code is shown in Appendix B.

Table 1: Parameters input to our simulation program

Diameter of the sphere	1.58 cm
Diameter of the pins	0.5 cm
Borders	$-7.71 \le \mathrm{x} \le 7.71$ cm $-7.71 \le \mathrm{y} \le 7.71$ cm
Time duration	2 hours
Time step dt	$1/240$ s
Restitution coefficient $\mathrm{e_w/e_p}$	0.406/0.735
Precession frequency	0.3 & 0.6 Hz

Figure 6 in the next page show the probability distribution $p\,(J_\tau)$ of our simulation result (left one for f=0.3Hz, right one for f=0.6Hz).



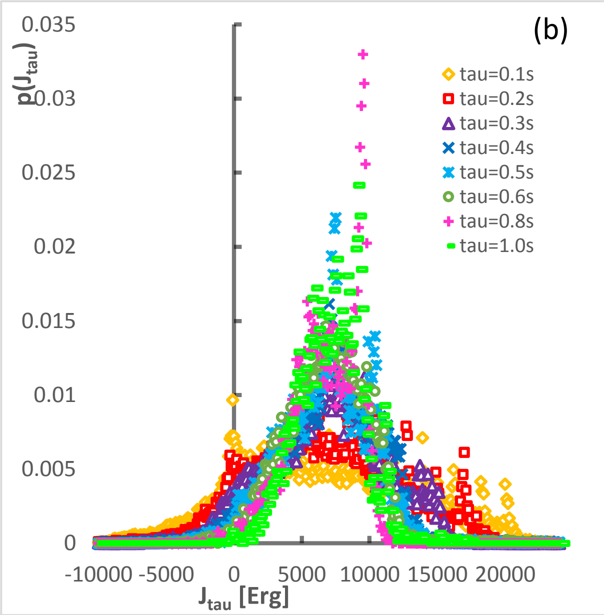
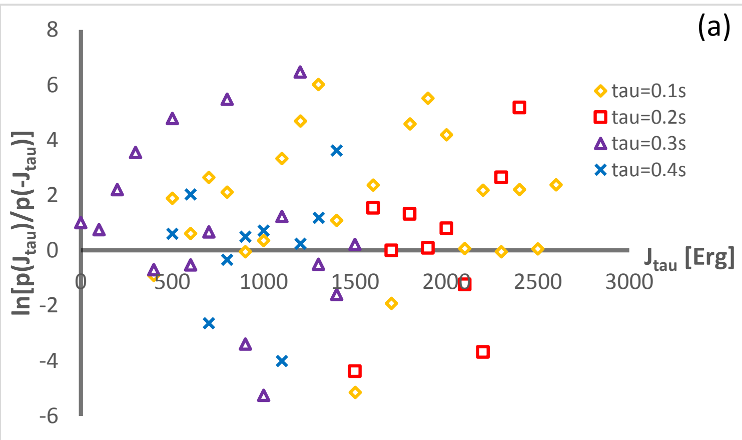
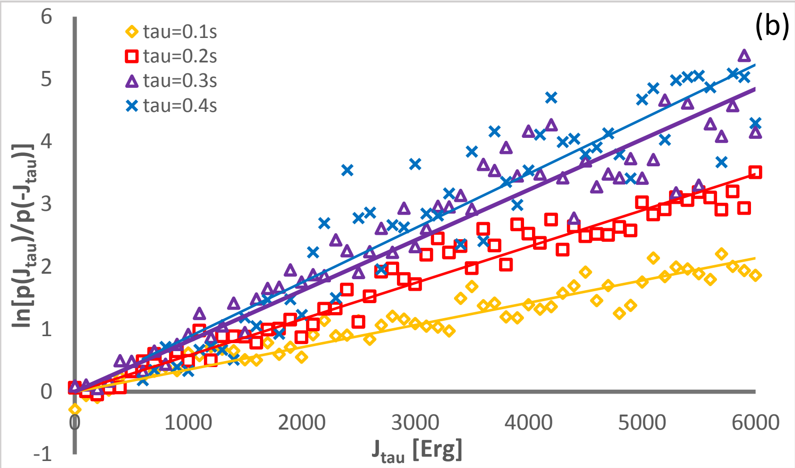


Fig 6: Simulation results. Different icons (colors) stands for different time interval τ (a) Probability distributions of the mean power J_τ for f=0.3Hz. (b) Probability distributions of the mean power J_τ for f=0.6Hz.

Under the frequency $f = 0.6$ Hz, the probability distribution is similar to our result. The distribution become narrower as the time interval τ increases. The parts of negative values of J_τ decrease when τ become larger. However, under the frequency $f = 0.3$ Hz, the probability distribution is completely different with that of our experiment. We do the same procedure as in experimental result— plot $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ in figure 5 (left one for f=0.3Hz, right one for f=0.6Hz). The tables beside Fig. 7 (b) display the coefficients of determination of the best-fit line.





tau (s)	R^2
0.1	0.9109
0.2	0.9518
0.3	0.9038
0.4	0.8737

Fig 7: Plot of $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ based on simulation results. Different icons (colors) stands for different time interval τ. (a) For $f = 0.3Hz$ (b) For $f = 0.3Hz$. The tables beside the Fig. 7(b) display the coefficients of determination of the best-fit line.

Since the probability of $-J_\tau$ for $\tau \geq 0.5s$ are too low of $f = 0.3Hz$, we could only plot the results for $\tau < 0.5s$. We could not see any clear relationship between $\ln[p(J_\tau)/p(-J_\tau)]$ and J_τ. The figure of the condition $f = 0.6Hz$ demonstrates the well-fitting of these two variables and prove the validation of the FT.

Comparing the experimental and simulation results of $f = 0.3Hz$, both show the breakdown of linearity of $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ. Even so, the plots of probability distribution and confirmation of linearity are not similar at all. We explain this results as follow. In the simulation program, the whole device move ideally just as how we want it to be. But in our real experiment, we are not able to set our device without any inaccuracy. Return to the Experimental setup, if the column does not connect with the upper box right at the center, the box will tilt more at one side. This unequal slope of the box would make the sphere tend to collide with pins. Figure 8 briefly demonstrates this situation. The box rotates counterclockwise (purple solid arrow). The green circle with solid line represents the "initial" position of the sphere. If the bearing (blue circle with solid line) connects with the box right at the center, the sphere would move along the red solid line. If the bearing does not connect with the box at the center (blue circle with dotted line), the box would tilt more at the direction as the arrow points. As a result, the sphere would follow the wathet blue solid line and collide with pins. That's the reason that system in our experiment under $f = 0.3Hz$ still remain randomness.

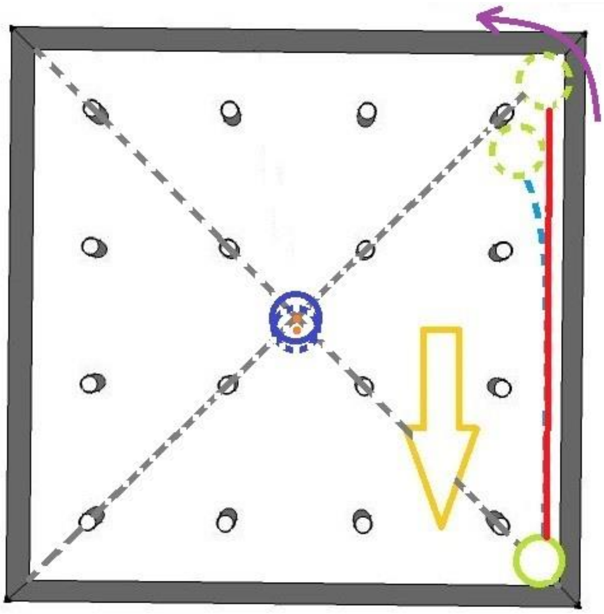
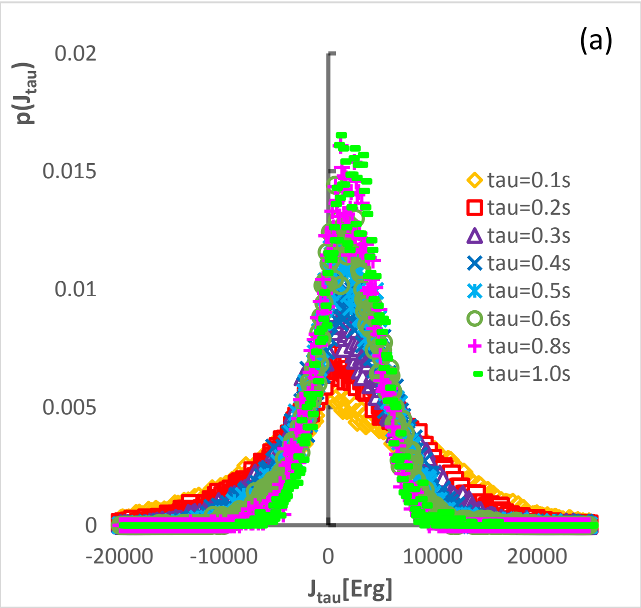
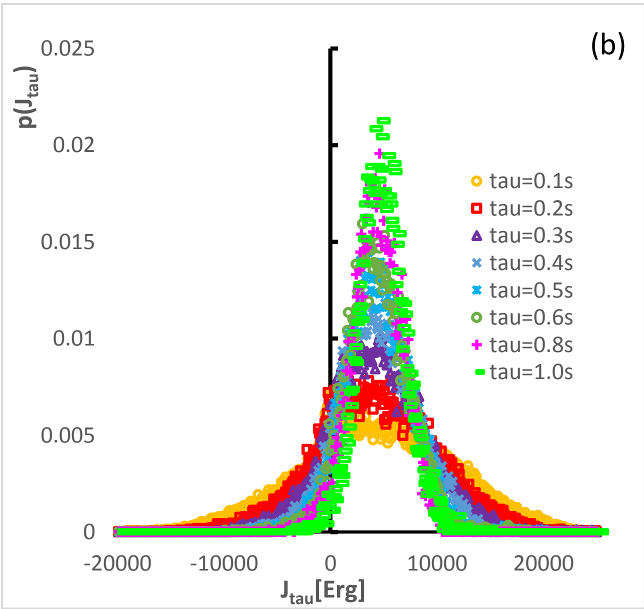


Fig. 8: If the bearing (blue circle with solid line) connects with the box right at the center, the sphere would move along the red solid line. If the bearing does not connect with the box at the center (blue circle with dotted line), the box would tilt more at the direction as the arrow points. As a result, the sphere would follow the wathet blue solid line and collide with pins.

Another contributing factor for spatial distribution is the restitution coefficient. We observe different combinations of e_w and e_p in simulation.





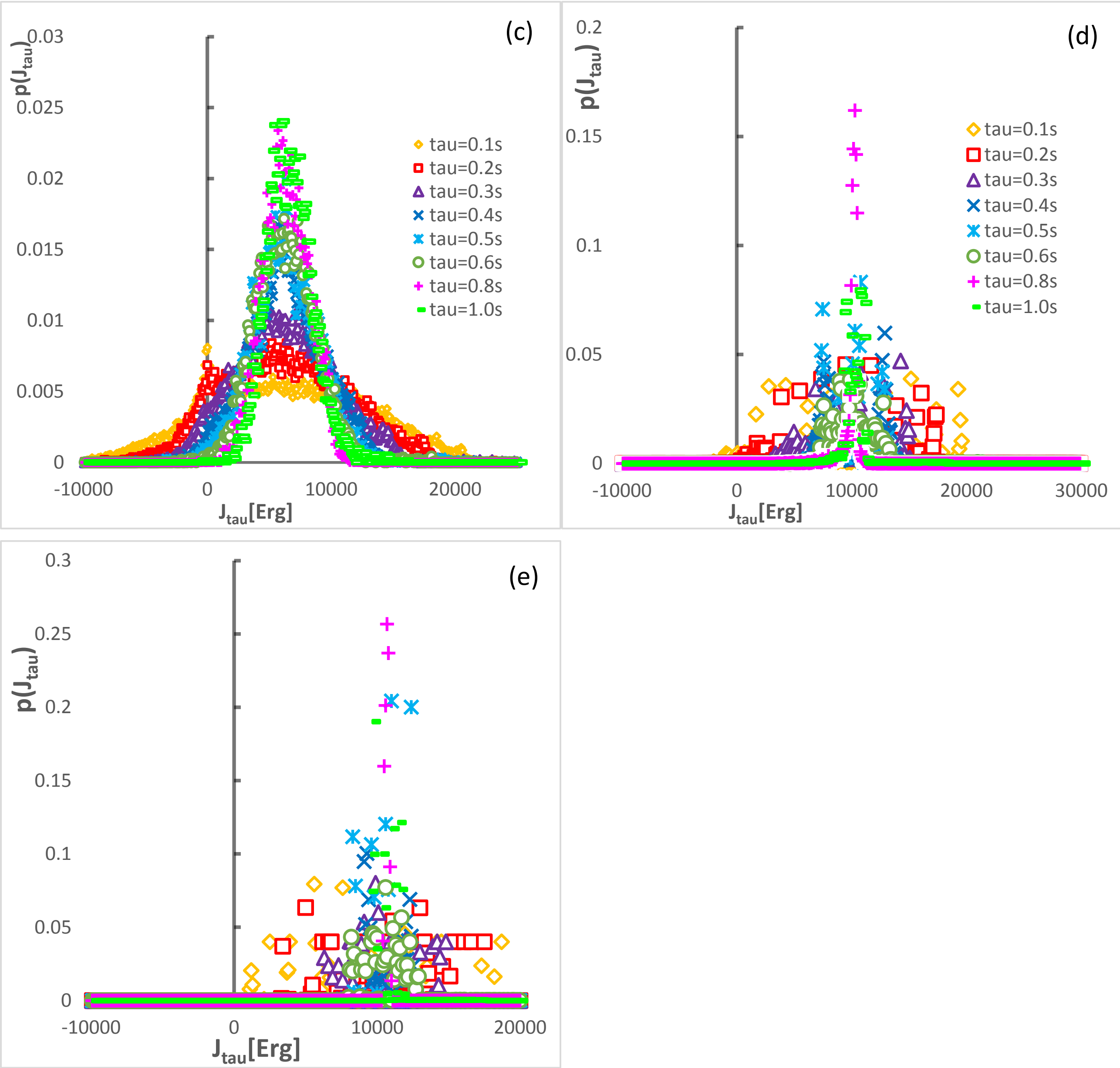


Fig. 9: Simulated probability distribution for different combinations of e_w and e_p. (a) $(e_w, e_p) = (1.0, 0.9)$ (b) $(e_w, e_p) = (0.7, 0.7)$ (c) $(e_w, e_p) = (0.5, 0.5)$ (d) $(e_w, e_p) = (0.3, 0.3)$ (e) $(e_w, e_p) = (0.1, 0.1)$

To confirm the FT, we plot $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ as what we did before (in the following figures we would not put the table just for simplicity). From Fig. 9(d) and (e), we see that the proportion of negative part of these two plot are very low. As a result, we are not able to plot $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ for these two conditions. However, the distribution "pattern" of these two plots are just similar to the simulation result of frequency $f = 0.3Hz$, in which the FT cannot hold. Therefore, we expect that the FT would not hold in these two conditions.

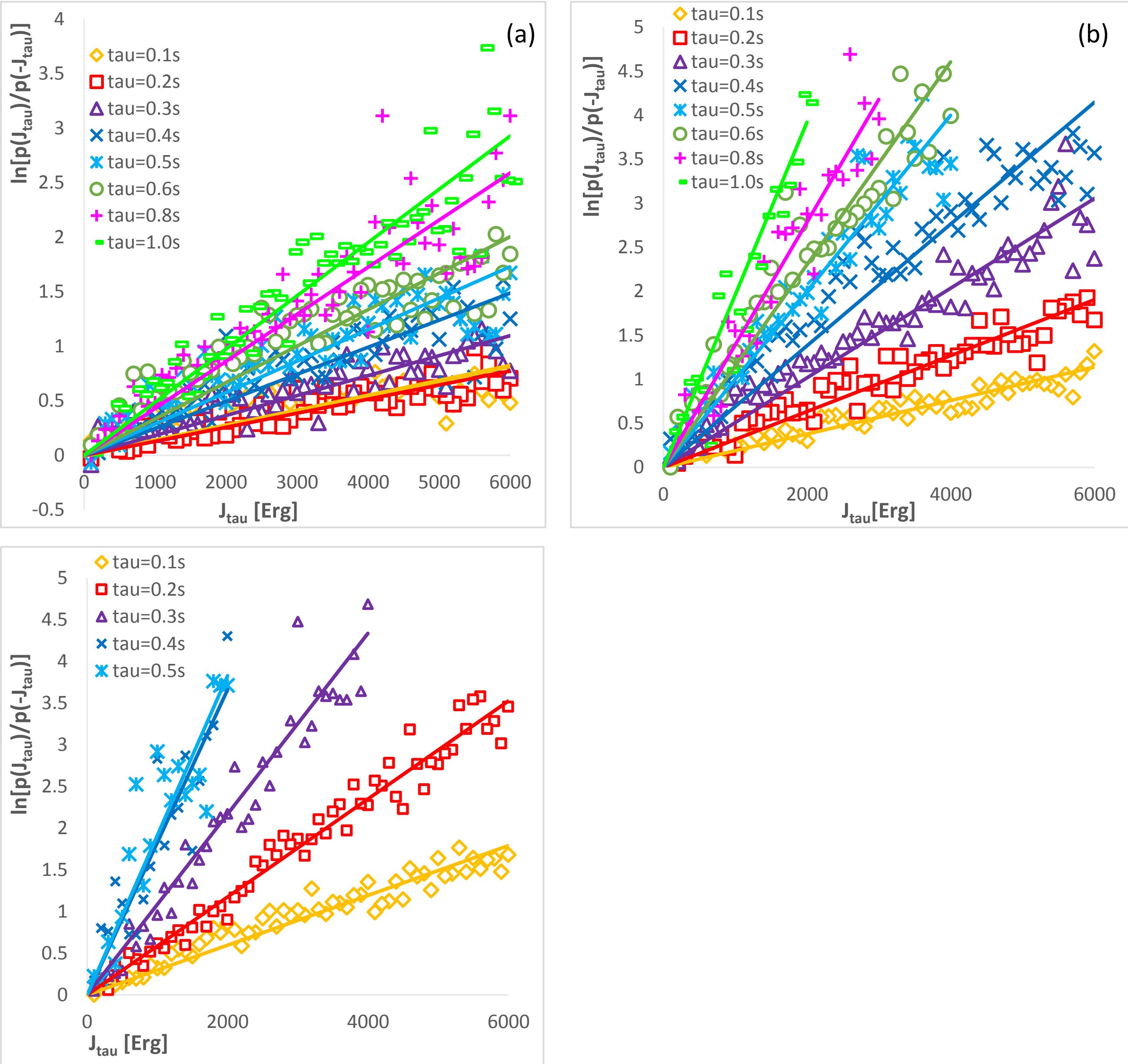


Fig. 10: Plot of $\ln[p(J_\tau)/p(-J_\tau)]$ versus J_τ for different combinations of $\mathrm{e_w}$ and $\mathrm{e_p}$. (a) $(\mathrm{e_w}, \mathrm{e_p}) = (1.0, 0.9)$ (b) $(\mathrm{e_w}, \mathrm{e_p}) = (0.7, 0.7)$ (c) $(\mathrm{e_w}, \mathrm{e_p}) = (0.5, 0.5)$

These three plots show the well-fitting of $\ln[p(J_\tau)/p(-J_\tau)]$ and J_τ, the conditions of the FT are satisfied in these three occasions. The higher restitution coefficients are, the more homogeneous the spatial distribution of the sphere will be, since the sphere is easier to bounce off pins or walls. The necessary disordered state is not given in low restitution coefficients.

4 Conclusion

In our experiment results, we showed that the system will return energy to its power supply for small time scale. Observing two different conditions to examine the FT, we failed in the condition of low revolving frequency $f = 0.3Hz$ but succeeded under $f = 0.6Hz$. In our simulation, we observe two contributing factors that would influence whether the FT is succeed, and confirm the results from experiment. We concluded that system consisting only one particle can be described by the FT and gave support with both experimental evidence and numerical simulations.

Reference

[1]Malte Schmick, Qi Liu, Qi Ouyang, and Mario Markus, Fluctuation theorem for a single particle in a moving billiard: Experiments and simulations, PHYSICAL REVIEW E 76, 021115 (2007).

[2]Malte Schmick, Alexander Hasselhuhn, Qi Liu, Qi Ouyang, and Mario Markus, Fluctuation theorem for a single particle in a Sinai-billiard geometry: predictions and measurements, East-West Journal of Mathematics 01/2007

[3] Malte Schmick and Mario Markus, Fluctuation theorem for a deterministic one-particle system, PHYSICAL REVIEW E 70, 065101(R) (2004)

Appendix A (For simulation)

```
#include <iostream>
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <time.h>
using namespace std;
void randomize()
{
	int i;
	time_t t;
	srand((unsigned) time(&t));
}

int main(void)
{
	double pi = M_PI ;

	double s_x , s_y , x=0 , y=0 ;
	 double xtemp , ytemp ;

	 double vx , vy ;
	 double phi_0 ;
	 double alpha = pi/18 ;
	 double f ;
	 double omega ;
	 double dt = 0.004166667 , tsum = 0 , T ;

	 double g = 980 ;
	 double d[16][3] ;

	 double ep , ew ;

	 FILE *fptp;
	 fptp=fopen("D:\\仁\\物理小年會\\simulation\\pin coordinate.txt","r");
	 for(int i=0;i<16;i++)
	 {
		 for(int j=0;j<3;j++)
		 {
			 fscanf(fptp,"%lf",&d[i][j]);
		 }
	 }
	 fclose(fptp);

	 //printf("time duration(s): ");
	 //scanf("%lf",&T);

	 printf("frequency(Hz): ");
	 scanf("%lf",&f);
	 omega = 2*pi*f ;

	 printf("coefficient of the pin: ");
	 scanf("%lf",&ep);
	 printf("coefficient of the wall: ");
	 scanf("%lf",&ew);

	 randomize();

	 vx = 0 ;
	 vy = 0 ; // the sphere's velocity is assumed to be 0 at t=0
	 phi_0 = (double) (rand() % 36000) / 100 ;
```

```
char fileout1[60];
int N;
printf("半分鐘數: ");
scanf("%d",&N);

for(int i=0;i<N;i++)
{
	tsum=30*i;

	sprintf(fileout1, "%d.txt",i+1);
	FILE *fptr=fopen(fileout1,"w");

	while(tsum<=30*(i+1))
	{

	s_x = vx*dt + 0.5*sin(alpha)*cos(phi_0*(pi/180) + omega*tsum) ;
	s_y = vy*dt + 0.5*dt*dt*g*sin(alpha)*sin(phi_0*(pi/180) + omega*tsum) ;
	xtemp = x ;
	ytemp = y ;
	x = x + s_x ;
	y = y + s_y ;
	vx = vx + dt*g*sin(alpha)*cos(phi_0*(pi/180) + omega*tsum) ;
	vy = vy + dt*g*sin(alpha)*sin(phi_0*(pi/180) + omega*tsum) ;

	for(int a=0;a<=15;a++)
	{
		d[a][2] = sqrt(pow(x-d[a][0],2)+pow(y-d[a][1],2));

		if(d[a][2]<=1.04)
		{
			x=xtemp;
			y=ytemp;

			d[a][2] = sqrt(pow(x-d[a][0],2)+pow(y-d[a][1],2));

			vx = (-ep-1)*((vx*(x-d[a][0])+vy*(y-d[a][1]))/(d[a][2]*d[a][2]))*(x-d[a][0]) + vx;
			vy = (-ep-1)*((vx*(x-d[a][0])+vy*(y-d[a][1]))/(d[a][2]*d[a][2]))*(y-d[a][1]) + vy;

			s_x = vx*dt;
			s_y = vy*dt;
			x = x + s_x ;
			y = y + s_y ;

			break;
		}
	}

	if(x<=-7.710000||x>=7.710000)
	{
		x=xtemp;
		y=ytemp;
		vx=(-1)*ew*vx;
		vy=vy;

		s_x = vx*dt;
		s_y = vy*dt;
		x = x + s_x ;
		y = y + s_y ;
	}
	if(y<=-7.710000||y>=7.710000)
	{
		x=xtemp;
```

```
            y=ytemp;
            vx=vx;
            vy=(-1)*ew*vy;

            s_x = vx*dt;
             s_y = vy*dt;
             x = x + s_x ;
             y = y + s_y ;
        }

         fprintf(fptr,"%.10f %.10f %.10f %.10f %.10f %.10f %.10f
         \n",tsum,x,y,vx,vy,cos(phi_0*(pi/180)+omega*tsum),sin(phi_0*(pi/180)+omega*tsum));
       tsum=tsum+dt;

        }
  fclose(fptr);
   }


  system("pause");
  return 0;
}
```

Appendix B (For data analysis)

```
#include<stdlib.h>
#include<stdio.h>
#include<math.h>
#include<string.h>
#define datanumber 3597

int cmp( const void *a, const void *b)
{
	return *((double*) a) - *((double*) b) ;
}

int main()
{
	double num[2000][2],p[2000][2];
	int n;
	for(int t=0;t<2000;t++)
	{
		p[t][0]=-20000+t*20;
		p[t][1]=0;
	}
	int data_serial,data_number=1;
	printf("data serial: ");
	scanf("%d",&data_serial);

	while(data_number<53)
	{
		double pi = M_PI ;

		char filein1[60],filein2[60];
		char first[20]={"20150513-"};
		char end1[20]={"-LED.dat"};
		char end2[20]={"-sphere.dat"};
		char serial[10];
	char number[10];
		sprintf(serial, "%d", data_serial);
		sprintf(number, "%d", data_number);

		strcpy(filein1,first);
		strcat(filein1,serial);
		strcat(filein1,"-");
		strcat(filein1,number);
		strcpy(filein2,filein1);
		strcat(filein1,end1);
		strcat(filein2,end2);
		FILE *fpt1;
		FILE *fpt2;
		fpt1=fopen(filein1,"r");
		fpt2=fopen(filein2,"r");

		double LED[3597][3];
		double sphere[3597][5];
		char data1[20];
		char data2[20];
		if(fpt1!=NULL)
		{
			fscanf(fpt1,"%s",data1);
			fscanf(fpt1,"%s",data1);
			fscanf(fpt1,"%s",data1);
			fscanf(fpt1,"%s",data1);
```

```
        int sum=0;
        for(int i=0;i<3597;i++)
        {

                      for(int j=0;j<3;j++)
            {
                             fscanf(fpt1,"%s",data1);
                      int e_location;
                            e_location=11;

                            data1[e_location]=0;

                            LED[i][j]=strtod(data1,NULL);

                            if(data1[e_location+1]=='-')
                            {
                                     LED[i][j]=LED[i][j]/pow(10,data1[e_location+2]-'0');
                            }
                            else
                            {
                            LED[i][j]=LED[i][j]*pow(10,data1[e_location+1]-'0');
                     }
            }
    }
}


if(fpt2!=NULL)    //input sphere data
{
          fscanf(fpt2,"%s",data2);
        fscanf(fpt2,"%s",data2);
        fscanf(fpt2,"%s",data2);
        fscanf(fpt2,"%s",data2);
        fscanf(fpt2,"%s",data2);
        fscanf(fpt2,"%s",data2);

        int sum=0;
        for(int i=0;i<3597;i++)
        {
            for(int j=0;j<5;j++)
            {
                     fscanf(fpt2,"%s",data2);
                      int e_location;
                            if(data2[0]=='-')
                     {
                      e_location=12;
                     }
                     else
                     {
                      e_location=11;
                            }


                            data2[e_location]=0;

                            sphere[i][j]=strtod(data2,NULL);

                            if(data2[e_location+1]=='-')
                            {
                              sphere[i][j]=sphere[i][j]/pow(10,data2[e_location+2]-'0');
                     }
                     else
                            {
```

```
                                        sphere[i][j]=sphere[i][j]*pow(10,data2[e_location+1]-'0');
                                }
                        }
                }
            }

            double sincos[3597][3], fdotv[3597];
            for(int i=0;i<3597;i++)
            {
               sincos[i][0]=LED[i][0];
               sincos[i][1]=LED[i][2]/sqrt(LED[i][1]*LED[i][1]+LED[i][2]*LED[i][2]);
                    sincos[i][2]=LED[i][1]/sqrt(LED[i][1]*LED[i][1]+LED[i][2]*LED[i][2]);
                    fdotv[i]=(2.21*980*(sphere[i][3]*sincos[i][2]+sphere[i][4]*sincos[i][1])*sin(pi/18))/240;
            }

            double J_tau[24];
             for(int N=0;N<24;N++)
             {
              J_tau[N]=0;
              for(int i=144*N+1;i<144*N+145;i++)
              {
                     J_tau[N]=J_tau[N]+fdotv[i];
                    }
                    J_tau[N]=J_tau[N]/0.6;
                    //printf("%d %f\n",data_number, J_tau[N]);
             }

             qsort(J_tau,24,sizeof(J_tau[0]),cmp);
             for(int i=0;i<24;i++)
             {
              J_tau[i]=J_tau[i]+20000;
              n=J_tau[i]/20;
              num[n][1]=num[n][1]+1;
             }

            data_number=data_number+1;
        }

        int sum=0;
        for(int i=0;i<2000;i++)
        {
              sum=sum+num[i][1];
        }
        printf("%d\n",sum);
        for(int i=0;i<2000;i++)
        {
              p[i][1]=num[i][1]/sum;
        }

        char fileout1[60];
            sprintf(fileout1, "possibility 20150513-8 tau=0.6s.txt");
       FILE* fptr1 = fopen(fileout1, "w");
       for(int j=0;j<2000;j++)
       {
            fprintf(fptr1,"%.0f %.0f %.10f \n",p[j][0],num[j][1],p[j][1]);
       }
       fclose(fptr1);

        system("pause");
        return 0;
}
```