An Optimized Fountain Model

Team 807 : B

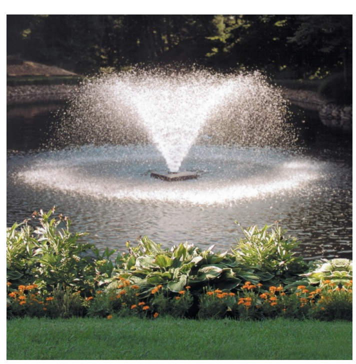
Abstract

Our task for the problem was to determine the probability density that any molecule ejected from the fountain hits the pond at the point (x, y), then the relevant problems can be solved. To achieve the goal,we first modeled the water flux as discrete small drops(with radius $10^{-3}m \sim 10^{-2}m$), and finally got the analytic solutions. In this simple model, we ignored the air resistance and interaction between water drops. We then considered the correction of drops' interaction by correcting ejection function E. Then we added a correction for the drag, After rough estimate of Reynolds number, we found that the air resistance acted on the water drops satisfies a quadratic form, we got numerical solutions to this problem with the help of MatLab.

Contents

1 Introduction

Projectile motion is a basic problem in kinematics, however, as the figure shows [4], when we consider the situation of plenty of water drops ejected from a water fountain in the air, we will be interested in the statistical result of these parabolas(like the drops' density falling into the pond), especially when air resistance correction and drops' interaction are included. In this paper, the ideal case's result of drop density in the pond is derived theoretically with arbitrary ejection distribution, then the correction analyses is followed since we cannot ignore the influence of air resistance and drops' interaction.



2 Models' Assumptions

In the first stage, we assume that there is no air resistance and interaction between drops. The analytic coordinates which drops fall down can be calculated. The ejection distribution function $\mathrm{E}(\theta, \varphi)$ and speed distribution function $f(v)$ in our model can be taken arbitrarily. The δ-like function $f(v)$ is applied to obtain analytical solution.

In the second stage, the correction from interaction between water drops can be partly included. We assume the fountain flux is small enough such that drops only interact with other drops at the beginning of ejection. Since drops diverge in the air, any water drop have less chance to reach other ones. Thus we ignore this part of interaction and only consider the correction of ejection function $\mathrm{E}(\theta, \varphi)$.

In the final stage, more practically, air resistance will be considered according to the Reynolds Number $\mathcal{R}$ of the drops in the air. We will see the result cannot be solved analytically. Numerical methods are used in this section. Water drops will be assumed to be spheres with radius range $10^{-3}\mathrm{m} \sim 10^{-2}\mathrm{m}$, we also set ejection velocity magnitude approximately be $10m/s$. Further analyses in this paper under NPT(normal pressure and temperature) will show the correction of r is not a ignorable quantity.

3 Models in vaccum

3.1 Kinematic Theory

Since the symmetry of the case, we prefer polar coordinates (r, ϕ) if we analyse the water drop distribution in the pond, and sphere-surface coordinates (θ, φ) is convenient to describe the ejection direction of the fountain. Because of the principle of kinematics,

$$\ddot{r} = 0 \tag{3.1}$$
$$\ddot{z} = -g \tag{3.2}$$

a parabola solution can be obtained by solving these differential equations, which is a well known fact that projectile motion's trajectory is a parabola. As the figure shows, we first consider a water drop ejected from (0,0) by velocity $\mathbf{v}_0$ along the direction (θ, ϕ) and hit the pond by velocity $\mathbf{v}_f$ at coordinate(r, ϕ),

$$v_{0r} = v_0 \sin\theta \tag{3.3}$$
$$v_{0z} = v_0 \cos\theta \tag{3.4}$$
$$v_{fr} = v_f \sin\theta' \tag{3.5}$$
$$v_{fz} = v_f \cos\theta' \tag{3.6}$$

we take a shortcut here for the benefit of the symmetry of the parabola, which is $\pi - \theta = \theta'$, we get:

$$v_{0r} = v_{fr} \tag{3.7}$$
$$-v_{0z} = v_{fz} \tag{3.8}$$

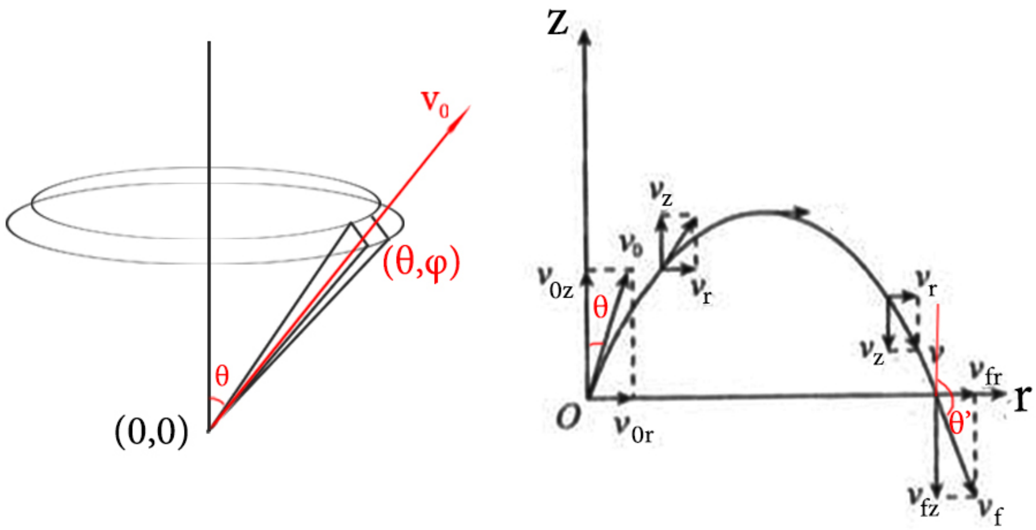


Figure 1: Drops ejection schematic diagram and parabola diagram

and by (3.2), $\dot{v}_z = -g$, hence

$$v_{fz} - v_{0z} = -2v_{0z} = \int_0^t -g\mathrm{d}t = -gt \tag{3.9}$$

therefore, time interval t between ejection and hitting in the pond is

$$t = \frac{2v_{0z}}{g} = \frac{2v_0 \cos\theta}{g} \tag{3.10}$$

and we find the mapping of $(\theta, \varphi) \mapsto (r, \phi)$:

$$r = v_{0r}t = \frac{v_0^2 \sin 2\theta}{g} \tag{3.11}$$

$$\phi = \varphi \tag{3.12}$$

The first equation shows a maximum distance of r is v_0^2/g. And for a particular r, it is possible to correspond two different θ which located in $[0, \pi/4)$ and $(\pi/4, \pi/2]$ separately.

If a arbitrary ejection flux distribution function $\mathrm{E}(\theta, \varphi)$ under a particular eject speed v_0 is considered, water drops ejected from $\mathrm{d}\theta\mathrm{d}\varphi$ hit $r\mathrm{d}r\mathrm{d}\phi$ (figure 3), we find

$$\mathrm{E}(\theta, \varphi)\mathrm{d}\theta\mathrm{d}\varphi = \rho(r, \phi)r\mathrm{d}r\mathrm{d}\phi = \rho(r, \phi)r\left|\frac{\partial(r, \phi)}{\partial(\theta, \varphi)}\right|\mathrm{d}\theta\mathrm{d}\varphi \tag{3.13}$$

where $\rho(r, \phi)$ is the drop distribution function in the pond, and $\frac{\partial(r,\phi)}{\partial(\theta,\varphi)}$ is the Jacobian determinant, according to (3.11) and (3.12)

$$\frac{\partial(r, \phi)}{\partial(\theta, \varphi)} = \frac{2v_0^2 \cos 2\theta}{g} = \frac{2}{g}\sqrt{v_0^4 - r^2 g^2} \tag{3.14}$$

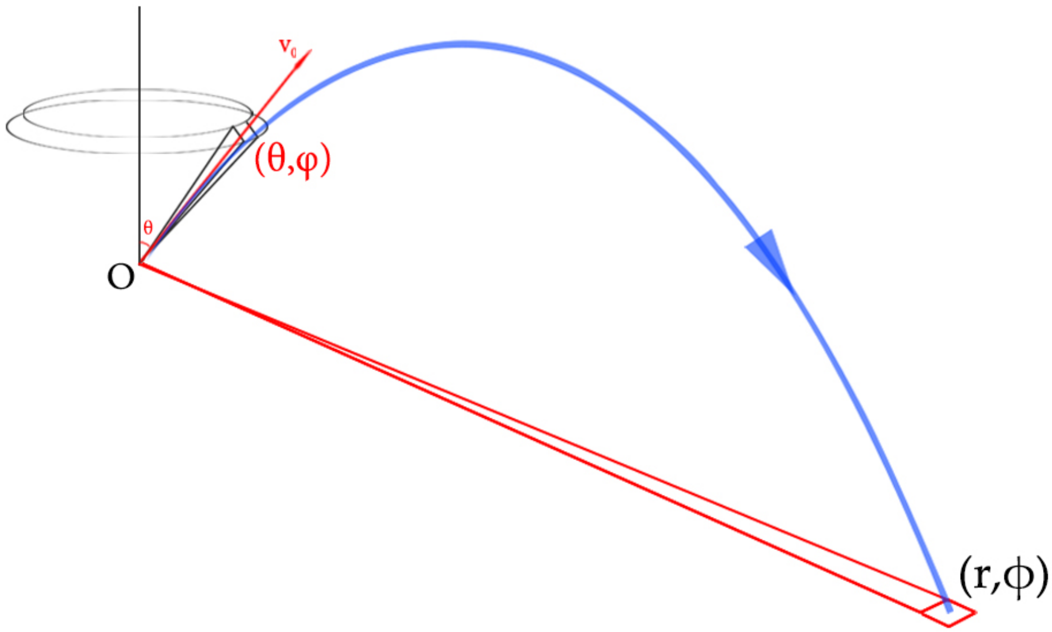


Figure 2: Drops ejected from general coordinates interval $\mathrm{d}\theta\mathrm{d}\varphi$ hit interval $r\mathrm{d}r\mathrm{d}\phi$

Notice that one particular r corresponds to two θ for most cases (except $\theta = \pi/4$), the inverse function of (3.11) can be written as two parts

$$\theta_1(r, v_0) = \frac{1}{2}\arcsin(rg/v_0^2) \tag{3.15}$$

$$\theta_2(r, v_0) = \frac{\pi}{2} - \frac{1}{2}\arcsin(rg/v_0^2) \tag{3.16}$$

here θ_1 takes value from 0 to $\pi/4$, while θ_2 takes value from $\pi/4$ to $\pi/2$. Hence, according to (3.13), $\rho(r, \phi)$ can be written as two parts of E:

$$\rho(r, \phi) = \frac{g[\mathrm{E}(\theta_1(r, v), \phi) + \mathrm{E}(\theta_2(r, v), \phi)]}{2r\sqrt{v_0^4 - r^2 g^2}} \tag{3.17}$$

Moreover, if the speed distribution function $f(v)$ of water-drop ejection is considered, $\rho(r, \phi)$ can be expressed by

$$\rho(r, \phi) = \int \frac{g[\mathrm{E}(\theta_1(r, v), \phi) + \mathrm{E}(\theta_2(r, v), \phi)]}{2r\sqrt{v^4 - r^2 g^2}} f(v)\mathrm{d}v \tag{3.18}$$

and (3.17) is just the special case of $f(v) = \delta(v - v_0)$.

Now, the equation of drops' distribution in the pond is obtained, which satisfies arbitrary angular ejection of the fountain. A step-like function $\mathrm{E}(\theta, \varphi)$ can be applied if we want to confine the fountain angular ejection.

3.2 Analyses of δ-like $f(v)$

Since we have too many freedom degrees to choose $\mathrm{E}(\theta, \varphi)$ and $f(v)$, moreover, the range of θ is uncertained for the fountain. In order to give examples, we choose a velocity magnitude distribution function:

$$f(v) = \frac{1}{2}\sqrt{\frac{\pi}{a}}\exp(-a(v-v_0)^2), \quad 0 < v < +\infty \tag{3.19}$$

First, we take $a \to +\infty$, that means $f(v)$ is converging to δ function:

$$f(v) = \delta(v - v_0), \quad 0 < v < +\infty \tag{3.20}$$

$v_0 = 10m/s$ is chosen as a example for following analyses. We also take $\mathrm{E}(\theta, \varphi)$ as:

$$\mathrm{E}(\theta, \varphi) = \begin{cases} 1, & 0 \le \theta \le \theta_0 \\ 0, & \theta_0 < \theta \le \pi/2 \end{cases} \tag{3.21}$$

We simply set function E as 1 where $0 \le \theta \le \theta_0$, which do not change the relative value of the distribution. Because function $\mathrm{E}(\theta, \varphi)$ is uniform respect to variable φ, $\rho(r, \phi)$ is uniform respect to ϕ. Respectively, we are interested in the $\rho - r$ graph. By taking different θ_0, we plot out Figure 3 by MatLab.

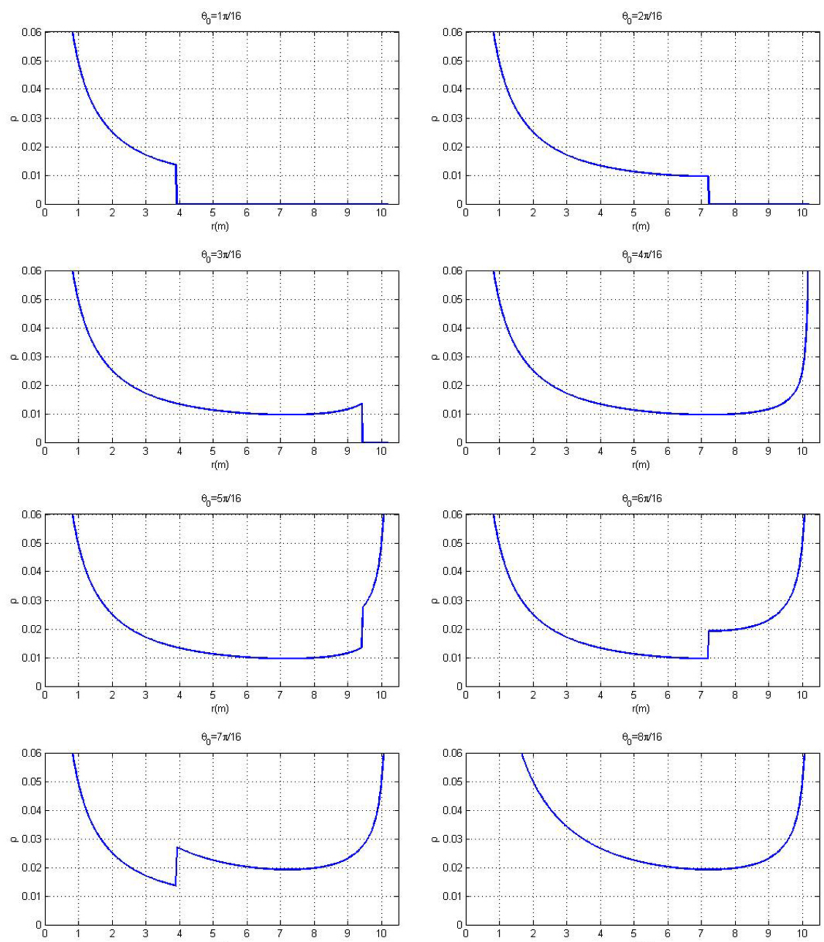


Figure 3: $\rho - r$ graph when $f(v)$ is a delta function,the unit for $\rho(\theta, \phi)$ is arbitrary, $\theta_0 = k\pi/16$, k=1,2,...,8

As (3.18) has predicted, Figure 3 shows us $\rho \to \infty$ when $r \to 0$ and $r \to v_0^2/g$ (when $\theta_0 \ge \pi/4$), which tells us **the most probable place for drops falling**

in is central point ($r = 0$) **and the boundary** ($r = v_0^2/g$). The unit here for ρ is arbitrary, and the distribution changes as θ_0 growing.

To determine the circle around the fountain inside which exactly half of the water lands, a analytic integral can be given according to (3.18)(3.20)(3.21):

$$\int_{\text{within r}} \rho(r,\phi) r \mathrm{d}r \mathrm{d}\phi = \int_0^r \frac{2\pi g \mathrm{d}r}{\sqrt{v_0^4 - r^2 g^2}} \tag{3.22}$$

where we assume $\theta_0 = \pi/2$. So the water percentage β lands in radius r is :

$$\beta(v_0, r) = \frac{\int_0^r 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}}{\int_0^{r_{max}} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}} = \frac{2\arcsin(r/r_{\max})}{\pi} \tag{3.23}$$

where $r_{\max} = v_0^2/g$ is known from (3.11). Let β be 1/2, then we can find

$$r_{\text{half}} = \frac{r_{\max}}{\sqrt{2}} = \frac{v_0^2}{\sqrt{2} g} \tag{3.24}$$

which characterized the median circle. For the example we given above, $r_{\text{half}} = (10m/s)^2/(\sqrt{2} \times 9.8m/s^2) \approx 7.215m$.

More generally, if θ_0 is random, general form of β can be derived. We denote $r_0 = v_0^2 \sin 2\theta_0/g$, then, for $0 < \theta_0 < \pi/4$

$$\beta(v_0, r) = \frac{\int_0^r 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}}{\int_0^{r_0} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}} = \frac{\arcsin(r/r_{\max})}{\arcsin(r_0/r_{\max})},\ r < r_0,\ \text{otherwise}, \beta = 1 \tag{3.25}$$

for $\pi/4 < \theta_0 < \pi/2$ and $r < r_0$, we have,

$$\beta(v_0, r) = \frac{\int_0^r 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}}{\int_0^{r_{\max}} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2} + \int_{r_0}^{r_{\max}} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}} = \frac{\arcsin(r/r_{\max})}{2\theta_0} \tag{3.26}$$

other wise, in other words, $r_0 < r < r_{\max}$

$$\beta(v_0, r) = \frac{2\int_0^r 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2} - \int_0^{r_0} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}}{\int_0^{r_{\max}} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2} + \int_{r_0}^{r_{\max}} 2\pi g \mathrm{d}r/\sqrt{v_0^4 - r^2 g^2}} = \frac{\arcsin(r/r_{max}) + \theta_0 - \pi/2}{\theta_0} \tag{3.27}$$

Hence, **general solution of r_{half} which characterized median circle can be solved by setting** $\beta = 1/2$ (under different situations):

$$(3.25) \rightarrow r_{\text{half}} = r_{\max} \sin(\frac{1}{2}\arcsin(\frac{r_0}{r_{\max}})) = r_{\max} \sin\theta_0 \tag{3.28}$$

$$(3.26) \rightarrow r_{\text{half}} = r_{\max} \sin\theta_0 \tag{3.29}$$

$$(3.27) \rightarrow r_{\text{half}} = r_{\max} \cos(\frac{\theta_0}{2}) \tag{3.30}$$

3.3 Correction of ejection interaction

It's obvious that water drops can interact each other. Momentum can be transmitted from a drop to another, which leads to a dispersion of velocity distribution. Actually, Gaussian distribution for ejection speed is a better approximation than δ function. a in (3.19) is chosen as 10 to describe dispersion in the following example.

$$f(v) = \frac{1}{2}\sqrt{\frac{\pi}{10}}\exp(-10(v-10)^2), \quad 0 < v < +\infty \tag{3.31}$$

For the reason that (3,18) cannot be integrated analytically, we obtain the result by MatLab: (Figure 4)

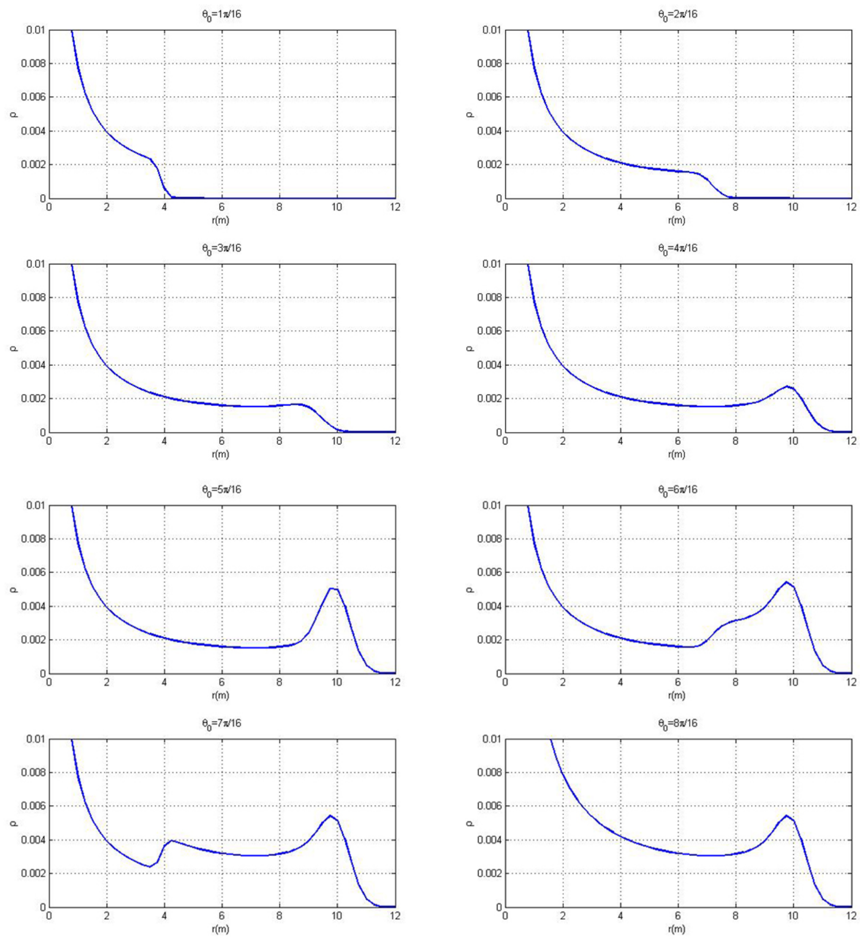


Figure 4: $\rho - r$ graph when $f(v)$ is a Gaussian function,with $a = 10$,the unit for $\rho(\theta,\phi)$ is arbitrary, $\theta_0 = k\pi/16$, k=1,2,...,8

We can see that the infinite divergence disappears when r is taken to be v_0^2/g, instead of infinity, a peak is nearby. This is due to the dispersion of Gaussian distribution, some water drops are able to breakthrough the 'boundary' v_0^2/g. For the sake of the dispersion of Gaussian distribution, $\rho(\theta,\phi)$ becomes continuous, which is quite different from Figure 3 when $\theta_0 \neq \pi/2$. But **the same point that both δ-like $f(v)$ and Gaussian-like $f(v)$ have is, the most probable place for drops falling in is the central point ($r = 0$) of the fountain**. ρ goes to infinity due to $\mathrm{E}(\theta,\varphi)$ is non-zero at $\theta = 0$, that makes $\rho \sim 1/r$ at $r = 0$.

Since the Figure 4 is the summary(integral) of different ejection speed, the water percentage β lands in radius r can be written as:

$$\widetilde{\beta}(r) = \int \beta(v,r) f(v) \mathrm{d}v \tag{3.32}$$

$\beta(v,r)$ can be seen from (3.25~27). **Obviously**

$$r_{\text{half}} = \widetilde{\beta}^{-1}(\frac{1}{2}) \tag{3.33}$$

which determines the median circle size.

4 Models with Air Resistance

According to Newton's 2nd law, if a drag force $F_d\hat{\mathbf{v}}$ of resistance applies on a droplet, the motion equation becomes:

$$m\dot{\mathbf{v}} = -m\mathbf{g} - F_d\hat{\mathbf{v}} \tag{4.1}$$

where $\hat{\mathbf{v}}$ is the unit vector along $\mathbf{v}$.

In order to determine the exact form of $F_d\hat{\mathbf{v}}$, the hydromechanics of air must be concerned. Since we set the radius a range of the water drops is 10^{-3}m $\sim 10^{-2}$m, and the ejection speed approximately be $10m/s$. Under the circumstance of NPT, we know the coefficient of air viscosity η is $1.57\times10^{-5}m^2/s$, air density ρ_a is $1.184\times kg\cdot m^{-3}$ [1]. We are able to find Reynolds Number:

$$1 < \mathcal{R} = \frac{2a\rho_a v}{\eta} < 10^5$$

hence, drag force has the quadratic form [1]:

$$F_d = \frac{C_D \rho_a A v^2}{2} \tag{4.2}$$

where C_D is a experiencing coefficient, which can be determined in Figure 5. It's fortunate to see that C_d correspond to our $\mathcal{R}$ is very stable (~ 0.5)

For sphere-like water drop, mass and cross section are:

$$m = \frac{4\pi a^3}{3}\rho_w \tag{4.3}$$
$$A = \pi a^2 \tag{4.4}$$

We denote k as $C_d A \rho_a/2$ and rewrite (4.1) as

$$\ddot{r} = -\frac{k}{m}\dot{r}\sqrt{\dot{r}^2 + \dot{z}^2} \tag{4.5}$$
$$\ddot{z} = -g - \frac{k}{m}\dot{z}\sqrt{\dot{r}^2 + \dot{z}^2} \tag{4.6}$$

We set two kinds of drops with radius $a_1 = 1mm$ and $a_2 = 5mm$. Then mass $m_1 = 4.189\times10^{-6}kg$ and $m_2 = 5.236\times10^{-4}kg$, while coefficient $k_1 = 9.3\times10^{-7}s^2\cdot m^{-2}$

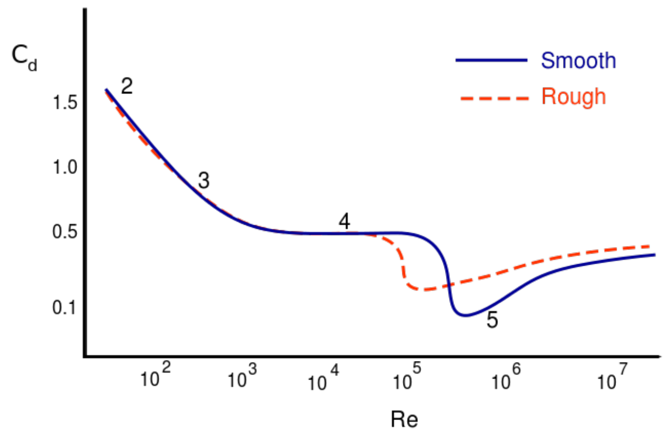


Figure 5: $C_D - \mathcal{R}$ graph [1]

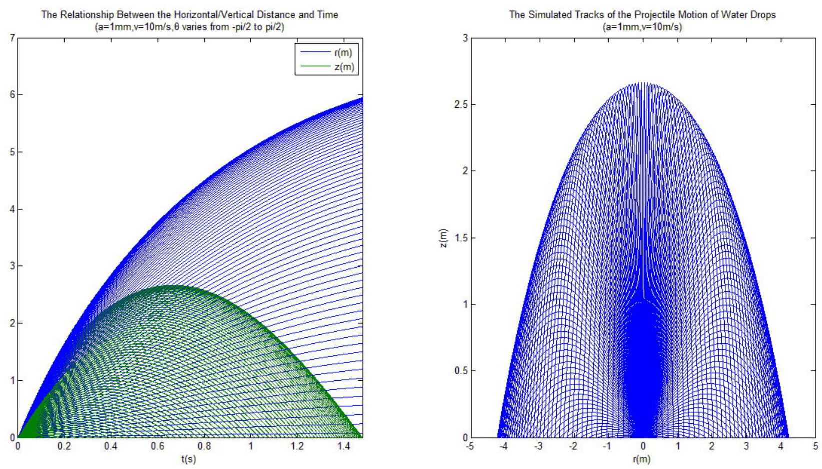


Figure 6: Drops radius $a_1 = 1mm$

and $k_2 = 2.325 \times 10^{-5} s^2 \cdot m^{-2}$.With the help of MatLab, we use RKF45 method(Runge-Kutta-Fehlberg Method) [3] to solve the differential equations (4.5) (4.6). Two sets of figures related to the motions of drops are displayed as follows.

It's remarkable that $r_{\max}$ is reduced to $8m$ for $5mm$ radius and even $4m$ for $1mm$ radius, which not a tiny correction for origin $r_{\max} \approx 10m$. As what we have done in section 3.2, we plot out histograms as $\rho - r$ graph when $\theta_0 = \pi/2$ (Figure 8 & 9). The result is similar to the last picture of Figure 3, since we only consider a particular speed $v_0 = 10m/s$ here (δ-like $f(v)$). **The most probable place for drop falling into the pond, is $r = 0$ and $r = r_{\max}$ in this case**. By sum up $\rho \, r \delta r \delta \phi$ by MatLab, water quantity lands within radius r can be determined. Since Figure 8 &

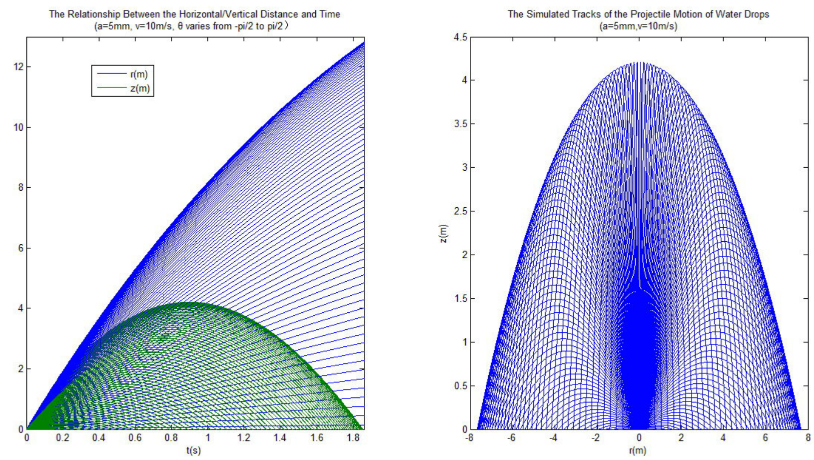


Figure 7: Drops radius $a_2 = 5mm$

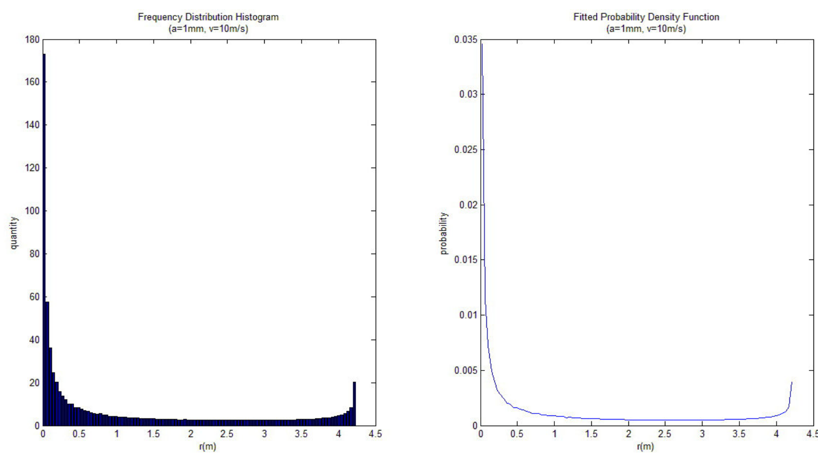


Figure 8: $\rho - r$ graph for Drops radius $a_1 = 1mm$

9 are un-analytical with discrete points, we take the closest point such that

$$\beta = \frac{\sum_{r=0}^{r_{\text{half}}} \rho\, r \delta r \delta \phi}{\sum_{\text{r}=0}^{r_{\max}} \rho\, r \delta r \delta \phi} \approx \frac{1}{2} \tag{4.7}$$

In this special case, **the radius of median circle, is**

$$r_{\text{half}} = 3.3024m \quad \text{for} \quad a_1 = 1mm \tag{4.8}$$

$$r_{\text{half}} = 5.6770m \quad \text{for} \quad a_2 = 5mm \tag{4.9}$$

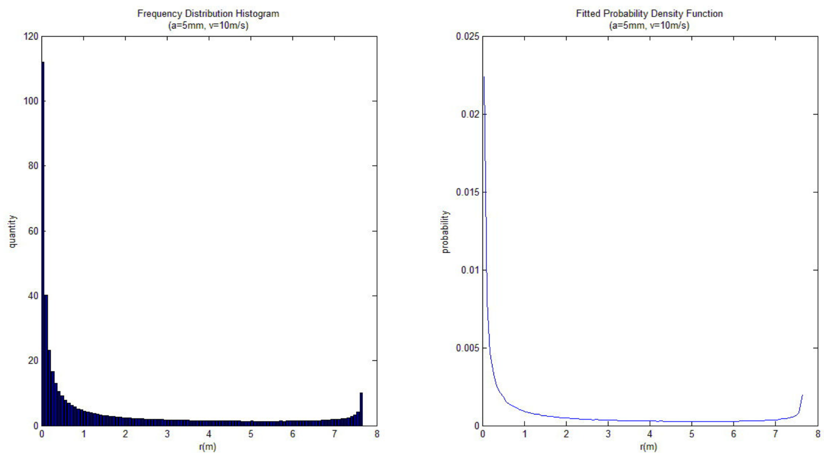


Figure 9: $\rho - r$ graph for Drops radius $a_2 = 5mm$

5 Discussion

5.1 The Weakness

1.Since we don't have a closed analytic solution of the two order differential equations,then we just obtained some special solutions for certain initial conditions.
2.We assumed that $E(\theta, \varphi)$ is uniform in θ and φ for the general model,thus our result might deviate from the actual situation.

5.2 The Strength

1. We use $F_d \propto v^2$ to simulate the drag force, actually $F_d \propto v$ is possible only for fountains whose height is around 0.02m, this tiny fountain is of course out of interest. Thus we obviously decrease the computational error.
2. We have discussed the proper evaluations of all the parameters and finally we provided the corresponding figures, it's intuitionistic to understand the effect of each parameter.
3. We simplified the interaction model and just concentrated on the main influence of water drops interaction, that's to say, we give up analyzing and calculating the mechanics behind the interaction, since multi-body-interaction will make this problem so complicated that no one can get a precise solution, thus we took a statistical view of the interaction and ignored the low probable collisions, treating the macroscopic effect of interaction as a Gaussian redistribution of ejecting velocity.
4. Based on the law of large number, we calculated the frequency distribution histogram by computer and then obtained an approximated density distribution of the water flux which is much complicated if we calculated directly.

References

[1] Wikipedia: http://en.wikipedia.org/wiki/Drag_%28physics%29

[2] Parker G W. Projectile motion with air resistance quadratic in the speed[J]. Am. J. Phys, 1977, 45(7): 606-610.

[3] John.Mathews, Kurtis D.Fink. Numerical Methods Using MATLAB. P469∼P471, Third Edition

[4] The fountain photo:http://images.hayneedle.com/mgen/imagebywidth.ms?w=500&minh=300&ma

A Appendix

A.1 Codes for section 3

A.1.1 Codes for δ-like $f(v)$

```
  %
g=9.8;a=10;v0=10;
r=[0:0.01:v0^2/g];
rho=zeros(length(r),8);
A=0.5*sqrt(pi/a);
syms v F
for k=1:8
    theta0=pi*(k)/16;
    if theta0<=pi/4 && theta0>=0
    for i=1:length(r)
        judge=(r(i)<=(v0^2*sin(2*theta0)/g));
        F=g/(2*r(i)*sqrt(v0^4-g^2*r(i)^2))*judge;
        rho(i,k)=vpa(F);
    end
elseif theta0>pi/4 && theta0<=pi/2
    for i=1:length(r)
        judge=(r(i)>=(v0^2*sin(2*theta0)/g));
        F=g/(2*r(i)*sqrt(v0^4-g^2*r(i)^2))*(1+judge);
        rho(i,k)=vpa(F);
    end
else
    warning('theta0 should be in the right range');
end
subplot(2,2,k);plot(r,rho(:,k),'-b','linewidth',2);grid on;axis([0,v0^2/g+0.3,0,0.06]
xlabel('r(m)');ylabel('\rho');title(['\theta_0=',num2str(k),'\pi/16'])
end
```

A.1.2 Codes for Gaussian-like $f(v)$

```
%plot out the graphs when v0=10, a=10, f(v) is Gaussian-like
g=9.8;a=10;v0=0;
r=[0:0.25:16];
rho=zeros(length(r),8);
A=0.5*sqrt(pi/a);
syms v F
for k=1:8
    theta0=(k)*pi/16;
if theta0<=pi/4 && theta0>=0
    for i=1:length(r)
        F=g/(2*r(i)*sqrt(v^4-g^2*r(i)^2))*A*exp(-a*(v-v0)^2);
        vc1=sqrt(r(i)*g/sin(2*theta0));
        rho(i,k)=vpa(int(F,v,vc1,+inf),5);
```

```
    end
elseif theta0>pi/4 && theta0<=pi/2
    for i=1:length(r)
        F=g/(2*r(i)*sqrt(v^4-g^2*r(i)^2))*A*exp(-a*(v-v0)^2);
        vc2=sqrt(r(i)*g/sin(2*theta0));
        vc3=sqrt(r(i)*g);
        rho(i,k)=vpa(int(F,v,vc3,+inf)+int(F,v,vc3,vc2),5);
    end
else
   warning('theta0 should be in the right range');
end
subplot(2,2,k);plot(r,rho(:,k),'-b','linewidth',2);hold on
grid on;xlabel('r(m)');ylabel('\rho');title(['\theta_0=',num2str(k),'\pi/16']);
axis([0,12,0,0.01]);
end
```

A.2 Codes for section 4

A.2.1 Codes for parabola

```
m=4.1888*10^(-6);k=9.3*10^(-7);g=9.8;
%parameter m=(water)(4*pi*a^3)/3,k=0.25*pi*a^2*rho(air).
ts=0:0.001:2;%time
for i=-pi/2:0.02:pi/2  %varying theta from -pi/2 to pi/2
z0=[0;0;10*sin(i10*cos(i)];
%initial value :r(0)=0 z(0)=0 r'(0)=10*sin(i) z(0)'=cos(i)
fun=@(t,z)[z(3);z(4);
    -k/m*z(3)*sqrt(z(3)^2+z(4)^2);
    -k/m*z(4)*sqrt(z(3)^2+z(4)^2)-g];
[t,z]=ode45(fun,ts,z0);
subplot(121),plot(t,z(:,1:2));
xlabel('t(s)');legend('r(m)','z(m)');
title({'The Relationship Between the Horizontal/Vertical Distance and Time';
    '(a=1mm,v=10m/s, varies from -pi/2 to pi/2)'})
axis([0 1.48 0 7 ])
hold on
subplot(122),plot(z(:,1),z(:,2));
axis([-5 5 0 3 ])
title({'The Simulated Tracks of the Projectile Motion of Water Drops';
    '(a=1mm,v=10m/s)'})
xlabel('r(m)');ylabel('z(m)');
hold on
end
```

A.2.2 Codes for $\rho - r$ graphs

```
function []=probability3
m=4.1888*10^(-6);k=9.3*10^(-7);g=9.8;%parameters
```

```
N=5000;  %Divide pi/2 into N parts.
ts=[0 2];
fun=@(t,z)[z(3);z(4);...
    -k/m*z(3)*sqrt(z(3)^2+z(4)^2);...
    -k/m*z(4)*sqrt(z(3)^2+z(4)^2)-g];
op=odeset('Events',@evenfun);
ang=linspace(0,pi/2,N);
x=nan(size(ang));
for i=1:length(ang);
    z0=[0;0;10*sin(ang(i)); 10*cos(ang(i))];
    [t,z,TE,ZE]=ode45(fun,ts,z0,op);
    if TE>eps,x(i)=ZE(1);end
end
x=x(~isnan(x));
[nw,xout]=hist(x,100);%Divide x into 100 parts.
subplot(1,2,1),bar(xout,nw./(2*pi*xout));
xlabel('r(m)');ylabel('quantity');
title({'Frequency Distribution Histogram';'(a=1mm, v=10m/s)'})
subplot(1,2,2),plot(xout,nw./(10000*pi*xout));
xlabel('r(m)');ylabel('probability');
title({'Fitted Probability Density Function';'(a=1mm, v=10m/s)'})
end
function [value,isterminal,direction]=evenfun(t,z)
    value=z(2);
    isterminal=1;
    direction=-1;
end
```

A.2.3 Codes for median circle

```
 function []=probability1

m=4.1888*10^(-6);k=9.3*10^(-7);g=9.8;%parameters.
N=5000;  %Divide pi/2 into N parts.
ts=[0 2];
fun=@(t,z)[z(3);z(4);...
    -k/m*z(3)*sqrt(z(3)^2+z(4)^2);...
    -k/m*z(4)*sqrt(z(3)^2+z(4)^2)-g];
op=odeset('Events',@evenfun);
ang=linspace(0,pi/2,N);
x=nan(size(ang));
for i=1:length(ang);
    z0=[    0;      0;   10*sin(ang(i)); 10*cos(ang(i))];
    [t,z,TE,ZE]=ode45(fun,ts,z0,op);
    if TE>eps,x(i)=ZE(1);end
end
x=x(~isnan(x));
```

```
subplot(1,2,1),hist(x,100)
xlabel('r(m)');ylabel('quantity');
title({'Frequency Distribution Histogram';'(a=1mm, v=10m/s)'})

[n,xout] = hist(x,100);
subplot(1,2,2),plot(xout,n/N);
xlabel('r(m)');ylabel('probability');
title({'Fitted Probability Density Function';'(a=1mm, v=10m/s)'})

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%Solve For Median Circle

half=0;
for p=1:1:100
  half=half+n(p);
  if half>2500
      break;
  end
end
area=max(x)
median=area*p/100
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% median value get
end

function [value,isterminal,direction]=evenfun(t,z)
    value=z(2)
    isterminal=1;
    direction=-1;
end
```