Abstract

In this paper we explore H. Aref’s blinking vortex system. We constructed an experimental apparatus to carry out the blinking vortex flow in a number of fluids under various conditions and parameters. We also numerically simulated the system and compared our results. We verified the advective, as well as the chaotic nature of the blinking vortex flow. We also characterized the Lyapunov exponent for glycerin at \( \sim 1000 \text{rpm} \)

*Keywords: advection, blinking vortex, Aref, simulation, chaotic*

1 Introduction

The study of the motion of fluids, known as *Fluid Dynamics* has wide range of practical applications, from understanding chemical processes, to the study of weather patterns and even in understanding nebulae formation in interstellar space.

*Advection* is the transport of substance by a fluid due to its motion. It differs from convection in that no diffusion takes place. Tracers are any fluid property that can be used to track the flow of a fluid. They can either be naturally occurring in the fluid, or be artificially introduced. *Chaotic mixing* is the process by which simple tracer fields in a mixture develop into complex fractals under the action of a fluid flow. Chaotic flows mean that small differences in initial position will lead to exponential diverging paths, i.e. the flows are highly sensitive to the initial conditions.

The *Lyapunov exponent* \((\lambda)\) can characterize this difference in path length over time, and give a measure of this chaotic nature. Here \(\Delta r_0\) gives the initial separation between two tracer particles and \(\Delta r(t)\) gives their separation at time \(t\).

\[
\Delta r(t) = \Delta r_0 e^{\lambda t}
\]
Chaotic dynamics can help to understand mixing in fluids, which is a widespread and important process. Until recently it has been hard to make the connection between these two things because of experimental limitations, but it is now possible to measure the stretching process that allows the connection between chaos and mixing to be understood deeply.

Another important quantity in studying fluids is the Reynolds number, a dimensionless number that gives the measure of the ratio of inertial forces to viscous forces in a fluid. A higher Reynolds number corresponds to more turbulent flow, while a lower Reynolds number indicates a more laminar flow. A laminar flow occurs where fluid flows in parallel layers without disruption between layers.

There are two main types of fluid mixing: Turbulent mixing and chaotic mixing. In both processes, there is stretching and folding of fluid elements, which causes nearby points to separate from each other irreversibly. **Turbulent mixing** is when random structures are produced by fluid instability at high Reynolds number (Re) and stretch and fold fluid elements. **Chaotic mixing** on the other hand is when laminar flows at modest Reynolds number can produce complex distributions of the material.

Only in recent years it been appreciated that a high Reynolds number (i.e. turbulent flow) is not necessary for complex particle trajectories in a fluid[4]. Hassan Aref in 1984 showed that a simple fluid setup – a blinking vortex flow – could be used to create laminar fluid flow that becomes chaotic[1].

## 2 Applications

Mixing of fluids is highly important process for many chemical and industrial processes. In many cases however, mixing occurs in a confined space where there is not much room for disorderly flow. In these kinds of cases, it is highly desirable to use a system such as Aref’s blinking vortex flow to efficiently mix two or more liquids while maintaining a laminar fluid flow. [3]

Chaotic flows can observed in many natural processes and system; in environmental flows such as the atmosphere and the ocean, where such irregular objects, like the unstable manifolds, are traced out by advected impurities or by biological species. This can be used to obtain new results in the field of chemical reactions and biological interactions of advected active tracers.

In recent times, this chaotic advection has been used in a number of fields to mix and blend materials together. The blinking vortex flow has been use in blending of polymers[8], creation of nanoscale structures[12] and more.
3 Theory

The blinking vortex system consists of two vortices that are equally spaced from the center of a cylindrical container. A vortex is a region in a fluid where the fluid spins around an imaginary axis. In the blinking vortex system, one of the vortices is turned on, while the other is off, and then they are reversed with the other being turned on while the first is turned off, and this cycle is repeated with a period $T$. This creates the blinking vortex effect, as described by Aref in his 1984 paper.[1]

A two dimensional blinking vortex flow can be described by the following equations [6]

$$
\dot{x} = -\frac{Ay}{x_s^2 + y^2} \\
\dot{y} = \frac{Ax}{x_s^2 + y^2}
$$

(2) (3)

Where $x_s$ is given by

$$
x_s = \begin{cases} 
  x + b & \text{if } 0 < t < T/2 \\
  x - b & \text{if } T/2 < t < T
\end{cases}
$$

(4)

Here $\dot{x}$ and $\dot{y}$ give the $x$ and $y$ velocities of fluid around the two vortices centered at $(\pm b, 0)$. One can see that since the $x$-velocity is dependent on the $y$-position and vice-versa, the motion of the fluid occurs in a circular fashion around the vortices. One can also note that as the position from the vortex $\sqrt{x_s^2 + y^2}$ increases, the velocity decreases.

For a single vortex, these equations reduce to

$$
\dot{x} = -\frac{Ay}{x^2 + y^2}; \dot{y} = \frac{Ax}{x^2 + y^2}
$$

(5)

Converting this to polar coordinates with $x = r \cos \theta$ and $y = r \sin \theta$, and squaring and adding together the two equations, we get

$$
\dot{x}^2 + \dot{y}^2 = -\frac{A^2(x^2 + y^2)}{(x^2 + y^2)^2}
$$

(6)

Since $x = r \cos \theta$ and $y = r \sin \theta$, $\dot{x} = -r \sin(\theta) \dot{\theta}$ and $y = r \cos(\theta) \dot{\theta}$

$$
\dot{x}^2 + \dot{y}^2 = r^2 \dot{\theta}^2 = \frac{A^2}{r^2}
$$

(7)

So, with $v = \omega r$, and $\omega = \dot{\theta}$,

$$
v = \left( \frac{A}{r^2} \right) r = \frac{A}{r}
$$

(8)
Figure 1: Circuit diagram showing DC motor controlled by Arduino via 5V relay (Based in part on schematic on arduino.cc)

4 Experimental Setup

This system was based in part on the experimental system described by the Applied Mathematics department at University of Colorado [7]. The experimental setup for my blinking vortex consists of two glass rods (0.25in in diameter) as the two vortices. These rods are connected to two high DC motors, with an input voltage rating of 0-18V with flexible joints. These joints are made by connecting the motor via a thin wooded dowel to the glass rod by heat shrinking layers of tubing over the connections, to create a flexible yet sturdy connection.

The motors are connected to the variable DC output of a power supply. Two rheostats are placed in series with the output to stabilize the voltage across the motors, which can fluctuate quite rapidly when switching between loads. Rheostats are used in place of standard resistors, because smaller resistors tend to burn out when the relatively large current flows through them. The rheostats also help to control the current flowing to each motor and ensuring that they both rotate at the same speed.

The two motors are controlled through two 5V relay switches that are switched on and off through the simple script running on the Arduino Mega 2560 board. The period of the motors can be changed with ease, but most of the cases we used a period $T$ between 20 to 60s. This range of values gives us sufficient time to record the movement created by one vortex before the effects of the other vortex are felt.
These glass rods were fully immersed in the fluid, which was placed in a 125mm diameter and 65mm deep Pyrex cylinder\footnote{Courtesy of Dr. Anes Kovacevic, Chemistry Department}. Since the motion of the fluid could not be tracked directly, we introduced light tracer particles in the fluid that would not interfere with its motion. After a number of potential tracer particles, we settled on using small plastic spheres in the range of 2mm to 5mm. The motion of tracer particles was recorded from the bottom of the glass container using one a Logitech HD Pro C920 webcam connected to a computer through Logger Pro.

To increase the visibility of these tracer particles, we use a black background so that the light tracer particles show up clearly. To reduce the friction of the glass rod on the support system, we used a system of high-speed mechanical ball bearings that support each rod at two points on either extreme of the rod (see figure ??). Care must be taken in aligning these bearings so that it does not induce additional vibrations into the setup, especially at high speed.

To accurately detect the speed of the motors and thus the vortices, we researched and investigated a number of different methods. Measuring the ripple generated across the DC motor due to the nature of the DC motor was one method, but did was not easy to process due to the small amplitude of this signal. Another method was using a stroboscope on a flag on the rod, but this interfered with data collection with the webcam. A third methods we tried was using slivering half the rod and measuring the current across the rod. However due to mechanical limitations, it was hard to maintain a steady contact between the wire and the rod, so we abandoned this method as well.

Due the various drawbacks of these methods, the final method we used was that based on a photo-transistor. A bright LED is placed on one side of the rod and a photo-transistor is placed on the other side, with the output connected to an oscilloscope. A opaque flag is placed on the rod. As the rod rotates, the flag interrupts the light falling on the photo-transistor, causing a spike in the output voltage. The frequency of these spikes is the frequency of the vortex. This signal was transmitted to an oscilloscope as well as an analog input port on the Arduino board. We build two of such setups, one for each of the vortices in use.

To process this data on the Arduino in real time, we used a simple algorithm which counted the number of times that the signal passed an experimentally determined threshold. If the signal passes the threshold nine times, it completes four periods (4T). Thus the frequency of the motors is given by

\[
f = \frac{1}{T}Hz = \frac{4}{(4T)}Hz = \frac{4 \times 60}{(4T)} \text{rpm}
\]

where rpm is the revolutions per minute.
Figure 2: Full setup as of 2/27/2013

(a) Glass rods supported by bearings.  
(b) Electronics control: Arduino Board and relays

Figure 3: Some parts of the experimental setup
Figure 4: A periodic function passes a threshold 9 times over 4 wavelengths.

Figure 5: Circuit setup of LED as well as phototransistor. The voltage probe lead is connected to both an oscilloscope as well as analog input of the Arduino.

Figure 6: Phototransistor based speed sensor.
5 Simulation

We used a MATLAB program to numerically solve the coupled differential equations describing the blinking vortex flow. Since the functions are defined piecewise, the differential equations are solved in segments, and the final state of one solution is used as the initial condition for the next segment to create a continuous solution. The differential equation solver used was the inbuilt \texttt{ode45()}, which uses a variable step Runge-Kutta Methods to solve the differential equations.

The simulations tracks a single particle over time. To see the complete trace of the particle over time, we conglomerated all these points to get figure 7 which describes the path taken by a particle over some period. The details of the implementation can be seen in the appendix.

![Figure 7: Results of theoretical simulation for a particle over three periods. Note the colors refers to the motion undergone due to a particular vortex, in this case blue for the left and red for the right.](image)

6 Data

To record data from this system we used a camera positioned below the setup. The camera recorded video data which was analyzed with aid of LoggerPro software. The motion of the particles was calibrated and traced with the aid of this software. The constant in
equation $A$ was determined using the fit for the equation of $v$ vs $r$. Note that this constant depends on the speed of the vortex, which can be changed.

The frequency used in the system was usually between 1000 – 1200rpm. Though we tested frequency ranges between 500 rpm and 6000 rpm, we found that a lower frequency was more stable. At higher frequencies the vibrations in the vortices build up and generated bubble in the liquid used.

The period used for the blinking vortex flow ranged from 1 s to 1 min, though for most of our tests we used a period of 20s, i.e. 10 second for each motor. This value was mainly based of obtaining sufficient data to record the position of the tracer particles using video data at 15 frames per second.

In most cases liquid used was glycerin, though we tested other fluids such as corn syrup and water. The corn syrup used had the highest viscosity, though it tend to form a thick hard film on top, which is why glycerin was used instead. water too was tried, but the viscosity of the water was so low that the vortices did not rotate the fluid at all.

Before we began testing our blinking vortex flow, we wanted to check the validity of the equations that we assumed to be true. The first thing that we did was to verify that the $v$, velocity of the fluid (or in our case, a tracer particle in the fluid) decreased as $1/r$ as we increased the radius from the vortex $r$. For a single vortex at a constant speed, we traced 8 particles across multiple periods to find values for $v$ and $r$. The results of this are shown in figure ??.
Another test that we conducted was to check the advective nature of this system. If it was truly advective, the system would return to its original state upon time reversal. Experimentally this means that if we reversed the motion of the vortices, a tracer particle

To accomplish this, we ran the system for a half integer period, such that the system stopped spinning the same vortex that it started on. In this way, when running the second time, the reversing switch shown in figure 10 could be flipped such that the polarity of the DC motors would be reversed, and thus the whole system would run backwards.

Theoretically these conditions would be exactly opposite to those in the forward direction, but due to mechanical limitations, it is not quite possible. Nevertheless we attempted to get as close as possible to the initial state in this manner. The results of our experiment are shown in figure 11 after 3 sets of rotations (1.5 periods).

Figure 9: Velocity as a function of radius for a single vortex rotating at 1200 rpm.

Figure 10: Schematic for a reversing switch.
Figure 11: The x-position of a particle upon reversal after 1.5 periods in a blinking vortex flow at \( \sim 1000 \text{ rpm} \)

Figure 12: The y-position of a particle upon reversal after 1.5 periods in a blinking vortex flow at \( \sim 1000 \text{ rpm} \)
Figure 13: Exponential separation of two particles with similar initial conditions in a blinking vortex flow with vortices spinning at 1000 rpm

\[ y = 0.4509e^{1.0354x} \]
\[ R^2 = 0.9527 \]

Figure 14: The separation between the same two particles over a longer period of time. Notice that the exponential trend only continues for a short period at the fluid is contained.
7 Results and Conclusion

We successfully constructed an experimental blinking vortex flow. We have measured position of test tracer particles in a variety of fluids at various velocities over time. The parameters of the blinking vortex system, such as period, size etc can easily be modified. The system has been shown to be advective in nature, as it can be reversed. We have also verified the equations that we assumed to be true, both using our numerical simulation, as well as with experimental data.

We have shown that the position of the particles deviates exponentially (figure 13) and have characterized the Lyapunov exponent. For glycerin at ~1000rpm this dimensionless parameter is 1.0354. The separation over longer period is not exponential (as shown by figure 14), and this is because the system is contained, and thus while the particles might diverge exponential over a small period, over longer timespan the separation between any two particles is limited and thus will oscillate over time.

The blinking vortex system is thus a successful system with which to accomplish chaotic mixing. This is an advective system, where particles with small initial separation will diverge exponentially. It has a slew of applications, mainly in industry, for the mixing of materials both large and small.
References


8 Appendix: Code

Listing 1: "Blink Flow"

/*
  Blinking motors

  This script turn one output high, then the other with a delay of 1 second.
*/

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:

// Output pins are 4 and 12
int out1 = 4;
int out2 = 12;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(out1, OUTPUT);
  pinMode(out2, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(out2, LOW); // turn the LED on (HIGH is the voltage level)
  digitalWrite(out1, HIGH); // turn the LED off by making the voltage LOW
  delay(30000);  // wait for 10 seconds
  digitalWrite(out1, LOW); // turn the LED off by making the voltage LOW
  digitalWrite(out2, HIGH); // wait for 10 seconds
}

Listing 2: "Read Speed from Sensor"

/* Read Analog Voltage of one motor and prints frequency of
signal from analog input to serial via MegunoLink*/

// digital output pins of the two DC motor relay controls
int motor1 = 4;
int motor2 = 12;

// middle of voltage signal for frequency reading
int threshold = 35;
/ *misc vars*

```c
int flag = 0, old_flag = 0, count = 0;
unsigned long time, t1;
double freq = 0;
```

// runs at reset

```c
void setup() {
    // high output rate (bps)
    Serial.begin(500000);
    pinMode(motor1, OUTPUT);
    pinMode(motor2, OUTPUT);
}
```

```c
void loop() {
    digitalWrite(motor1, HIGH); // start one motor
    int sensorValue = analogRead(A0); // read analog pin 0

    // save old value for comparison
    old_flag = flag;

    // check if crosses
    if (sensorValue > threshold)
        flag = 1;
    else
        flag = 0;
    // 9 flags 4 wavelengths
    if (flag != old_flag)
    {
        if (count == 0)
        {
            time = micros();
            count++;
        }
        else if (count == 8)
        {
            t1 = micros() - time;
            freq = 240000000.0 / ((double)t1);
            Serial.println(freq);
            count = 0;
        }
        else
        {
            count++;
        }
    }
    //sendPlotData("voltage",sensorValue);
```
sendPlotData("freq", freq); // rpm
//sendPlotData("time", t1);
//sendPlotData("count", count);
//sendPlotData("flag", flag);

// float voltage = sensorValue; // * (5.0 / 1023.0); // scale read value
// Serial.println(voltage);
//sendPlotData("voltage", voltage);
}

void sendPlotData(String seriesName, float data){
    Serial.print("{");
    Serial.print(seriesName);
    Serial.print(",T, ");
    Serial.print(data);
    Serial.println("}");
}

Listing 3: "ODE Simulation"

% BLINKING VORTEX FLOW
% Simulate the two coupled differential equations for a blinking vortex
% flow
%
% x' = -Ay/(xs^2 + y^2)
% y' = Axs^2/(xs^2 + y^2)
%
% xs = {
%   x - b   if 0 < t < T/2
%   x + b   if T/2 < t < T
%
% Solving Notes: Use ode45, Solve for each little interval, use final
% conditions for each next initial condition

function senior_ode_fullanimation()
    % Main Driver Function

    % Initial conditions
total_time = [0, 20.5];
initial_pos = [1;1];
global A b;
A = 1; b = .5;

    count = 1;

    for i=1:5
% Solve first set [Function f(t) = x(t), y(t)]
[t, f] = ode45(@vortex1, total_time, initial_pos);
for j = 1:size(f,1)
    plot(f(j,1), f(j,2), 'o');
    hold all;
    M(count) = getframe;
    count = count + 1;
end
final_pos = [f(end,1), f(end,2)];
% Start second set with first initial conditions
[t, f] = ode45(@vortex2, total_time, final_pos);
for k = 1:size(f,1)
    plot(f(k,1), f(k,2), 'o');
    hold all;
    M(count) = getframe;
    count = count + 1;
end
initial_pos = [f(end,1), f(end,2)];
end
movie(M, 1, 100);
end

function dydt = vortex1(t, y)
% Vortex 1 (right)
% y(1) = x, y(2) = y
global A b;
  dydt = [(A*(-y(2)))/((y(1)-b)^2 + y(2)^2); (A*(y(1)-b))/(y(2)^2+(y(1)-b)^2)];
end

function dydt = vortex2(t, y)
% Vortex 2 (left)
% y(1) = x, y(2) = y
global A b;
  dydt = [(A*(-y(2)))/((y(1)+b)^2 + y(2)^2); (A*(y(1)+b))/(y(2)^2+(y(1)+b)^2)];
end