

CEE 4530 Final Research Project: Gravity-Driven CMFRs In Series

Group 11: Benjamin Lee, Wenduo Nie, and Wesley Sluga

May 12th, 2019

Introduction

Clean, potable water is not easily accessible in all parts of the world. Cornell AguaClara team puts great effort in designing low-cost water treatment systems that can be implemented anywhere. The key to such designs include: no outside electricity inputs, simple materials that are easily replaceable, can be locally assembled, and are flexible enough to accommodate different environments. One method of water treatment is to create a series of completely mixed flow reactors (CMFRs). The CMFRs in series can be used to create a steady-state flow that allows pollutants to get in contact with treatment chemicals and outputs treated water. However, a CMFR requires some form of mechanical stirring, which makes it difficult or sometimes impossible to implement in rural or remote areas that lack consistent energy grids.

Thus, this experiment aimed to explore if it's possible to create a gravity-driven CMFR series with potential energy that results from elevation difference between reactors that can replicate the effects of a mechanically-driven CMFR. If proven feasible, it will align with the goals of AguaClara mentioned above by making CMFR in series water treatment accessible in remote areas. This is because the mechanical mixing segment of water treatment could be replaced with gravity-driven CMFRs and still be just as effective, even without the need for outside energy sources.

This exploration was centralized around the dynamics of fluid flow and mixing characteristics of basins. Based on our previous course experience, we hypothesized that it was possible to generate enough energy through potential energy height differences between reactors so that the stirring mimics a mechanically mixed CMFR, but that height difference will be significantly larger than the height of the water level in the reactor.

Objectives

If there is an elevational change between reservoirs, a gravity-induced water jet can be used to mix the next reactor, thus forming a gravity-driven CMFR. In order to evaluate how high the reactors need to be elevated, we first needed to examine how the energy dissipated from an eddy relates to the height since the energy dissipated can be related to how well mixed a reactor is.

The first important characteristic to design was the eddy turnover time, which is (1) $t_{eddy} \approx \frac{L_{eddy}}{v_{eddy}}$. We wanted $t_{eddy} = 0.1\theta$ so that the mixing occurred fast enough that dye doesn't immediately flow out of the effluent, leading to short circuiting. The other important parameter is the rate at which energy is dispersed from an eddy given by: (2) $\bar{\epsilon} \approx \frac{v_{eddy}^3}{L_{eddy}}$ and (3) $\bar{\epsilon} \approx \frac{v_{jet}^2}{2\theta}$.

After setting the two equations equal to each other, we got: (4) $\frac{v_{jet}^2}{2\theta} = \frac{v_{eddy}^3}{L_{eddy}}$

We rearranged the eddy turnover time (equation 1) to get (5) $v_{eddy} = \frac{L_{eddy}}{t_{eddy}}$. Substituting equation 5 into equation 4 yielded: (6) $\frac{v_{jet}^2}{2\theta} = \frac{L_{eddy}^2}{t_{eddy}^3}$

Using the condition that $t_{eddy} = 0.1\theta$, we got: (7) $\frac{v_{jet}^2}{2\theta} = \frac{L_{eddy}^2}{.1^3\theta^3}$

We knew that (8) $v_{jet} \approx \sqrt{2gh} + \frac{Q}{A}$, which is the sum of velocity from potential energy plus the rate of pumping, where Q is the flow rate and A is the area of tubing. Substituting equation 8 into 7 gave us: (9) $\frac{(\sqrt{2gh} + \frac{Q}{A})^2}{2\theta} = \frac{L_{eddy}^2 * 10^3}{\theta^3}$

Solving equation 9 for h: (10) $\sqrt{2gh + \frac{Q}{A}} = \sqrt{\frac{L^2 * 10^3}{\theta^2}}$ We see that $h \propto L^2$ and since the reactor is less than a meter tall, the height difference between reactors necessary to generate mixing only needs to be within a water height of each other. This indicates that our hypothesis is plausible but excessive, as we thought it was possible but would require multiple water heights to generate enough energy.

However, this only accounts for the head required to mix, not to overcome losses from reactor to reactor. Therefore, we found the major and minor head losses for our reactor system two primary equations. First, the Darcy-Weisbach equation for major losses:

$$(11) h_f = f \frac{L}{D} \frac{v^2}{2g}$$

And second, the minor losses equation:

$$(12) h_m = \sum K \frac{v^2}{2g}$$

Combining the three head values, we found the total minimum head required to generate mixing within each reactor.

Procedure

We tested if this works by creating a series of 3 reactors and measuring the effluent concentration of red dye #40. First, we varied the height between the reactors, in an attempt to create an effective CMFR that replicates that of mechanical stirring (which will serve as the control). Second, we explored how well the epa code was able to model our system of 3 reactors in a row. We expected that increasing the elevation differences between reactors would be able to generate enough mixing to be close to a CMFR and that the models would accurately follow the data and predict the correct number of reactors in series.

Shown below are the experimental diagrams we used to set up our control and experimental diagrams respectively.

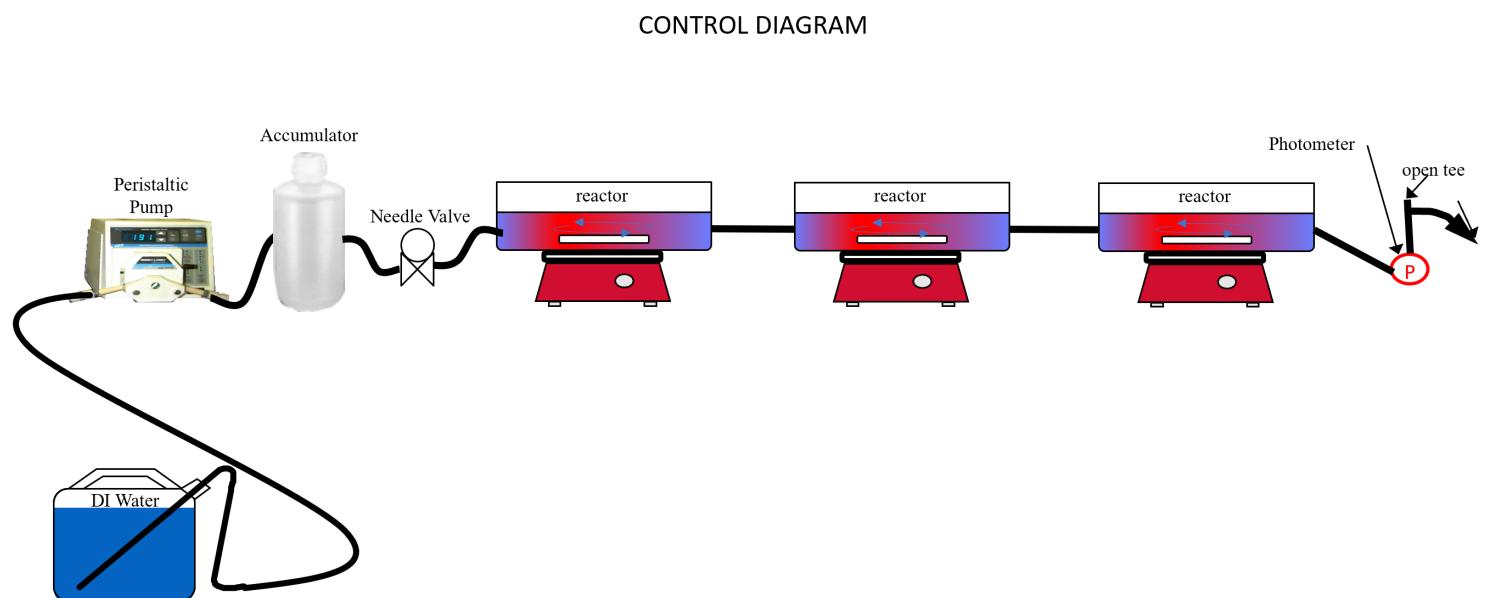


Figure 1. Control Diagram

EXPERIMENTAL DIAGRAM

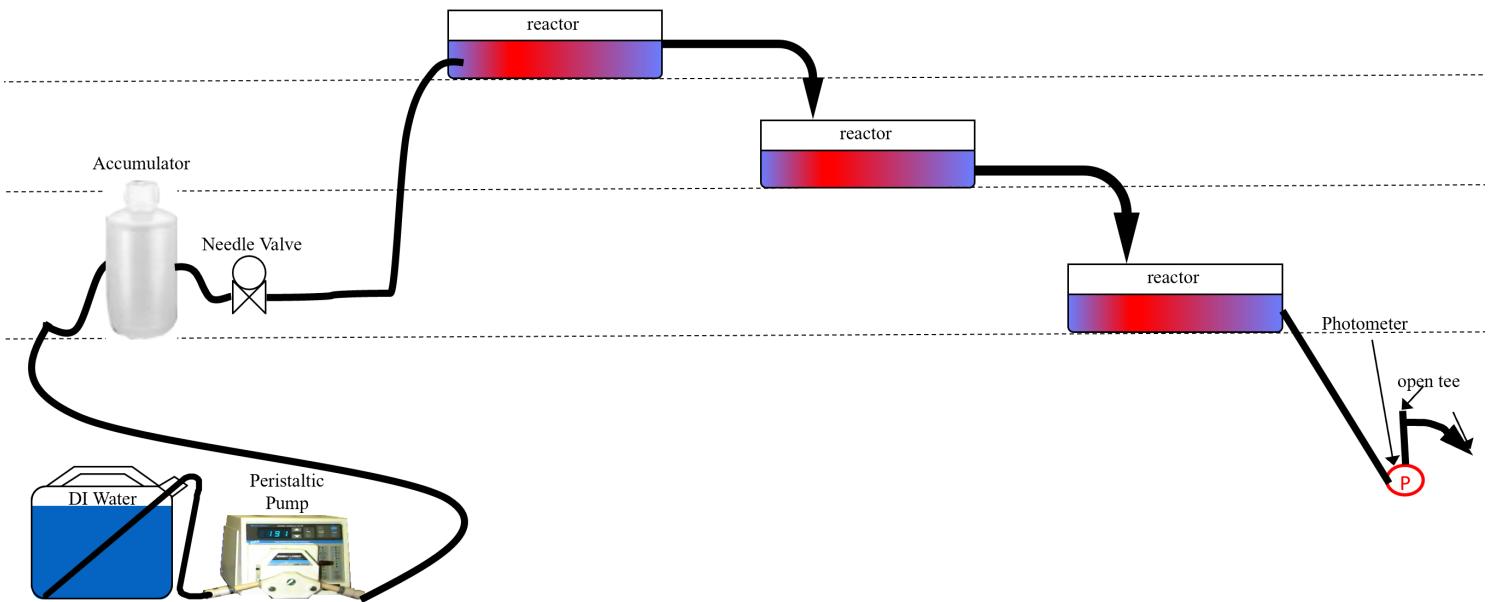


Figure 2. Experimental Diagram

Similar to the reactor characteristics lab, we injected red dye into our reactor system at the influent location and measured the concentration at the effluent using a photometer.

1. To set up the lab, first, the photometer was connected to ProCoDA. The photometer was calibrated and used to measure the voltage, and in turn the concentration of red dye, accurately. Then ProCoDa generated a graph and the data was stored into a tsv file.
2. We gathered three ractors, three stir bars and stirrers, a peristaltic pump, a jerry can of deionized water, tubing, stands, clamps, 100g/L red dye and pipettes.
3. As shown in figure 1, we started the experiment with the control group. Three reactors were each sitting on a stirrer on the same elevation level.
4. We used stir bars in all reactors for mixing. We used tubes to connect from the distilled water to the pump, then to the three reactors in series. The inlet of the photometer takes in solution from the last reactor in series. The water level in all three reactors were kept at 2L.
5. We turned on the stirrers and used pipettes to obtain 1ml of the 100g/L red dye, because we wanted the red dye concentration to be within the measurable range of the photometer.
6. The pump was turned on and set at a flow rate of 10 rpm. The red dye was injected near the influent of the first reactor in series.
7. We can observe that the red dye slowly goes from the first reactor to the last reactor. We kept the equipment running for a couple hours and had ProCoDA record all the data.

After finishing the control group, we moved on to the experiment group 1, 2 and 3. All the equipments and set up were similar to the control except that we removed the stirrer and stir bars and put some buckets or boxes underneath reactors to create elevation differences as shown in figure 2. Instead of the

10 rpm flow rate we used in the control group, the pump was set at 100 rpm for the three experiments. The control rate was set at 10 rpm to minimize head loss, as the control had no elevational difference. When ran at the higher 100 rpm, the first tank filled with water just to overcome the head loss and backpressure to flow into the next tank, so we scaled down flow rate by 10 to minimize this impact. The volume of the water was still kept at 2L in each reactor, meaning the only differences between experiment group 1, 2 and 3 were the height difference. The height difference for the first experiment was 27.5 cm, which we calculated to be 3.5 times the total required head and the upper limit of the lab bench.

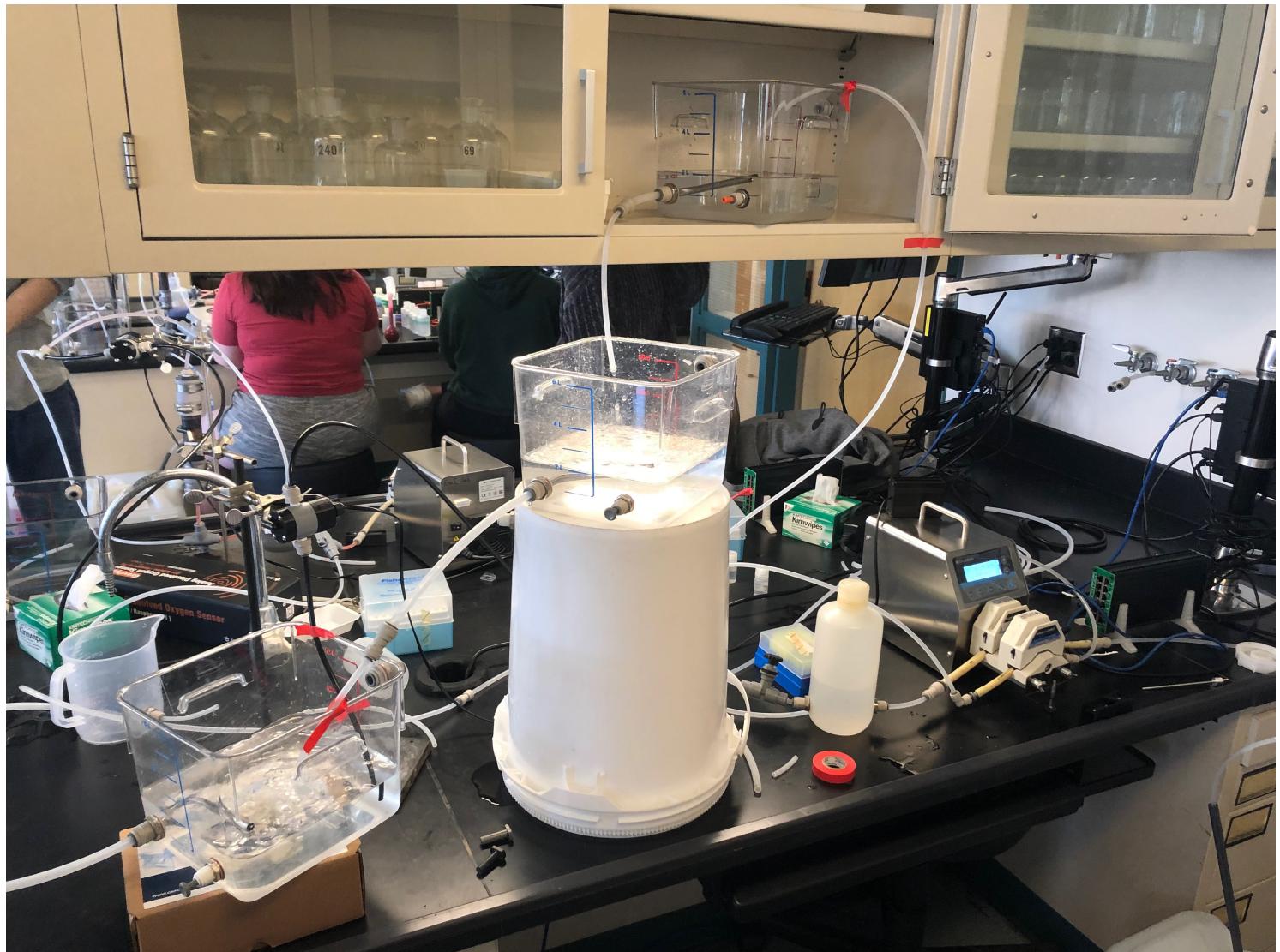


Figure 3. 27.5 cm Height Difference Experiment Set Up

The height difference for the second experiment was 15 cm, or about twice the total required head.

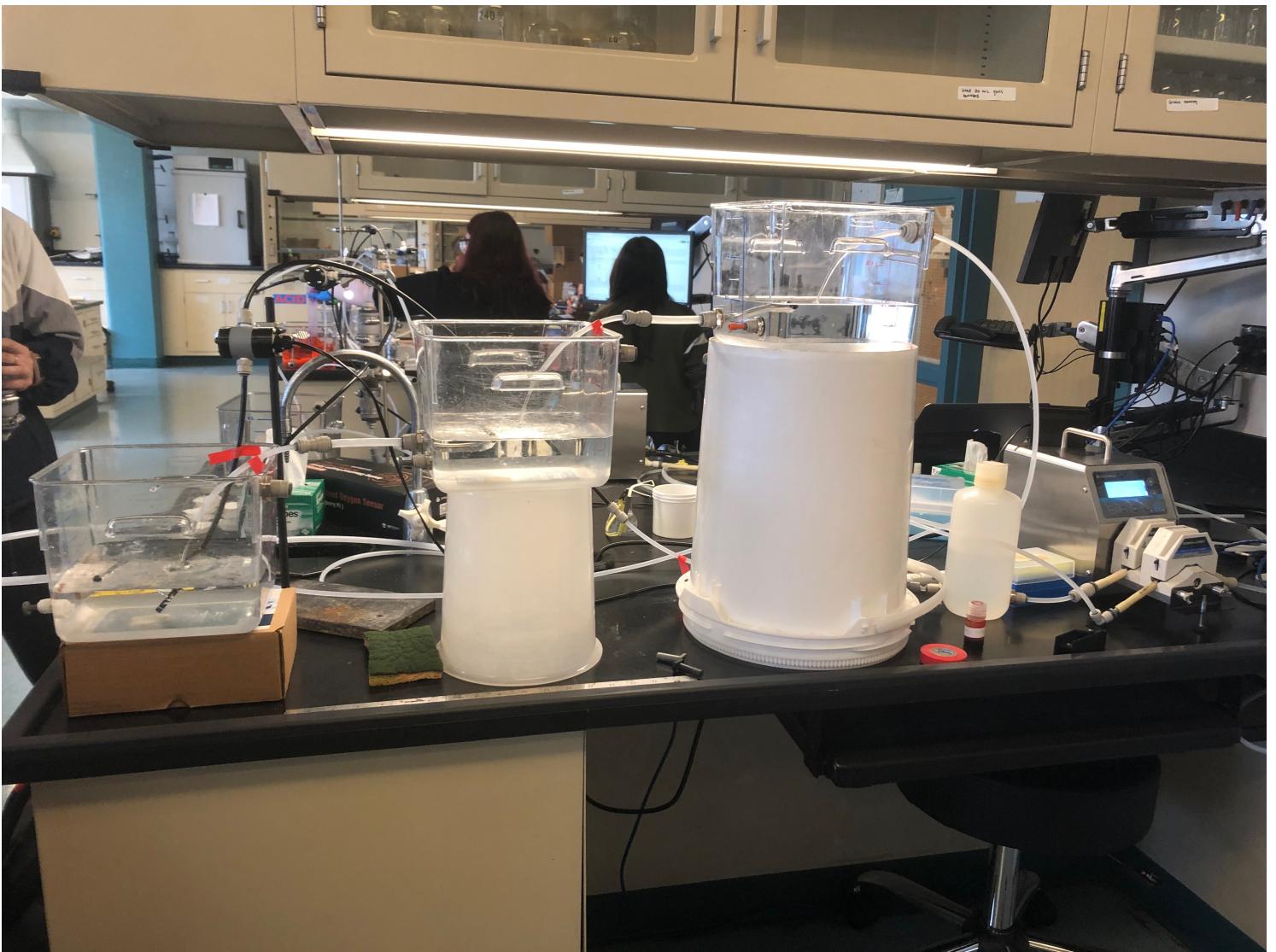


Figure 4. 15 cm Height Difference Experiment Set Up

The height difference for the third and final experiment was 8 cm, which was the absolute minimum requirement to account for total head loss and mixing.



Figure 5. 8 cm Height Difference Experiment Set Up

After setting up, all the rest procedures were the same as the control group. After gathering all the data, we used Python to analyze the data file in order to calculate the residence time, equivalent number of CMFRs, and the t^* value for which $F = 0.1$.

Cleaning Procedures

Between each experiment, we cleaned the reactors to ensure that there was not any red dye left that would affect the succeeding experiments.

1. After we have ended one run, we turned off the pump and closed off the accumulator.
2. We unlocked the pump clamps to allow the reactors to be moved.
3. We emptied all reactors of the remaining mix.
4. We setup the next height difference we wanted to measure and filled the reactors back up to 2L.

5. We began pumping water again for 5 minutes to re-calibrate the accumulator, check for leaks, and ensure that the water stayed at 2L for each reactor.

Results and Discussion

In our control run, we created a series of CMFRs with mechanical stirring to establish a baseline and an ideal goal to reach. We found that our control aligned quite nicely with the modeled CMFR that we created in Python, see Figure 6 below.

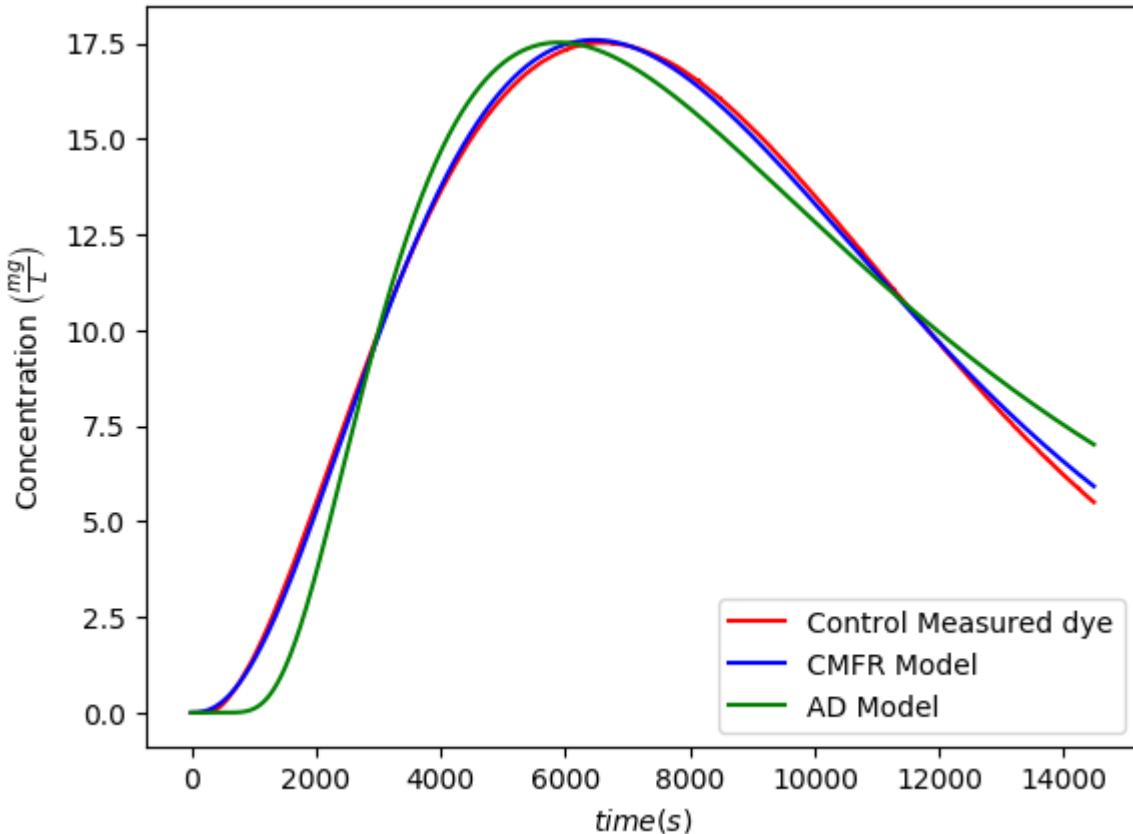


Figure 6. Control Experiment Data vs Models

Based on the calculations, the code estimated from the data that our CMFR was approximately 3.5 CMFRs in series, which is quite close to the actual 3 CMFRs. Our residence time was 7400 seconds, which is just over two hours due to the very slow flow rate utilized to minimize head loss.

In our first experiment with the 27.5 cm height difference, the data seems to resemble the AD and CMFR model at different points. It initially starts in between the two models and tracks the CMFR model until the decay where it then begins to follow the AD model. This can possibly be explained by dead volumes, leading it to not be well mixed in some sections.

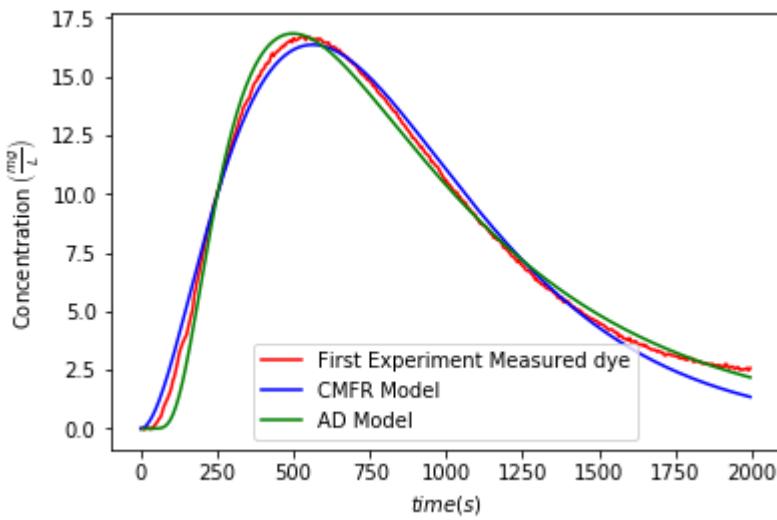


Figure 7. 27.5 cm Height Difference Data vs Models

We found the tracer residence time to be 640 seconds (10 and half minutes). This equates to roughly 2.94 CMFRs in series, which was even closer to our control of 3 CMFRs in series than the actual control run.

In our second experiment with the 15 cm height difference, our data tracks the CMFR model very closely.

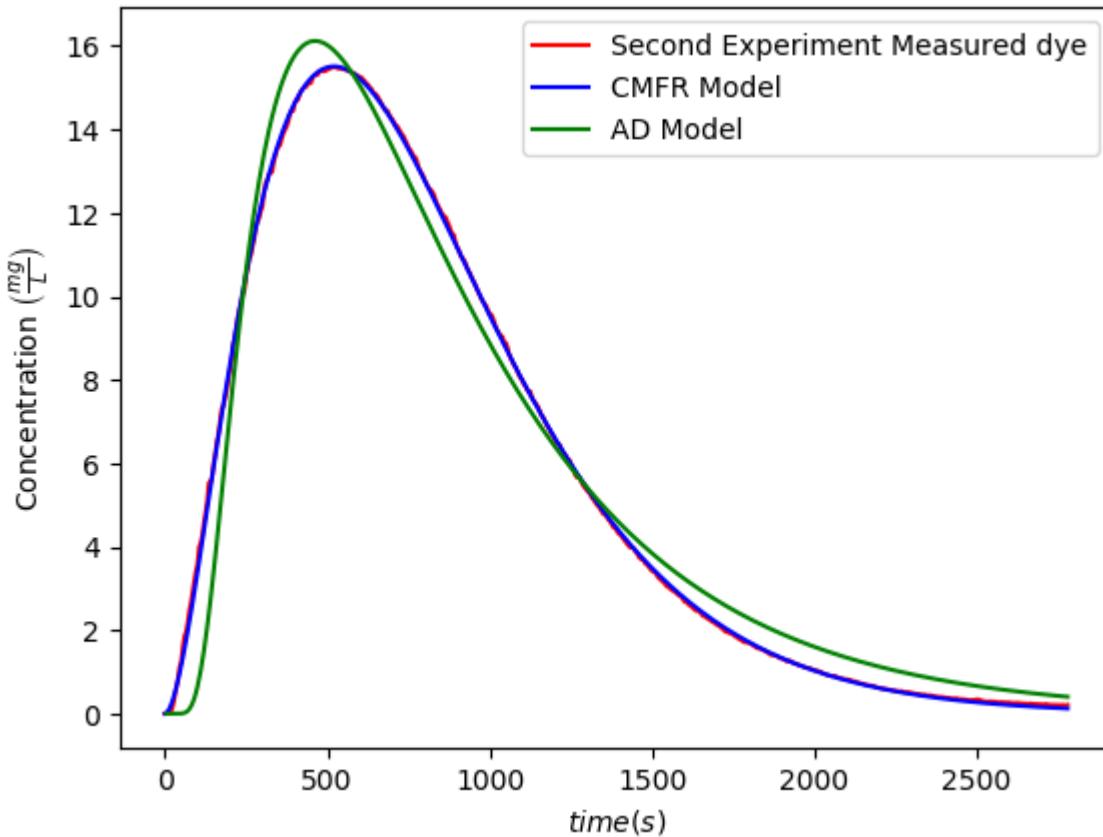


Figure 8. 15 cm Height Difference Data vs Models

We found the tracer residence time to be 590 seconds (9 and three quarter minutes). This equates to roughly 2.81 CMFRs in series.

In our third experiment with the 8 cm height difference, our data, again, tracks the CMFR model very closely.

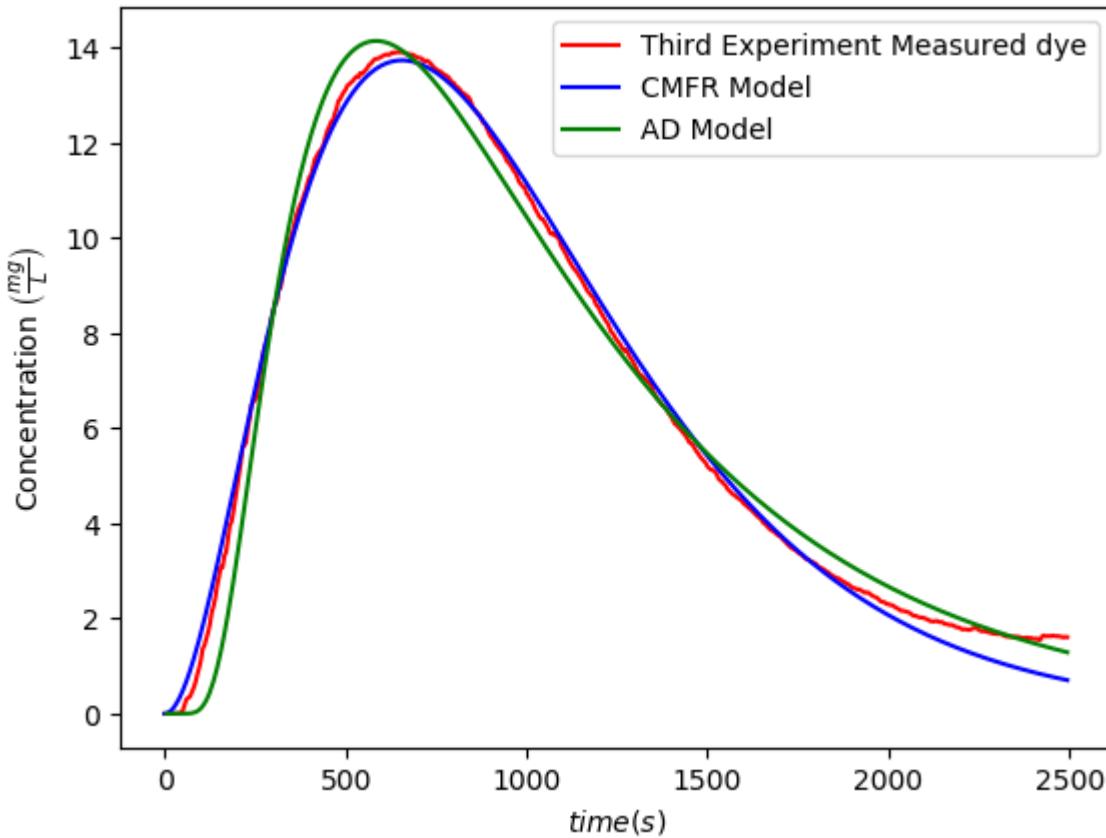


Figure 9. 8 cm Height Difference Data vs Models

We found the tracer residence time to be 740 seconds (12 and a half minutes). This equates to roughly 3.02 CMFRs in series, which was the most accurate of the experiments.

We found our t^* for $F = 0.1$ to be .51 by using a while loop to search through the data shown in Figure 10.

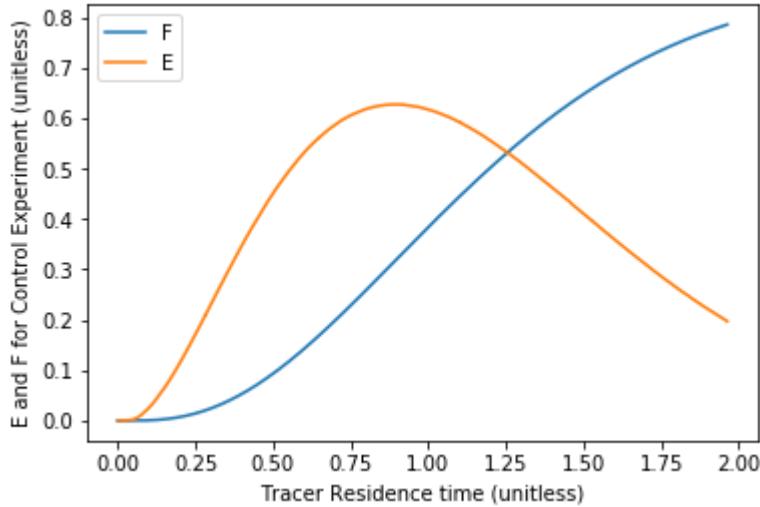


Figure 10. Control Experiment E & F

For our 27.5 cm height difference, we found our t^* for $F = 0.1$ to be .52, shown in Figure 11.

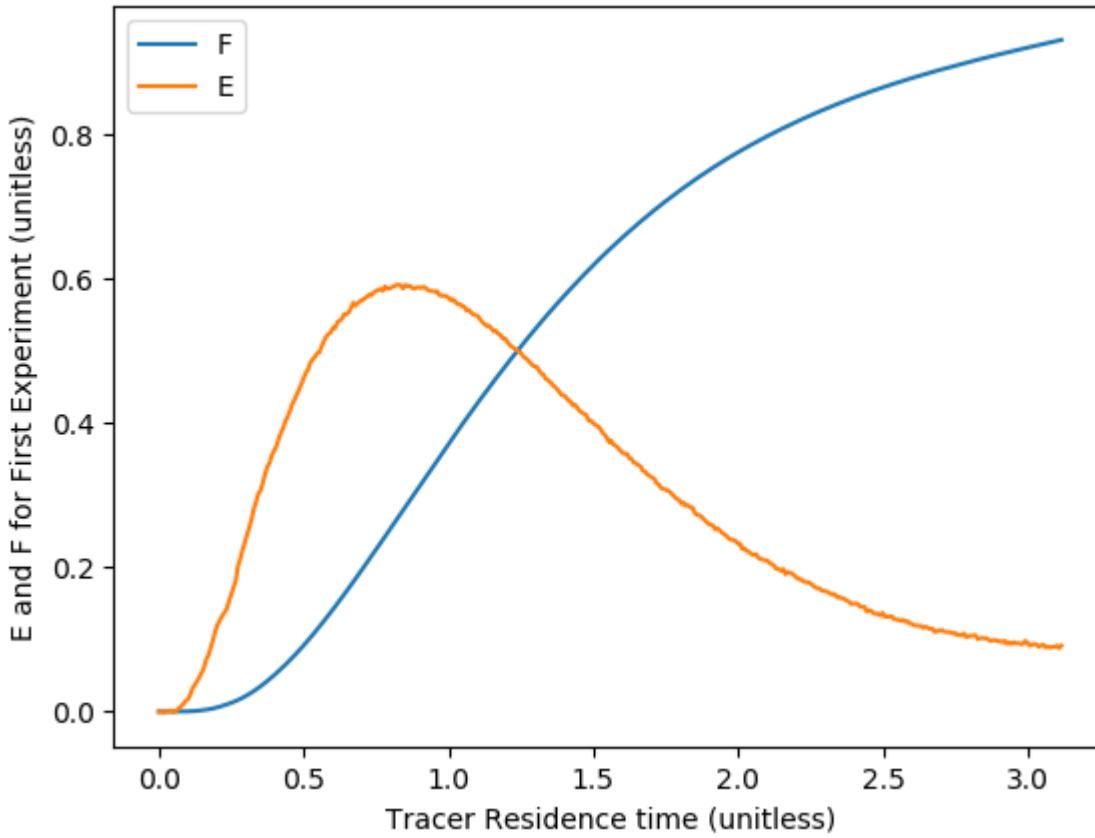


Figure 11. 27.5 cm Height Difference Experiment E & F

We found our t^* for F = 0.1 to be .49 for our 15 cm height difference.

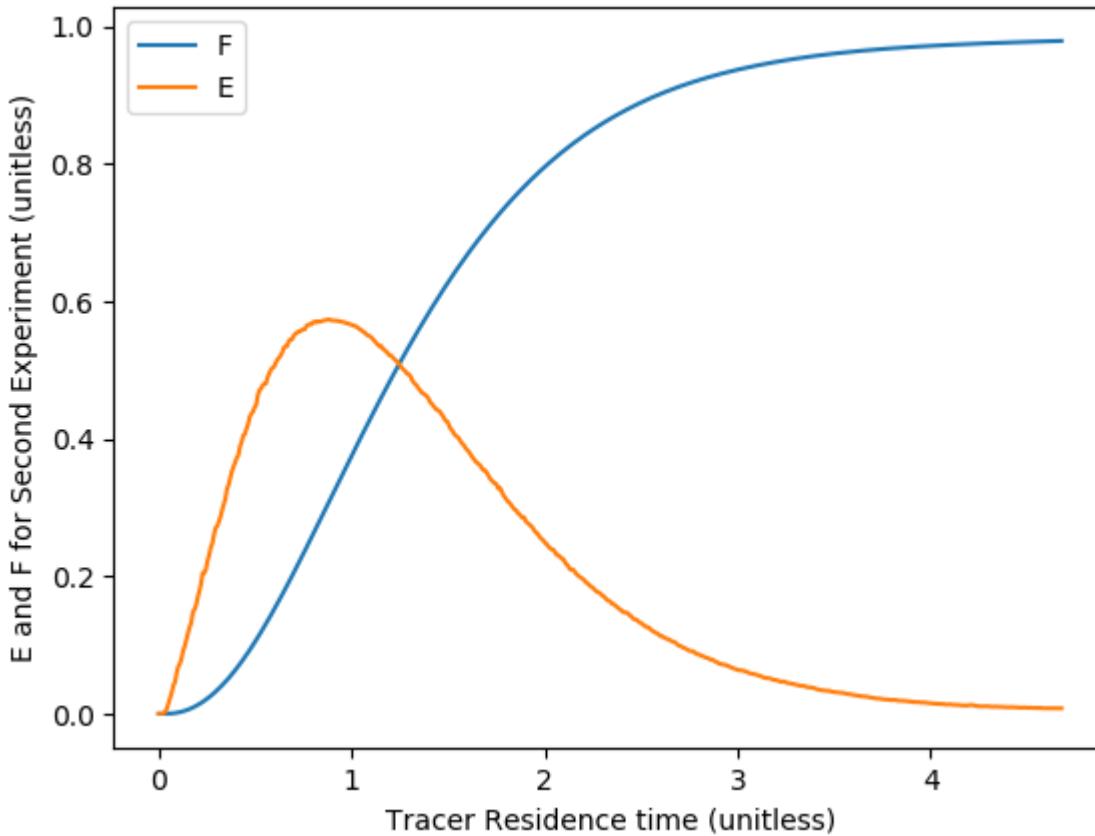


Figure 12. 15 cm Height Difference E & F

We found our t^* for $F = 0.1$ to be .51 for our last run with an 8 cm height difference.

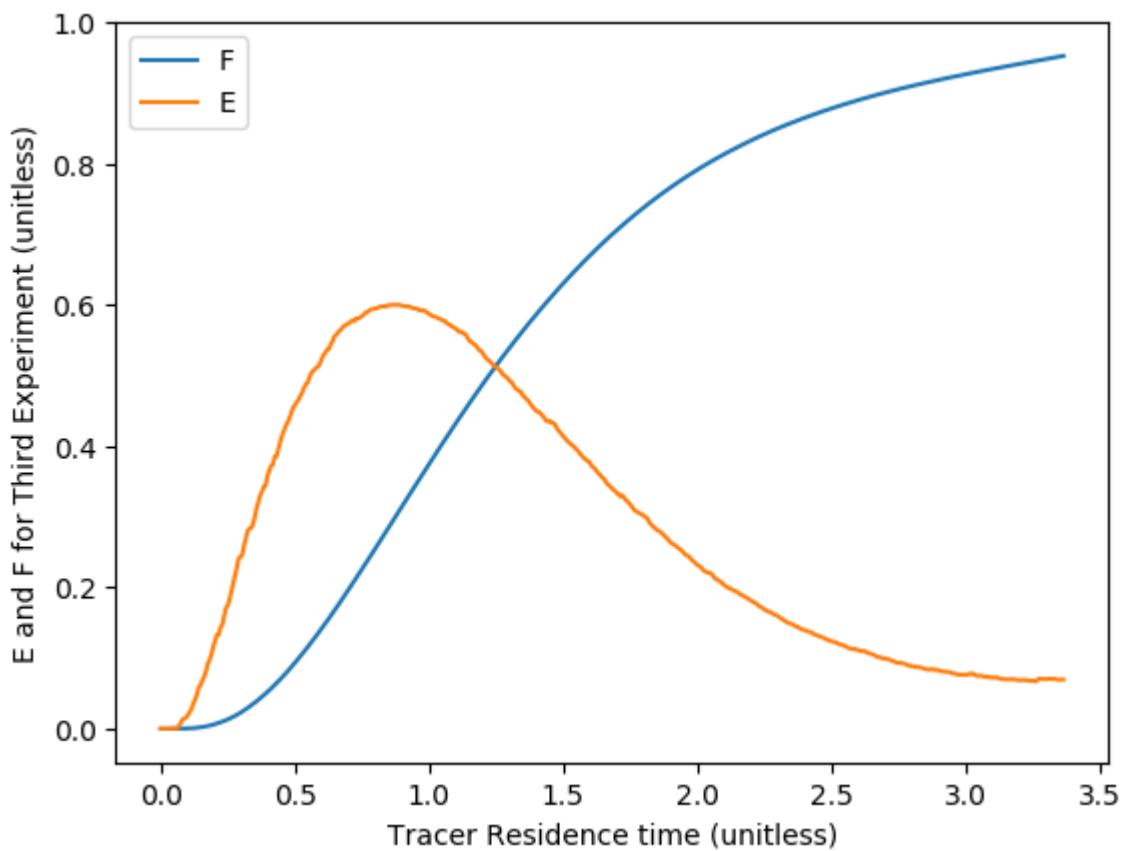


Figure 13. 8 cm Height Difference E & F

The t^* for $F = 0.1$ for all four of the experiments are always within a few hundredths of 0.5, which corresponds to an average baffling condition, meaning the reactors had a baffled inlet and outlet with some intrabasin baffles. A CMFR typically has a value of 0.1, so initially we were surprised, but our system was multiple CMFRs in series so the transition from one reactor to the next was likely interpreted as baffles. It is known that with enough CMFRs in series the flow begins to resemble plug flow, our experiment had 3 in series which was clearly not enough to approach plug flow but was better than a single CMFR.

From what we've learned in these experiments, it is possible to create CMFRs in series through gravity induced mixing. We can further conclude that they perform very closely to the CMFRs created through mechanical stirring with stir bars. We did notice some dead volumes in the corners of the tanks with the gravity induced mixing; however, it appears that they did not have a significant effect on the performance of those eddys.

Conclusion

Our trials with different elevation characteristics taught us how much elevation, and in turn energy, is required to fully mix reactors in order to approach the CMFR model. We experimented with different

heights to see if they would affect the mixing ability, eventually reaching the minimum required height difference of 8 cm.

Our CMFR control behaved exactly to what we had expected. The two highest height differences behaved well and performed close to the equivalent ideal CMFRs in series. Notably, the 15 cm height difference was almost exactly in line with the generated CMFR model. However, our last run with the 8 cm difference behaved the best, outperforming the other two height differences in terms of CMFRs in series.

Overall, we have learned that gravity induced mixing is effective as long as the conditions mentioned earlier are met, meaning that the height difference has to be large enough to account for head loss, eddy dissipation, and water pressure from fluid already present in the tank. With that being said, surprisingly, the energy required to mix fluid inside the reactor is very small compared to the energy required to overcome head loss. Therefore, we can conclude that as long as fluid is flowing from one reactor to the next, there is sufficient energy to mix the majority of the reactor quite well.

This has impressive implications, for example, wastewater treatment plants require heavy mixing at various stages in the process, but our conclusion would indicate that the flow of the fluid itself is enough to adequately mix the reactor without the need for mechanical mixing and the outside energy it requires. Further research could disprove this theory in large scale applications, but it is a route for future investigation.

Table 1. Summary of Values for the Different Runs

Experiment	Number of CMFRs in series	Residence Time (sec)	T star
Control	3.5	7400	.51
27.5 cm height difference	2.94	640	.52
15 cm height difference	2.81	590	.49
8 cm height difference	3.02	740	.51

Hurdles and challenges

We encountered several challenges during the lab. First of all, we had to decide the elevation difference between the reactors. We did the calculation but turned out our number was a lot smaller than the minimum elevation required in reality. This was due to the fact that the water level for the 8-cm-height-difference experiment was higher than the inlet when we set the height difference at minimum, and some other head losses were not accounted for.

We faced challenges when setting up the elevation difference. We had to use the top shelf to make sure we created enough elevation. We used books, boxes, small tanks, buckets and small plastic blocks to adjust the height. We had to run back and forth to refill water in the Jerry Can. We eventually connected the water tap and the Jerry Can with a long tube to ease this process.

We also saw some minor dead volume around the corner of the tank. Those regions took a longer time for mixing. To solve this, in the future we can try with a circular tank instead of a rectangle one. Since the inlet is smaller than the outlet, we had to use large pipes for outlet and small pipes for inlet. We thus needed to neck the pipe down at multiple joints. We noticed the pulsing flow came out from the pump and splash near the influent. To fix this, we used accumulator for steady flow and reduced the splash. We had to tape down tubes at multiple locations to lessen the shaking of tubes.

Comments and Suggestions

It was a great experience for us to design and conduct our own experiments. We were glad that most things worked out during the experiment. The data we gathered all made sense and there were no huge spikes in concentration that would have indicated improper photometer function. However, we had some problems, some of which we were able to fix and others that could be addressed in the future by additional research.

We observed some dead volumes near the corner of the tank. In the future we could try to use a circular tank instead of a rectangle one to avoid this problem. On the topic of tanks, we also think tanks that are better suited for this exact experiment would be a big boost. For example, the tanks we used had the influent and effluent ports on the same side, making it difficult to achieve a linear flow from tank to tank that minimized the required tubing. Also, the influent and effluent ports were different sizes, meaning the tubes had to contract/expand, further adding to the head loss of the system. The location of these ports also gave us trouble, the influent port was underneath the water level in the tank so backpressure added to required elevational difference.

Overall, these were minor inconveniences and could be addressed in future research that was better funded and more professional. We are incredibly satisfied in our outcome for this course and the knowledge and skills we have gained during this semester. We want to thank Professor Monroe Weber-Shirk for his advice and guidance, TA Jonathon for being patient with us all through the semester, and our peers who shared their space and time so that we could all learn in a low-stress and fun environment.

References

Weber-Shirk, Monroe, et al. The the Environmental Engineering Laboratory Research Textbook,
https://monroews.github.io/EnvEngLabTextbook/Reactor_Characteristics/Reactor_Characteristics.html

Weber-Shirk, Monroe, et al. The Aguacalara Textbook, aguaclara.github.io/Textbook/index.html.

Appendix

```

from aguaclara.core.units import unit_registry as u
import aguaclara.research.environmental_processes_analysis as epa
import aguaclara.core.utility as ut
import numpy as np
import scipy
import matplotlib.pyplot as plt

#Code to find the minimum height difference between reactors

#Finding the minimum head required for mixing
Q = 380*u.mL/u.min
V = 2*u.L
g = 9.81*u.m/u.s/u.s
L = .12*u.m
theta = V/Q
theta.to(u.min)
h = ((L**2)*1000)/((theta**2)*(2*g))
print('The minimum head required for mixing the reactor is',(h.to(u.cm)))

#Finding the major head loss of the system
e = .0015*u.millimeter
Dtubebig = 6*u.millimeter
Ltubebig = 12*u.centimeter
Dtubesmall = 4*u.millimeter
Lubesmall = 38*u.centimeter
vbig = Q/(3.1415*(Dtubebig/2)**2)
vsmall = Q/(3.1415*(Dtubesmall/2)**2)
kinvisc = 1.004*10**-6*u.m*u.m/u.sec
Rebig = Dtubebig*vbig/kinvisc
Resmall = Dtubesmall*vsmall/kinvisc
relroughbig = e/Dtubebig
relroughsmall = e/Dtubesmall
fDbig = .014
fDsmall = .016
majorhloss = (fDbig*Ltubebig/Dtubebig)*(vbig**2/(2*g))+(fDsmall*Lubesmall/Dtubesmall)*(vsmall**2/(2*g))
print('The major head loss of the system is',(majorhloss.to(u.cm)))

#finding the minor head loss of the system
ent = 0.5
ex = 1
contract = 0.26
bend = 0.3
Kbig = ent+contract+bend
Ksmall = ex
minorthloss = Kbig*(vbig**2/(2*g))+Ksmall*(vsmall**2/(2*g))
minorthloss.to(u.cm)
print('The minor head loss of the system is',(minorthloss.to(u.cm)))

backhead = 4.5*u.cm

#Finding total height required

```

```

totalhead=h+majorhloss+minorhloss+backhead
print('The total height required for the system is',(totalhead.to(u.cm)))

#Control Reactor Analysis
control_path = 'https://raw.githubusercontent.com/WesleySluga/CEE4530/master/ResearchControl.xls'
control_firstrow = epa.notes(control_path).last_valid_index() + 1
control_time_data = (epa.column_of_time(control_path,control_firstrow,-1)).to(u.s)
control_concentration_data = epa.column_of_data(control_path,control_firstrow,1,-1,'mg/L')

#the initial concentration measured by the photometer wasn't
#zero. Therefore we corrected that by subtracting the initial reading from the entire data set.

control_concentration_data = control_concentration_data - control_concentration_data[0]
control_V = 6.0*u.L
control_Q = 3.8 * u.mL/u.min
control_theta_hydraulic = (control_V/control_Q).to(u.s)
control_C_bar_guess = np.max(control_concentration_data)/2
#use solver to get the CMFR parameters
control_CMFR = epa.Solver_CMFR_N(control_time_data, control_concentration_data, control_theta_hydraulic)
control_CMFR.C_bar
control_CMFR.N
control_CMFR.theta.to(u.s)

control_CMFR_model = (control_CMFR.C_bar*epa.E_CMFR_N(control_time_data/control_CMFR.theta, control_CMFR.N))

#use solver to get the advection dispersion parameters
control_AD = epa.Solver_AD_Pe(control_time_data, control_concentration_data, control_theta_hydraulic)
control_AD.C_bar
control_AD.Pe
control_AD.theta

print('The model estimate of the number of reactors in series for the control was', control_CMFR.N)
print('The model estimated mass of tracer injected for the control was', ut.round_sf(control_AD.C_bar, 2))
print('The model estimate of the Peclet number for the control was', control_AD.Pe)
print('The tracer residence time for the control was', ut.round_sf(control_AD.theta , 2))
print('The ratio of tracer to hydraulic residence time for the control was',(control_AD.theta/control_theta_hydraulic))

control_tstar = control_time_data/control_AD.theta
E_control = control_concentration_data*control_V/(control_AD.C_bar*control_V)
F_control = []
for i in range(E_control.size):
    timedata = control_tstar[0:i]
    integration = np.trapz(E_control[0:i],timedata)
    F_control.append(integration)
x = F_control[0]
i = 0
while x < 0.1:
    i=i+1
    x = F_control[i]
tstar_control = control_tstar[i]
print('The t* for the Control Experiment at 0.1 F is',(tstar_control))

```

```

plt.plot(control_tstar,F_control)
plt.plot(control_tstar,E_control)
plt.xlabel('Tracer Residence time (unitless)')
plt.ylabel('E and F for Control Experiment (unitless)')
plt.legend(['F','E'])
plt.savefig('control_E&F.png', bbox_inches = 'tight')
plt.show()

#Create the advection dispersion model curve based on the solver parameters
control_AD_model = (control_AD.C_bar*epa.E_Advective_Dispersion((control_time_data/control_AD.thet

#Plot the data and the two model curves.
plt.plot(control_time_data.to(u.s), control_concentration_data.to(u.mg/u.L),'r')
plt.plot(control_time_data.to(u.s),control_CMFR_model,'b')
plt.plot(control_time_data.to(u.s), control_AD_model,'g')
plt.xlabel(r'$time (s)$')
plt.ylabel(r'Concentration $\left( \frac{mg}{L} \right)$')
plt.legend(['Control Measured dye', 'CMFR Model', 'AD Model'])
plt.savefig('controlmodels.png', bbox_inches = 'tight')
plt.show()

#First Experiment Design
first_experiment_path = 'https://raw.githubusercontent.com/WesleySluga/CEE4530/master/ResearchExp1
first_experiment_firstrow = epa.notes(first_experiment_path).last_valid_index() + 1
first_experiment_time_data = (epa.column_of_time(first_experiment_path,first_experiment_firstrow,-
first_experiment_concentration_data = epa.column_of_data(first_experiment_path,first_experiment_fi

count=0
while first_experiment_time_data[count].magnitude < 2000:
    count=count +1
first_experiment_time_data = first_experiment_time_data[:count]
first_experiment_concentration_data = first_experiment_concentration_data[:count]

#the initial concentration measured by the photometer wasn't
#zero. Therefore we corrected that by subtracting the initial reading from the entire data set.

first_experiment_concentration_data = first_experiment_concentration_data - first_experiment_conce
first_experiment_V = 6.0*u.L
first_experiment_Q = 380 * u.mL/u.min
first_experiment_theta_hydraulic = (first_experiment_V/first_experiment_Q).to(u.s)
first_experiment_C_bar_guess = np.max(first_experiment_concentration_data)/2
#use solver to get the CMFR parameters
first_experiment_CMFR = epa.Solver_CMFR_N(first_experiment_time_data, first_experiment_concentrat
first_experiment_CMFR.C_bar
first_experiment_CMFR.N
first_experiment_CMFR.theta.to(u.s)

first_experiment_CMFR_model = (first_experiment_CMFR.C_bar*epa.E_CMFR_N(first_experiment_time_dat

```

```

#use solver to get the advection dispersion parameters
first_experiment_AD = epa.Solver_AD_Pe(first_experiment_time_data, first_experiment_concentration_
first_experiment_AD.C_bar
first_experiment_AD.Pe
first_experiment_AD.theta

print('The model estimate of the number of reactors in series for the first experiment was', first
print('The model estimated the mass of tracer injected for the first experiment was',ut.round_sf(f
print('The model estimate of the Peclet number for the first experiment was', first_experiment_AD.
print('The tracer residence time for the first experiment was',ut.round_sf(first_experiment_AD.the
print('The ratio of tracer to hydraulic residence time for the first experiment was',(first_experi

first_experiment_tstar = first_experiment_time_data/first_experiment_AD.theta
E_first_experiment = first_experiment_concentration_data*first_experiment_V/(first_experiment_AD.C_
F_first_experiment = []
for i in range(E_first_experiment.size):
    timedata = first_experiment_tstar[0:i]
    integration = np.trapz(E_first_experiment[0:i],timedata)
    F_first_experiment.append(integration)
x = F_first_experiment[0]
i = 0
while x < 0.1:
    i=i+1
    x = F_first_experiment[i]
tstar_first_experiment = first_experiment_tstar[i]
print('The t* for the First Experiment at 0.1 F is',(tstar_first_experiment))

plt.plot(first_experiment_tstar,F_first_experiment)
plt.plot(first_experiment_tstar,E_first_experiment)
plt.xlabel('Tracer Residence time (unitless)')
plt.ylabel('E and F for First Experiment (unitless)')
plt.legend(['F','E'])
plt.savefig('first_experiment_E&F.png', bbox_inches = 'tight')
plt.show()

#Create the advection dispersion model curve based on the solver parameters
first_experiment_AD_model = (first_experiment_AD.C_bar*epa.E_Advective_Dispersion((first_experim

#Plot the data and the two model curves.
plt.plot(first_experiment_time_data.to(u.s), first_experiment_concentration_data.to(u.mg/u.L), 'r')
plt.plot(first_experiment_time_data.to(u.s),first_experiment_CMFR_model,'b')
plt.plot(first_experiment_time_data.to(u.s), first_experiment_AD_model,'g')
plt.xlabel(r'$time (s)$')
plt.ylabel(r'Concentration $\left( \frac{mg}{L} \right)$')
plt.legend(['First Experiment Measured dye','CMFR Model', 'AD Model'])
plt.savefig('FirstExpmodels.png', bbox_inches = 'tight')
plt.show()

#Second Experiment Design
second_experiment_path = 'https://raw.githubusercontent.com/WesleySluga/CEE4530/master/ResearchExp

```

```

second_experiment_firstrow = epa.notes(second_experiment_path).last_valid_index() + 1
second_experiment_time_data = (epa.column_of_time(second_experiment_path, second_experiment_firstrow))
second_experiment_concentration_data = epa.column_of_data(second_experiment_path, second_experiment_firstrow)

#the initial concentration measured by the photometer wasn't
#zero. Therefore we corrected that by subtracting the initial reading from the entire data set.

second_experiment_concentration_data = second_experiment_concentration_data - second_experiment_cc
second_experiment_V = 6.0*u.L
second_experiment_Q = 380 * u.mL/u.min
second_experiment_theta_hydraulic = (second_experiment_V/second_experiment_Q).to(u.s)
second_experiment_C_bar_guess = np.max(second_experiment_concentration_data)/2
#use solver to get the CMFR parameters
second_experiment_CMFR = epa.Solver_CMFR_N(second_experiment_time_data, second_experiment_concentration_data)
second_experiment_CMFR.C_bar
second_experiment_CMFR.N
second_experiment_CMFR.theta.to(u.s)

second_experiment_CMFR_model = (second_experiment_CMFR.C_bar*epa.E_CMFR_N(second_experiment_time_data))

#use solver to get the advection dispersion parameters
second_experiment_AD = epa.Solver_AD_Pe(second_experiment_time_data, second_experiment_concentration_data)
second_experiment_AD.C_bar
second_experiment_AD.Pe
second_experiment_AD.theta

print('The model estimate of the number of reactors in series for the Second Experiment was', second_experiment_CMFR.N)
print('The model estimated mass of tracer injected for the Second Experiment was', ut.round_sf(second_experiment_CMFR.Q))
print('The model estimate of the Peclet number for the Second Experiment was', second_experiment_AD.Pe)
print('The tracer residence time for the Second Experiment was', ut.round_sf(second_experiment_AD.t))
print('The ratio of tracer to hydraulic residence time for the Second Experiment was', (second_experiment_CMFR.N/second_experiment_AD.t))

second_experiment_tstar = second_experiment_time_data/second_experiment_AD.theta
E_second_experiment = second_experiment_concentration_data*second_experiment_V/(second_experiment_CMFR.Q)
F_second_experiment = []
for i in range(E_second_experiment.size):
    timedata = second_experiment_tstar[0:i]
    integration = np.trapz(E_second_experiment[0:i],timedata)
    F_second_experiment.append(integration)
x = F_second_experiment[0]
i = 0
while x < 0.1:
    i=i+1
    x = F_second_experiment[i]
tstar_second_experiment = second_experiment_tstar[i]
print('The t* for the Second Experiment at 0.1 F is',(tstar_second_experiment))

plt.plot(second_experiment_tstar,F_second_experiment)
plt.plot(second_experiment_tstar,E_second_experiment)
plt.xlabel('Tracer Residence time (unitless)')
plt.ylabel('E and F for Second Experiment (unitless)')

```

```

plt.legend(['F','E'])
plt.savefig('second_experiment_E&F.png', bbox_inches = 'tight')
plt.show()

#Create the advection dispersion model curve based on the solver parameters
second_experiment_AD_model = (second_experiment_AD.C_bar*epa.E_Advective_Dispersion((second_experi

#Plot the data and the two model curves.
plt.plot(second_experiment_time_data.to(u.s), second_experiment_concentration_data.to(u.mg/u.L), 'r')
plt.plot(second_experiment_time_data.to(u.s),second_experiment_CMFR_model,'b')
plt.plot(second_experiment_time_data.to(u.s), second_experiment_AD_model,'g')
plt.xlabel(r'$time (s)$')
plt.ylabel(r'Concentration $\left( \frac{mg}{L} \right)$')
plt.legend(['Second Experiment Measured dye','CMFR Model', 'AD Model'])
plt.savefig('SecondExpmodels.png', bbox_inches = 'tight')
plt.show()

#Third Experiment Design
third_experiment_path = 'https://raw.githubusercontent.com/WesleySluga/CEE4530/master/ResearchExp'
third_experiment_firstrow = epa.notes(third_experiment_path).last_valid_index() + 1
third_experiment_time_data = (epa.column_of_time(third_experiment_path,third_experiment_firstrow,-
third_experiment_concentration_data = epa.column_of_data(third_experiment_path,third_experiment_fi

count=0
while third_experiment_time_data[count].magnitude < 2500:
    count=count +1
third_experiment_time_data = third_experiment_time_data[:count]
third_experiment_concentration_data = third_experiment_concentration_data[:count]
#the initial concentration measured by the photometer wasn't
#zero. Therefore we corrected that by subtracting the initial reading from the entire data set.

third_experiment_concentration_data = third_experiment_concentration_data - third_experiment_conce
third_experiment_V = 7.0*u.L
third_experiment_Q = 380 * u.mL/u.min
third_experiment_theta_hydraulic = (third_experiment_V/third_experiment_Q).to(u.s)
third_experiment_C_bar_guess = np.max(third_experiment_concentration_data)/2
#use solver to get the CMFR parameters
third_experiment_CMFR = epa.Solver_CMFR_N(third_experiment_time_data, third_experiment_concentrat
third_experiment_CMFR.C_bar
third_experiment_CMFR.N
third_experiment_CMFR.theta.to(u.s)

third_experiment_CMFR_model = (third_experiment_CMFR.C_bar*epa.E_CMFR_N(third_experiment_time_data

#use solver to get the advection dispersion parameters
third_experiment_AD = epa.Solver_AD_Pe(third_experiment_time_data, third_experiment_concentration_
third_experiment_AD.C_bar
third_experiment_AD.Pe
third_experiment_AD.theta

print('The model estimate of the number of reactors in series for the Third Experiment was', thir

```

```

print('The model estimated mass of tracer injected for the Third Experiment was',ut.round_sf(third_experiment_C_bar))
print('The model estimate of the Peclet number for the Third Experiment was', third_experiment_AD.Peclet)
print('The tracer residence time for the Third Experiment was',ut.round_sf(third_experiment_AD.theta))
print('The ratio of tracer to hydraulic residence time for the Third Experiment was',(third_experiment_AD.F))

third_experiment_tstar = third_experiment_time_data/third_experiment_AD.theta
E_third_experiment = third_experiment_concentration_data*third_experiment_V/(third_experiment_AD.C_bar)
F_third_experiment = []
for i in range(E_third_experiment.size):
    timedata = third_experiment_tstar[0:i]
    integration = np.trapz(E_third_experiment[0:i],timedata)
    F_third_experiment.append(integration)
x = F_third_experiment[0]
i = 0
while x < 0.1:
    i=i+1
    x = F_third_experiment[i]
tstar_third_experiment = third_experiment_tstar[i]
print('The t* for the Third Experiment at 0.1 F is',(tstar_third_experiment))

plt.plot(third_experiment_tstar,F_third_experiment)
plt.plot(third_experiment_tstar,E_third_experiment)
plt.xlabel('Tracer Residence time (unitless)')
plt.ylabel('E and F for Third Experiment (unitless)')
plt.legend(['F','E'])
plt.savefig('third_experiment_E&F.png', bbox_inches = 'tight')
plt.show()

#Create the advection dispersion model curve based on the solver parameters
third_experiment_AD_model = (third_experiment_AD.C_bar*epa.E_Advective_Dispersion((third_experiment_time_data.to(u.s), third_experiment_concentration_data.to(u.mg/u.L), 'r')))

#Plot the data and the two model curves.
plt.plot(third_experiment_time_data.to(u.s), third_experiment_concentration_data.to(u.mg/u.L), 'r')
plt.plot(third_experiment_time_data.to(u.s),third_experiment_CMFR_model,'b')
plt.plot(third_experiment_time_data.to(u.s), third_experiment_AD_model,'g')
plt.xlabel(r'$time (s)$')
plt.ylabel(r'Concentration $\left( \frac{mg}{L} \right)$')
plt.legend(['Third Experiment Measured dye','CMFR Model', 'AD Model'])
plt.savefig('ThirdExpmodels.png', bbox_inches = 'tight')
plt.show()

```

