

# Final Report for Salt Treated Charcoal

Daniel Pyrek (6 hrs) and Jessie Powell (4 hrs) Group 8 5/13/19

## Introduction and Objectives

The purpose of this experiment was to activate charcoal with sodium chloride so that it might be used as a viable treatment for water and contaminant adsorption. A paper was found containing findings that coconut shells, made into charcoal and soaked in NaCl, appeared to have a positive effect on absorbency. However, the results were confusing because there was no comparison to either a substance with little adsorbance abilities, such as sand, or one with high adsorbency, such as activated carbon. Should salt treated charcoal actually have water filtering potential, this experiment would have significant social relevance. Developing countries with no access to a centralized water treatment plant would be able to filter their water with little resources and simple steps. Agricultural byproducts, such as old husks, shells, and other debris, could gain use by becoming the source of charcoal.

## Procedure

Four parameters were tested: sand, activated carbon, salt treated charcoal, and untreated charcoal. 30 grams of each substance were measured out and added to the column. For both charcoal tests, a coffee filter was added to each end of the column due to its fine, powdery nature. First, to remove air bubbles from the column, water was pumped at a flow rate of 10 rpm from the bottom of the tube up, flushing out any air. Once done, red dye #40, at a concentration of 0.05 g/L, was pumped down the column. The column was observed until the breakthrough time was achieved.

To begin with the treatment of the charcoal, grilling briquettes and table salt (99% NaCl) were purchased. The briquettes were ground into a semi fine powder and soaked in the salt solution. This solution was a 25% mass per volume solution. 50 mg was added to a beaker and filled with water until reaching 200 mL. The charcoal was soaked for 24 hours, then allowed to dry. However, there was a significant amount of salt still in the charcoal. In order to remove it, the sample was put into the column and had water pumped into it, dissolving most of the salt. The remaining charcoal was then redried.

## Results and Discussion

Using ProCoDA and a photometer, concentration was recorded as red dye #40 was added to the column. Figure 1 is a graph of all the data superimposed on one graph. The activated charcoal graph's scale is so large that it makes the other graphs hard to read.

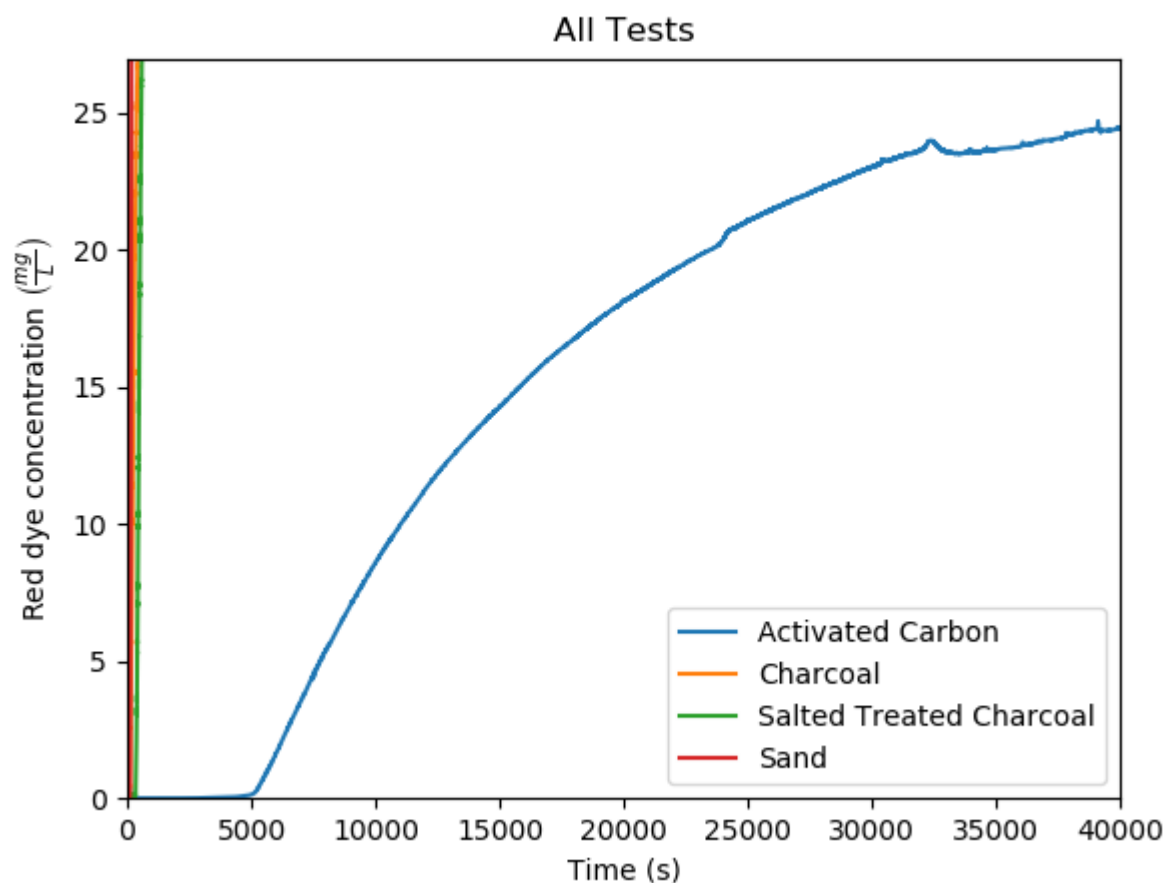


Figure 1: Breakthrough Graphs for Each Method

Figure 2 shows data purely for charcoal. Data was taken until half of the influent concentration. This concentration is marked with a red line.

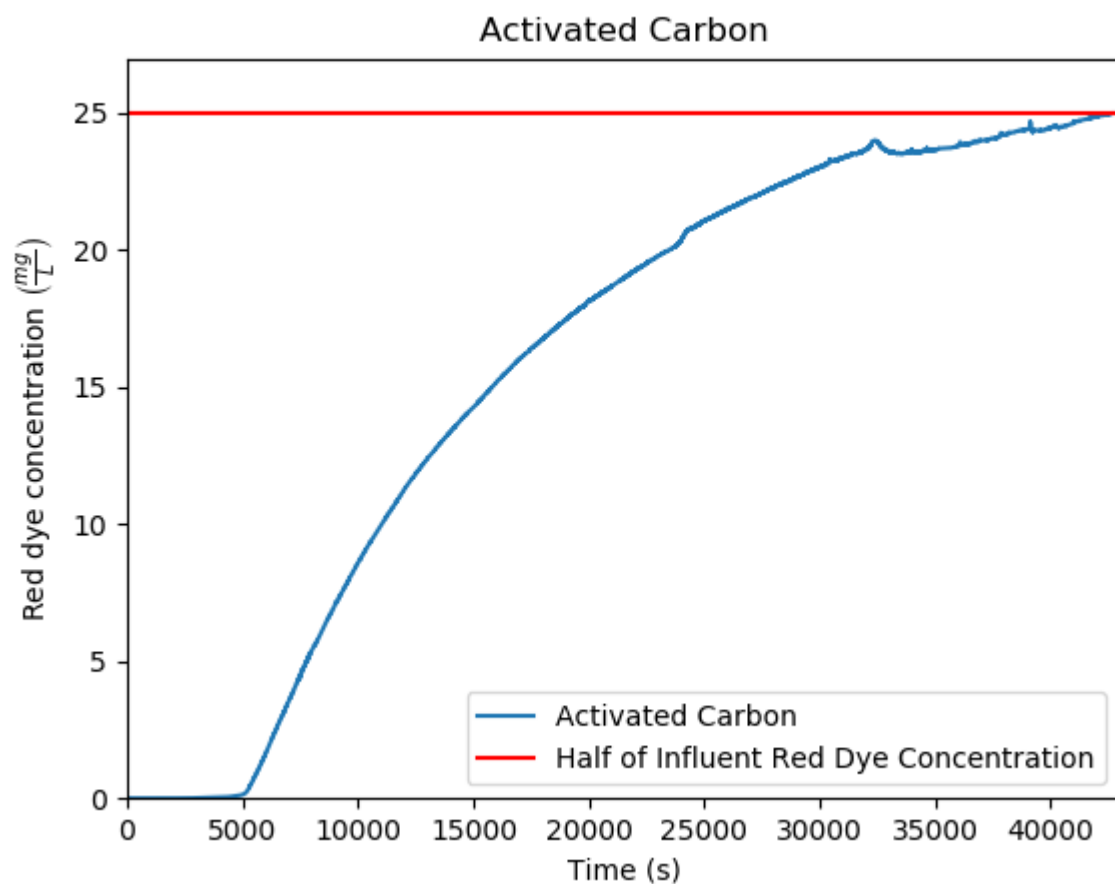


Figure 2: Breakthrough Curve for Activated Carbon

Figure 3 depicts the breakthrough graphs for charcoal, salt treated charcoal, and sand. Half of the influent concentration is graphed for comparison as well.

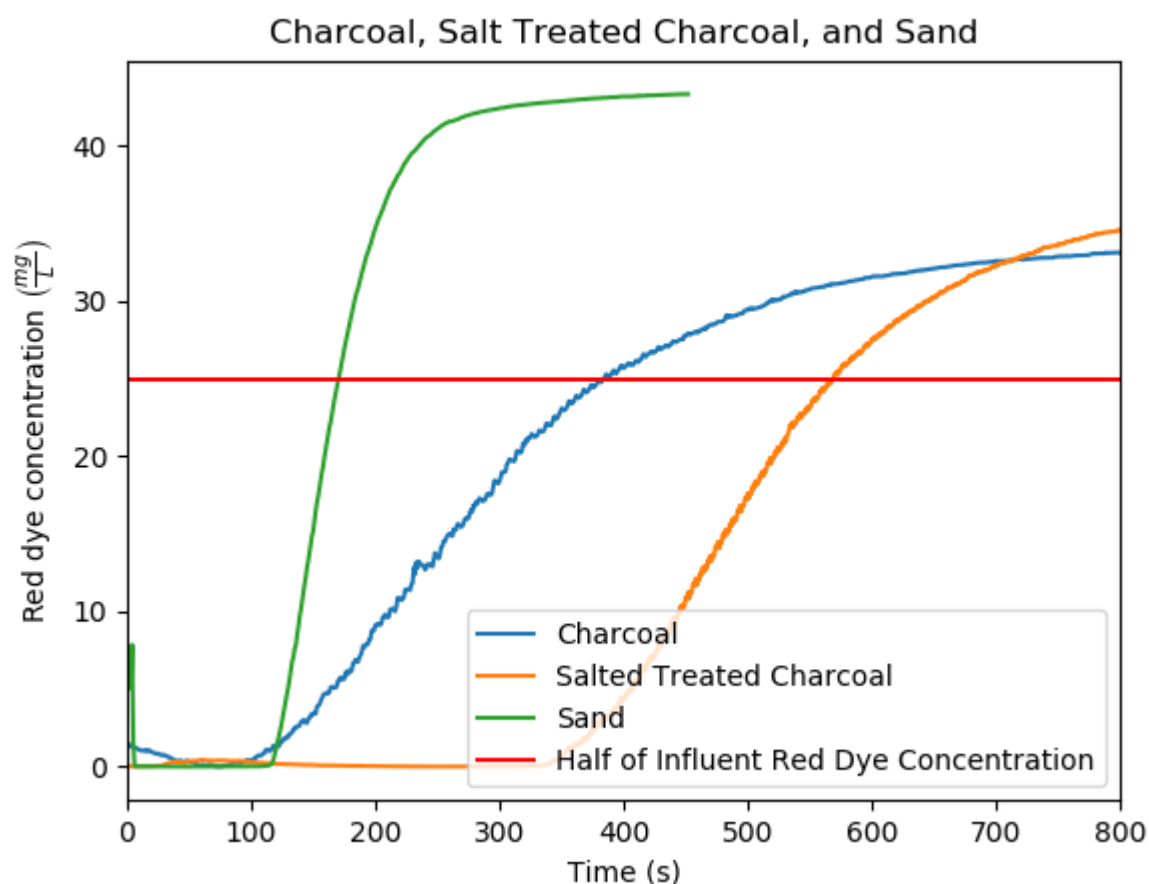


Figure 3: Breakthrough Graphs for Charcoal, Salt Treated Charcoal, and Sand

Visually, it can be seen that activated carbon is the best method by far. In order to confirm this, the breakthrough values were calculated for each method. Red dye 'breaks through' once the majority of the adsorbent reaches equilibrium with the influent concentration of red dye. An algorithm searched each data set for the first value above 1 mg/L. This allowed for some background movement. These breakthrough values are displayed in figure 4.

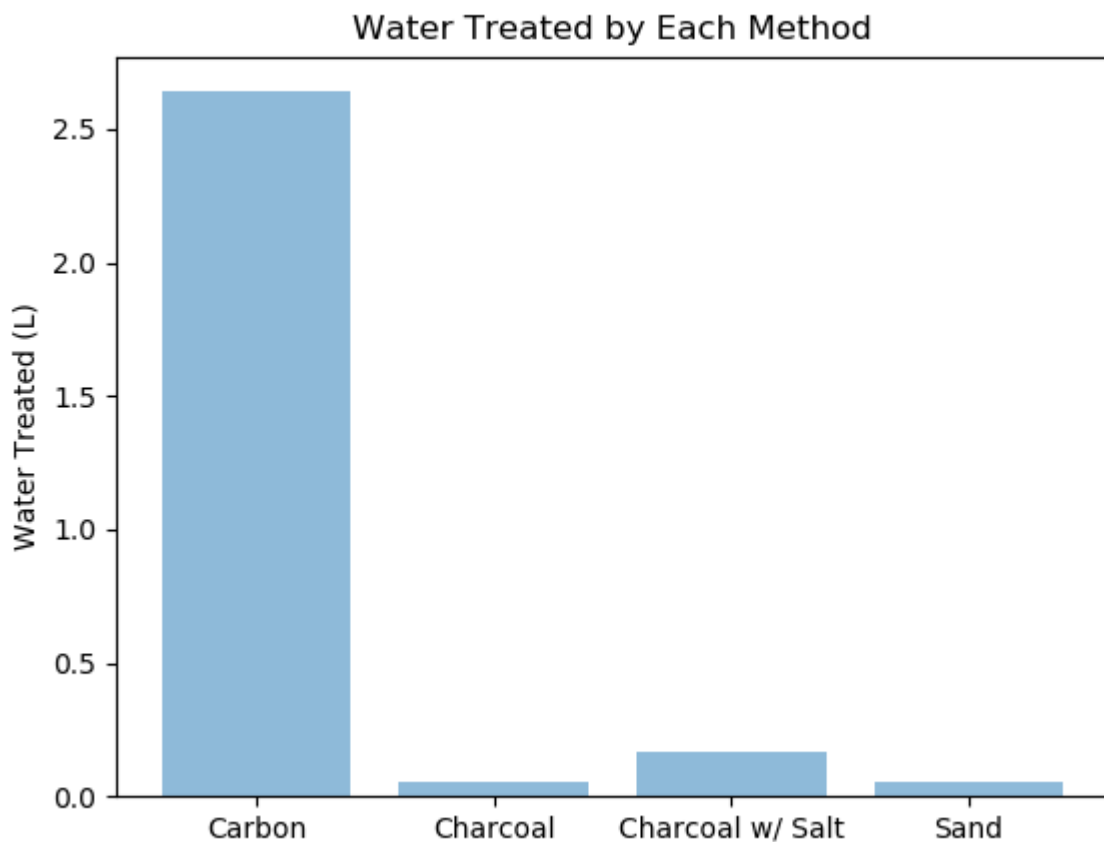
	Activated Carbon	Charcoal	Salt Treated Charcoal	Sand
Breakthrough Time (s)	5648	112	354	112

Figure 4: Breakthrough Values

Ten revolutions per minute was converted into liters per second. This was multiplied by the breakthrough times. This gave the number of liters treated by each method until the red dye made it through the filter. This data is displayed in figures 5 and 6:

	Activated Carbon	Charcoal	Salt Treated Charcoal	Sand
Treated Water (Liters)	2.64	0.05	0.17	0.06

Figure 5: Water Treated by Each Method



*Figure 6: Bar Graph of Water Treated by Each Method*

Charcoal and sand treated the least water at around 0.05 liters. Charcoal treated with salt treated slightly more at 0.17 liters. Industrial grade activated treated the most at 2.64 liters.

## Conclusions

Salt treated charcoal showed some improvement over regular charcoal and sand. However, in order to remove salt from the activated charcoal after salt treatment, the charcoal was rinsed with 0.4 liters of water in the column. Therefore, this treatment method resulted in a net negative amount of water treated. Salt treated charcoal is not an effective method to remove red dye #40. Industrial grade activated carbon is by far the most reliable treatment method of the group. Thirty grams of activated carbon treated 2.64 liters of water. Charcoal and sand were only able to treat 0.05 liters of water. This project is important because it debunks articles online claiming that salt treated charcoal is effective.

## Suggestions/Comments

Should this experiment be replicated in the future, there are several suggestions and comments that would make these attempts easier and potentially yield more sound results. In terms of advice to make the current procedure easier, a coffee filter should be added to each side of the column in order to prevent any charcoal removal. A mass per volume solution for salt is recommended out of convenience of measuring out water and salt ratios. There are also several alterations to what is being tested or how the materials are prepared that could produce interesting results. Because activated carbon is activated by cooking at extreme heats, one idea would be to bake or cook the charcoal to see if that increases its absorption abilities. Additionally, it is recommended that tests be repeated for better accuracy in results.

## **Appendix:**

```

from aguaclara.core.units import unit_registry as u
import aguaclara.research.environmental_processes_analysis as epa
import aguaclara.core.physchem as pc
import aguaclara.core.utility as ut
import numpy as np
import matplotlib.pyplot as plt
import collections
import os
from pathlib import Path
import pandas as pd

```

```

def adsorption_data(C_column, dirpath):
    """This function extracts the data from folder containing tab delimited
    files of adsorption data. The file must be the original tab delimited file.

```

Parameters

-----

C\_column : int  
 index of the column that contains the dissolved oxygen concentration data.

dirpath : string  
 path to the directory containing aeration data you want to analyze

Returns

-----

filepaths : string list  
 all file paths in the directory sorted by flow rate  
 time\_data : numpy array list  
 sorted list of numpy arrays containing the times with units of seconds

Examples

-----

"""

```

#return the list of files in the directory
metadata = pd.read_csv(dirpath + '/metadata.txt', delimiter='\t')
filenames = metadata['file name']
#extract the flowrates from the filenames and apply units
#sort airflows and filenames so that they are in ascending order of flow rates

```

```

filepaths = [dirpath + '/' + i for i in filenames]
#C_data is a list of numpy arrays. Thus each of the numpy data arrays can have different length
# cycle through all of the files and extract the column of data with oxygen concentrations and
C_data=[epa.column_of_data(i, 2,-1) for i in filepaths]
time_data=[(epa.column_of_time(i,2,-1)).to(u.s) for i in filepaths]

adsorption_collection = collections.namedtuple('adsorption_results','metadata filenames C_data time_data')
adsorption_results = adsorption_collection(metadata, filenames, C_data, time_data)
return adsorption_results

```

```
C_column = 1
dirpath = "https://raw.githubusercontent.com/DanielPyrek/4530/master/data/Final/"
```

```
metadata, filenames, C_data, time_data = adsorption_data(C_column,dirpath)
metadata
```

```
#zero the concentration data by subtracting the value of the first data point from all data points
```

```
for k in range(np.size(filenames)):
    test = 1000
    for z in range(np.size(time_data[k])):
        if (C_data[k][z] < test and C_data[k][z] > 0):
            test = C_data[k][z]
    for y in range(np.size(time_data[k])):
        C_data[k][y] = C_data[k][y] - test
```

```
#PLOT ALL ON SAME PLOT
```

```
i =0
plt.plot(time_data[i], C_data[i], '- ',label = "Activated Carbon");
i =1
plt.plot(time_data[i], C_data[i], '- ',label = "Charcoal");
i =2
plt.plot(time_data[i], C_data[i], '- ', label = "Salted Treated Charcoal");
i =3
plt.plot(time_data[i], C_data[i], '- ', label = "Sand");
plt.xlim([0,40000])
plt.ylim([0,27])
```

```
plt.xlabel(r'Time (s)');
plt.ylabel(r'Red dye concentration  $\left( \frac{\text{mg}}{\text{L}} \right)$ ');
plt.legend()
plt.title('All Tests')
plt.savefig('C:/Users/dapmo/github/4530/data/Final/FINALAll')
plt.show()
```

```
#PLOT CARBON
```

```
i =0
plt.plot(time_data[i], C_data[i], '- ',label = "Activated Carbon");
plt.xlim([0,43000])
plt.ylim([0,27])
```



```

plt.axhline(y=25, color='r', linestyle='--', label = "Half of Influent Red Dye Concentration")

plt.xlabel(r'Time (s)');
plt.ylabel(r'Red dye concentration $\left( \frac{\text{mg}}{\text{L}} \right)$');
plt.legend()
plt.title('Activated Carbon')
plt.savefig('C:/Users/dapmo/github/4530/data/Final/FINAL1')
plt.show()

#PLOT ALL OTHERS

i =1
plt.plot(time_data[i], C_data[i], '--', label = "Charcoal");
i =2
plt.plot(time_data[i], C_data[i], '--', label = "Salted Treated Charcoal");
i =3
plt.plot(time_data[i], C_data[i], '--', label = "Sand");

plt.axhline(y=25, color='r', linestyle='--', label = "Half of Influent Red Dye Concentration")

plt.title('Charcoal, Salt Treated Charcoal, and Sand')
plt.xlabel('Time (s)');
plt.xlim(0,800);
plt.ylabel(r'Red dye concentration $\left( \frac{\text{mg}}{\text{L}} \right)$');
plt.legend();
plt.savefig('C:/Users/dapmo/github/4530/data/Final/FINAL3')
plt.show()

#FIND BREAKTHROUGH POINTS
breakthrough = ();
for k in range(np.size(filenamees)):
    for z in range(np.size(time_data[k])):
        if (C_data[k][z] > 1 and time_data[k][z] > 50*u.second):
            breakthrough = np.append(breakthrough,time_data[k][z]);
            break

print(breakthrough)
breakthrough = np.asarray(breakthrough)

#Amount of water treated before breakthroughs
mLperRev = (2.8*(u.mL)/(u.revolution)).to((u.liter)/(u.revolution))
Q = (10*u.revolution/u.minute).to(u.revolution/u.second)
watertreated = [Q*mLperRev*breakthrough[0]*u.second,Q*mLperRev*breakthrough[1]*u.second,Q*mLperRev
print(watertreated)

water = [2.64,0.052,0.165,0.056]
objects = ('Carbon', 'Charcoal', 'Charcoal w/ Salt', 'Sand')
y_pos = np.arange(len(watertreated))

```

```
plt.bar(y_pos, water, align='center', alpha=0.5)
plt.xticks(y_pos, objects)
plt.ylabel('Water Treated (L)')
plt.title('Water Treated by Each Method')
plt.savefig('C:/Users/dapmo/github/4530/data/Final/FINALBAR')
plt.show()
```

...

