ECI 146: Routing of a Reservoir

Name: Ming Martin Liu

Date: 1/26/2021

1.Task

Brief: This document describes the procedure used to route a given reservoir inflow with the use of an orifice. The reservoir is assumed to have a square shaped cross sectional area.

The client has provided a flowrate in the reservoir given by the following equation.

$$Q_i = \frac{750}{\pi}\left[1 - cos\left(\frac{\pi t}{4500}\right)\right],\ 0 \le t \le 9000 \qquad (1)$$

The hydrograph of this flowrate can be plotted as shown below.

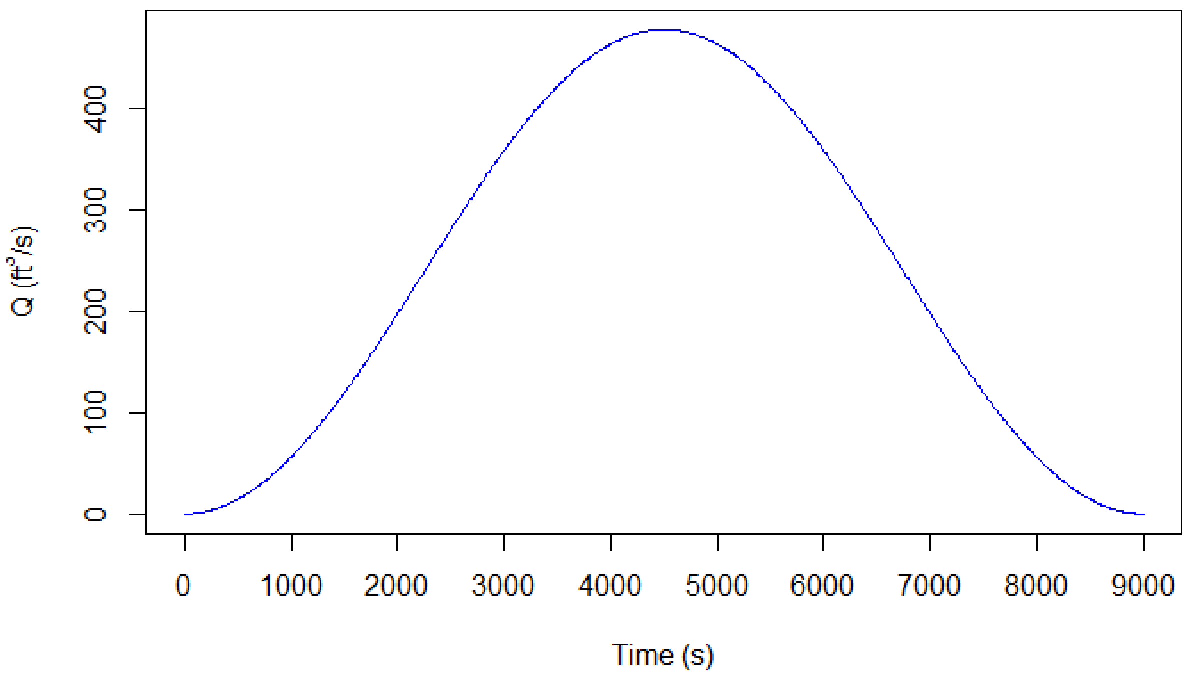

Figure 1: Graph of Flowrate vs Time

The maximum allowable height that the reservoir is allowed to reach is 6 feet. The client has also indicated that the maximum storage volume is 2,148,590 ft^3. Hence, the client has required the use of an orifice to route the hydrograph. The subsequent sections depict the analysis of flood routing with the use of different sized orifices.

2. Governing equations

$$V = \sum_{i=1}^{N-1} \frac{Q_i + Q_{i+1}}{2} \Delta T \qquad (2)$$

$$Q_o = C_D \mathrm{a}\sqrt{2gh}\,, \qquad (3)$$

Given:

$$C_D = 0.65,\ \mathrm{g} = 32.2\ \mathrm{ft/s}^2\ ,\ \mathrm{a} = 360000\ \mathrm{ft}^2$$

3. Solution procedure

All calculations were done using RStudio.
The calculations for volume were done with the trapezoidal method as shown in equation 2. This was mainly used to check if our timestep an accurate enough estimate for our subsequent calculations.

The calculations for storage were done with an iterative process commonly known as the Euler Method with a range of up to 9000 seconds, or when the storage of the reservoir is empty.

After obtaining Q_i from the client, the storage of filling the reservoir can be calculated using the equation below.

$$S^{k+1} = S^k + \left[\left(\frac{Q_i^k + Q_i^{k+1}}{2}\right)\right] \Delta t \qquad (4)$$

In this case, the storage is initially empty.
Following that, when the reservoir was at maximum capacity, the storage of emptying the reservoir can be calculated using the following equation.

$$S^{k+1} = S^k - \left[C_D a\sqrt{2g}\,(h^k)^{1/2}\right] \Delta t \qquad (5)$$

In this case, the storage is initially at maximum capacity.
These equations are further extended to represent a case where the reservoir is being filled as well as emptied at the same time. This can be done with the following equation.

$$S^{k+1} = S^k + \left[\left(\frac{Q_i^k + Q_i^{k+1}}{2}\right) - C_D a\sqrt{2g}\,(h^k)^{1/2}\right] \Delta t \qquad (6)$$

As can be seen, the storage is represented by both the inflow and the outflow.
With the storage available, the height of the reservoir can be easily calculated using the equation below.

$$h^{k+1} = \frac{S^{k+1}}{Area} \qquad (7)$$

Where the Area is said to be 360,000 ft^2.

4. Results

This section depicts the results from three scenarios, reservoir filling, reservoir emptying and a simultaneous case of filling and emptying. This section will also prove the accuracy of the calculations done with respect to the number of iterations being used.

4.1 Reservoir filling

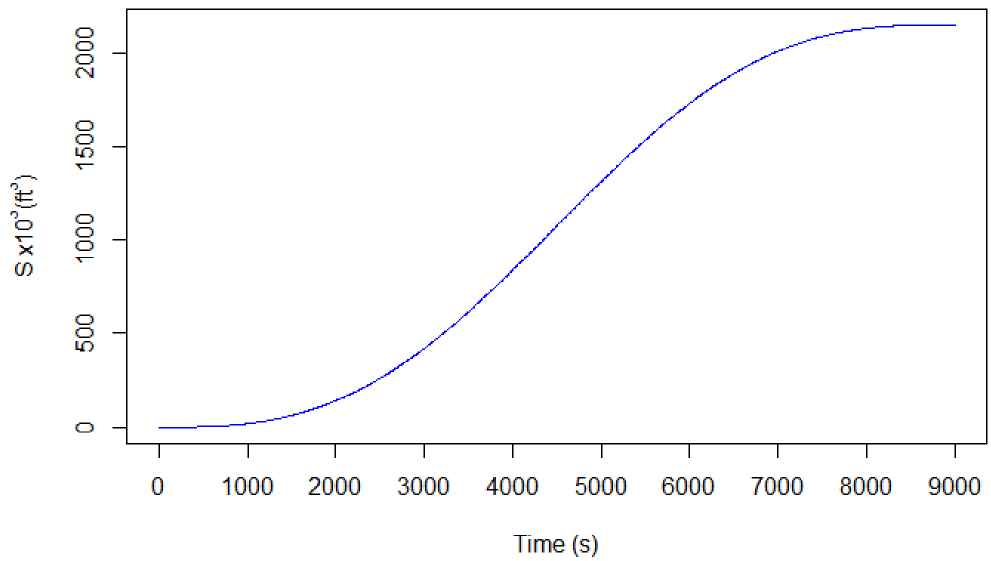

Figure 2: Graph of Filling Reservoir, Storage vs Time.

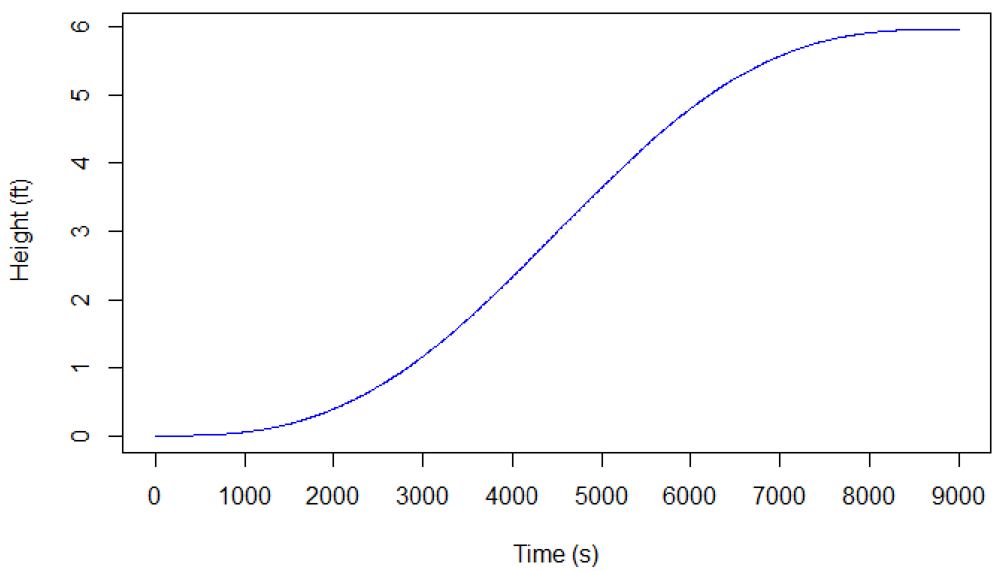

Figure 3: Graph of Filling Reservoir, Height vs Time.

As can be seen by Figure 2 and 3, the inflow due to Q_i will cause the reservoir to reach its maximum storage and maximum height. This shows that some form of routing should be implemented in order to ensure that the maximum height is lower than 6 feet.

4.2 Reservoir emptying

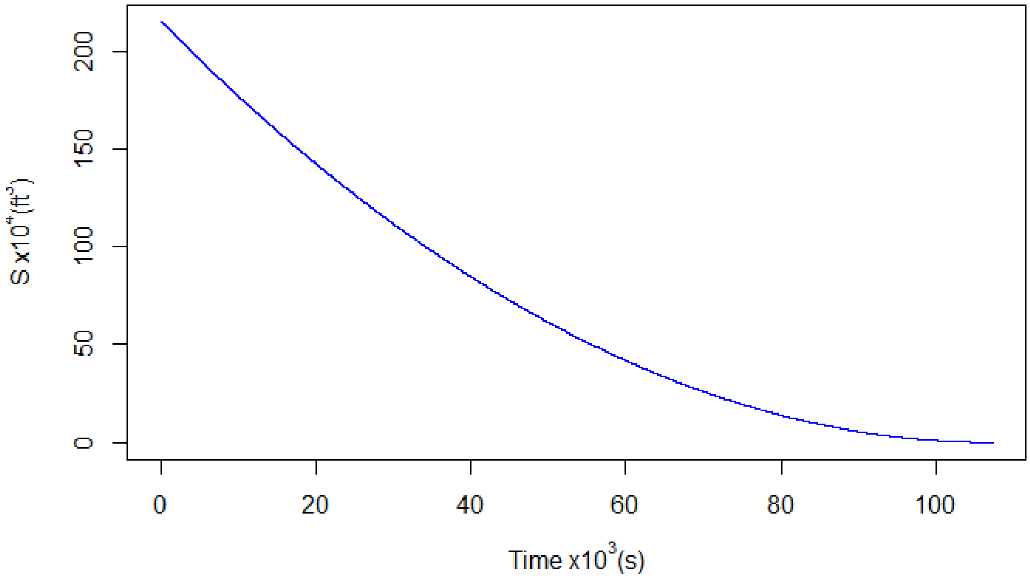

Figure 4: Graph of Emptying Reservoir, Storage vs Time.

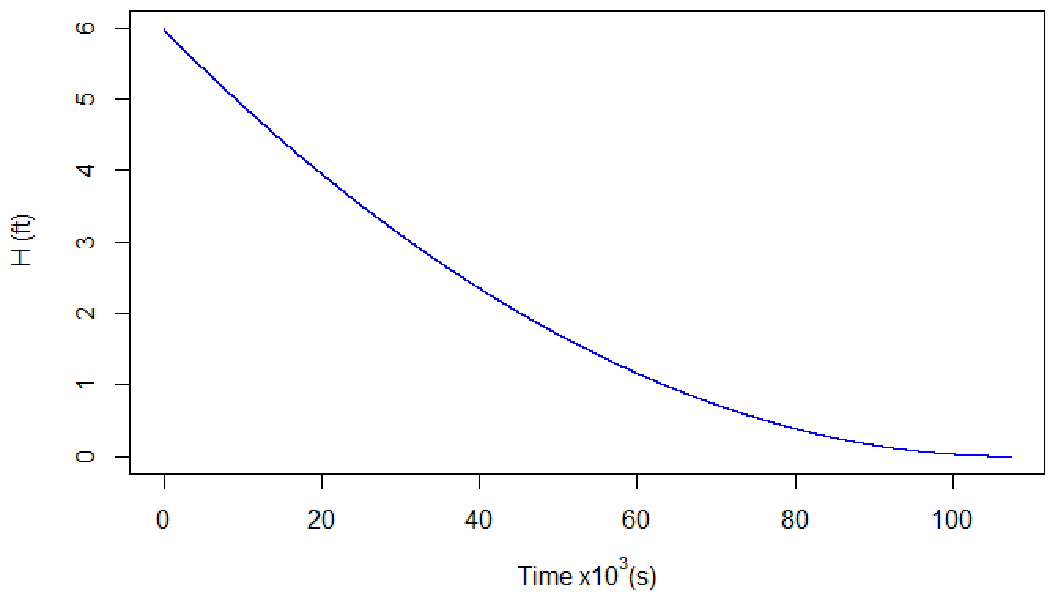

Figure 5: Graph of Emptying Reservoir, Height vs Time.

Figures 4 and 5 show how long it takes for the reservoir to empty out. The values from these plots are given based off equations 5 and 7. As can be seen, it takes a significantly longer amount of time for the water to be emptied out versus being filled.

4.3 Reservoir operations (simultaneous emptying and filling of reservoir)

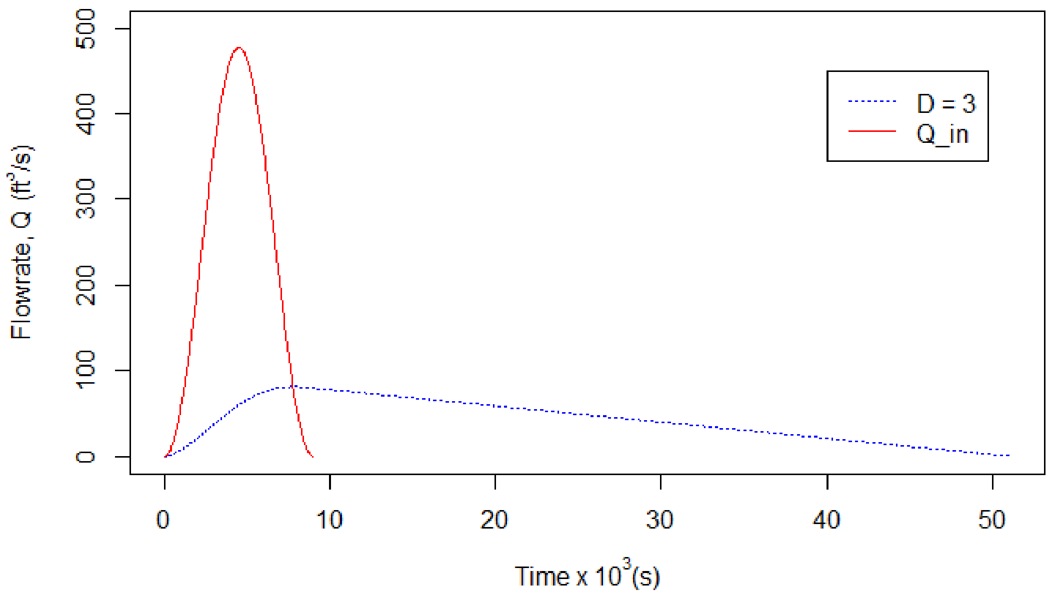

Figure 6: Graph of Flowrate vs Time. The red line shows the flowrate of the incoming wave. The blue dotted line shows the flowrate of the reservoir filling and emptying with an orifice diameter of 3 feet.

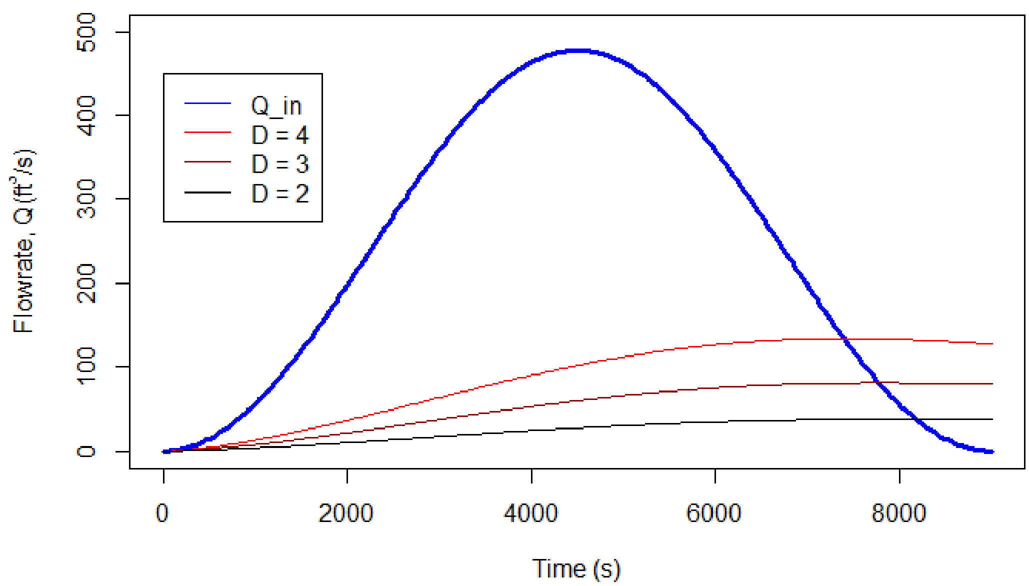

Figure 7: Graph of Flowrate vs Time. The thick blue line represents the initial flowrate. The red, dark-red and black line represents the net flowrate due to an orifice diameter of 4m, 3m, and 2m respectively.

As can be seen from Figure 6 and 7 the time for the reservoir to empty will take a significantly longer time later than the initial flowrate.

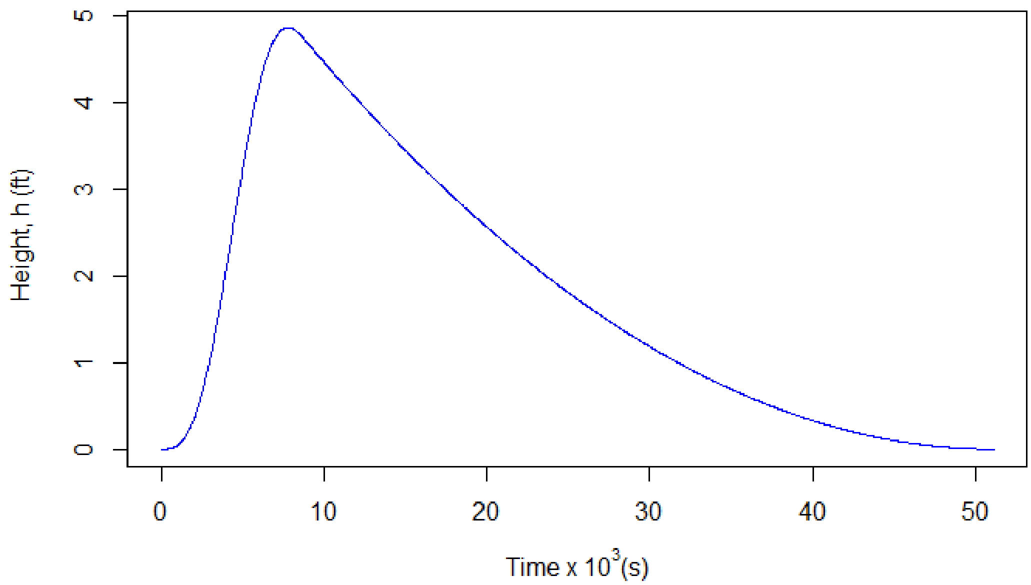

Figure 8: Graph of simultaneous emptying and filling of reservoir, Height vs Time. Diameter of orifice used is 3 feet.

As can be seen in Figure 8, with the use of an orifice to empty out the reservoir, the maximum height can be below 6 feet.

4.4 Accuracy checks

First, the number of iterations is checked. This is done by taking different numbers of iterations with the use of the trapezoidal rule when finding the volume.

Table 1: Comparison of Number of Iterations to the Volume

Timestep	Number of Iterations	Volume (ft^3)
1	9000	$2.148 * 10^6$
2	4500	$2.148 * 10^6$
3	3000	$2.148 * 10^6$
4	2250	$2.148 * 10^6$
5	1800	$2.148 * 10^6$
6	1500	$2.148 * 10^6$
7	1285	$2.148 * 10^6$
8	1125	$2.148 * 10^6$
9	1000	$2.148 * 10^6$
10	900	$2.148 * 10^6$

As can be seen by Table 1, the number of iterations is good even at 900 iterations. Hence, the number of iterations used can be lower as the level of accuracy still remains the same and the runtime will be lower.

The next factor of accuracy to check for is the maximum storage and height. The values for storage and height are calculated using equations 4 and 7, when the reservoir is being filled to maximum capacity.

Table 2: Comparison of Number of Iterations to the Storage and Height at a filled reservoir

Timestep	Number of Iterations	Storage (ft^3)	Height (ft)
1	9000	$2.148 * 10^6$	6
2	4500	$2.148 * 10^6$	6
3	3000	$2.148 * 10^6$	6
4	2250	$2.148 * 10^6$	6
5	1800	$2.148 * 10^6$	6
6	1500	$2.148 * 10^6$	6
7	1285	$2.148 * 10^6$	6
8	1125	$2.148 * 10^6$	6
9	1000	$2.148 * 10^6$	6
10	900	$2.148 * 10^6$	6

Once again, the number of iterations remain accurate even at 900 iterations as can be seen by the similar values in max storage and height of the reservoir.

Lastly, the accuracy of number of iterations is checked by measuring the amount of time needed for the reservoir to be emptied out.

Table 3: Comparison of Number of Iterations to the Time it takes to empty a full reservoir.

Timestep	Number of Iterations	Time to Empty (s)
1	9000	$1.073 * 10^5$
2	4500	$1.073 * 10^5$
3	3000	$1.073 * 10^5$
4	2250	$1.073 * 10^5$
5	1800	$1.073 * 10^5$
6	1500	$1.073 * 10^5$
7	1285	$1.073 * 10^5$
8	1125	$1.073 * 10^5$
9	1000	$1.073 * 10^5$
10	900	$1.073 * 10^5$

Using the same set of timesteps and number of iterations. The time to empty the reservoir is shown to be similar even at 900 iterations.

A further check at 900 iterations can be conducted by finding the area under the outflow vs time graph to confirm that the value calculated is similar to the maximum storage.

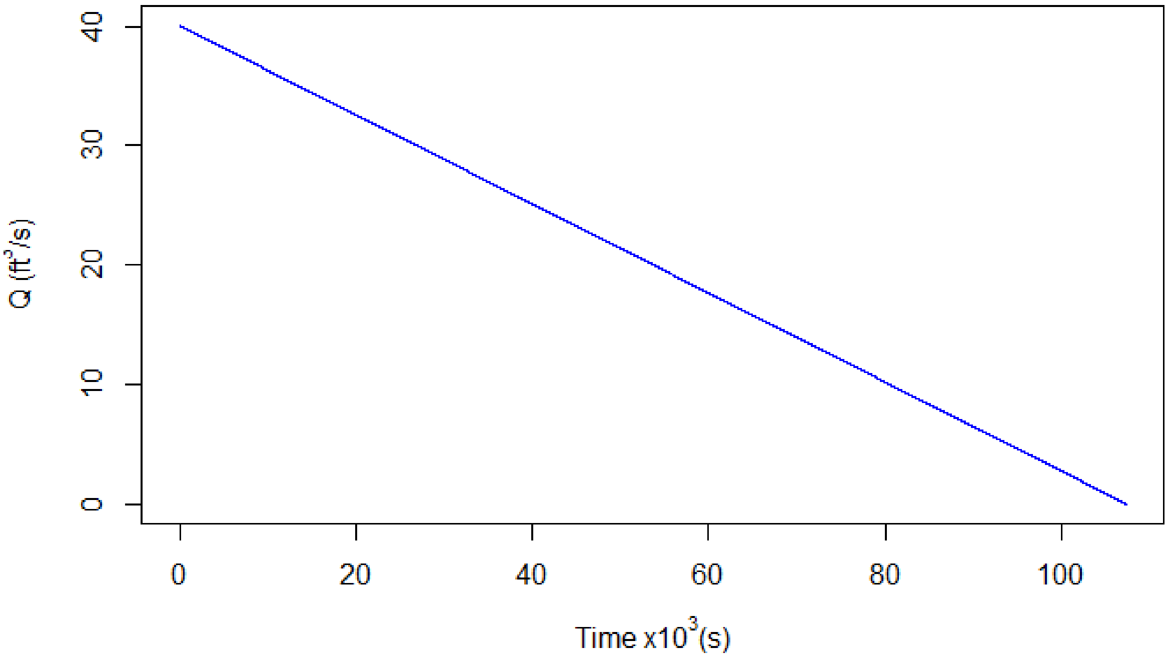

Figure 9: Graph of Emptying Reservoir, Outflow vs Time.

Figure 9 shows the Outflow of an emptying reservoir. The trapezoidal method is used to find the area under the graph. Using equation 2, the storage is found to be 2,148,389. The percentage difference can be calculated as shown below.

$$Percentage\ Difference, \% = \frac{2148570 - 2148389}{2148570}\ x\ 100\% = 0.0084\% \qquad (8)$$

While the storage calculated is not exactly the same, the difference between the two numbers is incredibly small, with a difference of 0.0084%, which is safe to presume that the number of iterations being at 900 is still considered to be accurate.

5. Closure

In order to determine the diameter of the orifice, the client has to either provide the maximum storage or the maximum height that the reservoir can be at. For demonstration purposes a diameter from 1.5 to 5 feet, with intervals of 0.5 feet is suggested.

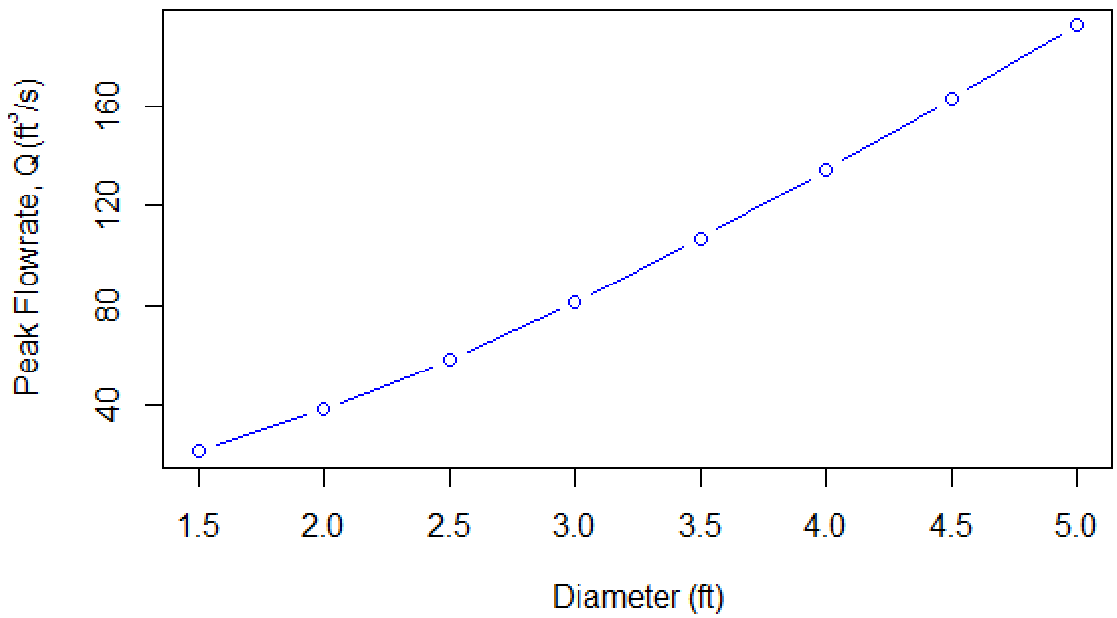

Figure 10: Graph of peak flowrate versus diameter of the orifice

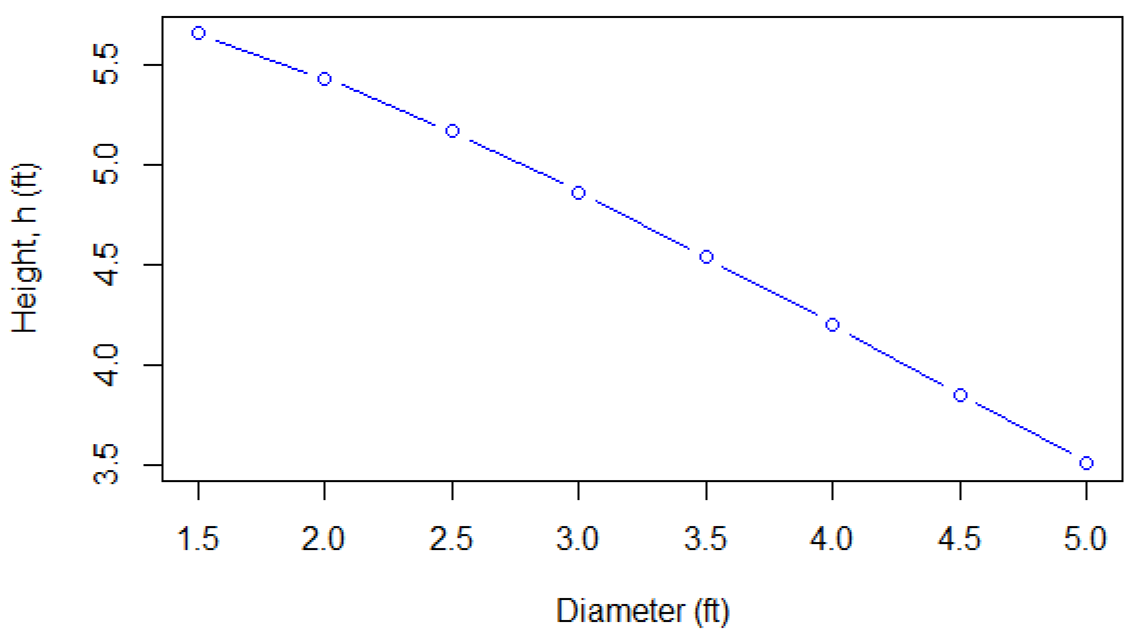

Figure 11: Graph of peak height versus diameter of the orifice

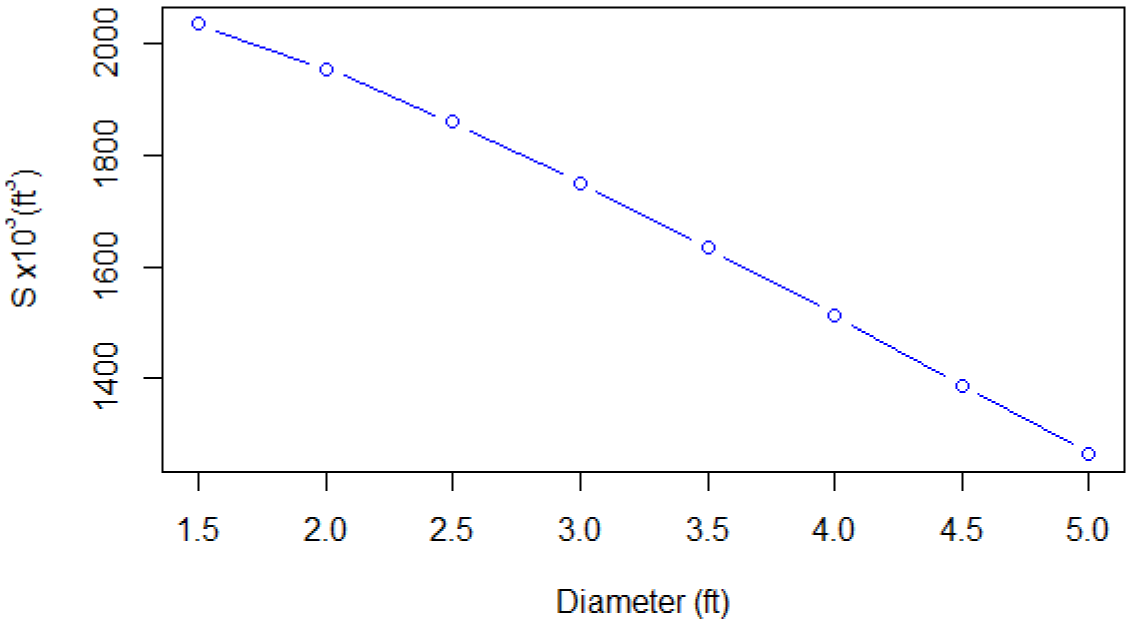

Figure 12: Graph of peak storage versus diameter of the orifice

As can be seen by figures 10,11 and 12, the peak outflow, max height and max storage vary based on the diameter of the orifice used. By specifying peak outflow, max height, or max storage, an orifice can be designed to route the hydrograph. Furthermore, this idea of specifying a particular variable can be extended to providing the time to peak outflow.

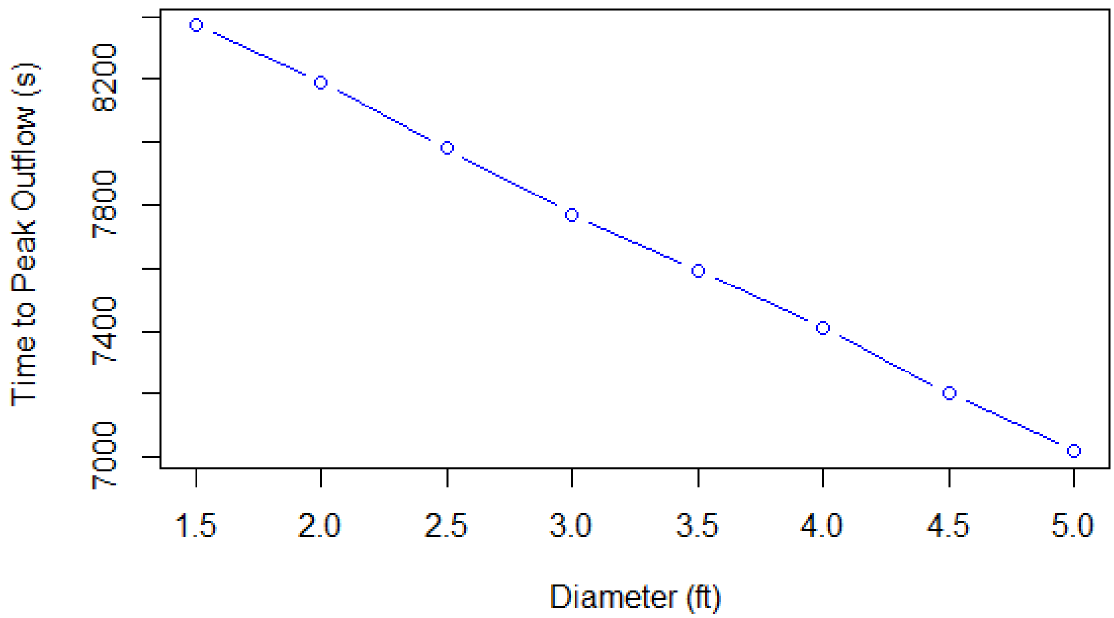

Figure 13: Graph of Time to Peak Outflow versus Diameter of the orifice

As can be seen by Figure 13, the time to peak outflow can also be specified by the client. This can be potentially useful in more tropical climates where the risk of insects breeding in pools of water is something that is of concern.

In conclusion, with figures 10,11,12 and 13, the client can choose between specifying the time taken to reach peak outflow, max height or max flowrate to design an orifice of diameter between 1.5 to 5 feet that will be able to successfully route their hydrograph given in equation 1.

Appendix Code

```
const=0.65*(2*32.2)^0.5

# Parameters: time
# Output: This is the initial Inflow

Q_in = function(t){
  c = 750/pi
  c*(1-cos((pi*t)/4500))
}

# Parameters: Cross sectional area and height
# Output: This is the outflow for one area and one height
Q_out = function(a,H){
  g=32.2
  0.65*a*(2*g*H)^0.5
}


# Parameters: cross sectional area and timestep
# Output: Maximum height one value

hmax_finder = function(A,dt){
  S = 0
  h = 0
  t = seq(0, 9000, by=dt)
  Q = Q_in(t)
  max_time_element = 0
  max_t = 0


  for (j in 1:length(t)){
    S[j+1] = S[j] + ((Q[j] + Q[j+1])*0.5 - const*A*(h[j]^0.5)) *dt
    h[j+1] = S[j+1]/360000
  }
  h = h[1:length(h)-1]
  return (max(h))
}

# Parameters: cross sectional area and timestep
# Output: maximum time one value

tmax_finder = function(A,dt){
  S = 0
  h = 0
  t = seq(0, 9000, by=dt)
  Q = Q_in(t)
```

```
  max_time_element = 0
  max_t = 0
  for (j in 1:length(t)){
    S[j+1] = S[j] + ((Q[j] + Q[j+1])*0.5 - const*A*(h[j]^0.5)) *dt
    h[j+1] = S[j+1]/360000
  }
  h = h[1:length(h)-1]

  max_time_element = match(max(h),h)
  max_t = t[max_time_element]
  return (max_t)
}

# Parameters: cross sectional area and max height
# Output: Q peak one value

Q_peak_finder = function(Area,hmax){
  Q_peak=0

  for (i in 1:length(Area)){
    Q_peak[i] = Q_out(Area[i],hmax[i])
  }
  return(Q_peak)
}

# Parameters: Area, timestep
# Output: A bunch of points for outflow

Q_outflow_finder = function(A,dt){
  S = 0
  h = 0
  t = seq(0, 9000, by=dt)
  Q = Q_in(t)
  Qo = 0


  for (j in 1:length(t)){
    S[j+1] = S[j] + ((Q[j] + Q[j+1])*0.5 - const*A*(h[j]^0.5)) *dt
    h[j+1] = S[j+1]/360000
  }
  h = h[1:length(h)-1]
  Qo = Q_out(A,h)
  return (Qo)
}

# Input A and dt,
# num = 1 gives Q outflow to empty
# num = 2 gives h to empty
# num = 3 gives t to empty
Q_outflow_finder_toempty = function(A,dt,num){
  S = 0
  h = 0
  t = 0
```

```r
  qt = c(seq(0, 9000, by=dt),rep(0,10000))
  Q = Q_in(qt)
  Qo = 0
  j = 1

  repeat{
    S[j+1] = S[j] + ((Q[j] + Q[j+1])*0.5 - const*A*(h[j]^0.5)) *dt
    h[j+1] = S[j+1]/360000
    t[j+1] = t[j] + dt
    j=j+1
    if (S[j]<0){
      break
    }
  }
  h = h[1:length(h)-1]
  t = t[1:length(t)-1]
  Qo = Q_out(A,h)

  if (num == 1 ) {
    return (Qo)
    } else if (num == 2) {
    return (h)
    } else if  (num == 3) {
    return (t)
    } else {
    print('please pick an option')
}
}
```

```r
D_orifice = seq(1.5,5, by =0.5)
cross_A = (pi*D_orifice^2)/4
Area = 600*600
h_max = 0
t_max = 0

DT= 9000/300
t_proj = seq(0,9000, by =DT)

#First find the max heights for respective orifice diameter

for (j in 1: length(cross_A)){
  h_max[j] = hmax_finder(cross_A[j],DT)
}

# Then find the peak Outflow

Q_peak = Q_peak_finder(cross_A,h_max)

# Next find the time to peak
for (j in 1: length(cross_A)){
  t_max[j] = tmax_finder(cross_A[j],DT)
}
```

```r
#Next find the main inflow wave with 3 different outflow curves
Q_inflow = Q_in(t_proj)
Q_outflow_2 = Q_outflow_finder(cross_A[2],DT)
Q_outflow_3 = Q_outflow_finder(cross_A[4],DT)
Q_outflow_4 = Q_outflow_finder(cross_A[6],DT)

Q_to_empty_D3 = Q_outflow_finder_toempty(cross_A[4],DT,1)
h_to_empty_D3 = Q_outflow_finder_toempty(cross_A[4],DT,2)
t_to_empty_D3 = Q_outflow_finder_toempty(cross_A[4],DT,3)
```

```r
plot(D_orifice,h_max,type='l',
     col='blue',
     main = 'Graph of max Height versus Diameter',
     xlab = expression('Diameter (ft)'),
     ylab = expression('Height, h (ft) '),
)
```

```r
plot(D_orifice,Q_peak,type='l',
     col='blue',
     main = 'Graph of Peak Outflow versus Diameter',
     xlab = expression('Diameter (ft)'),
     ylab = expression('Peak Flowrate, Q '*'('*ft^3*'/s)'),
     yaxp = c(0,200,5)

)
```

```r
plot(D_orifice,t_max,type='l',
     col='blue',
     main = 'Graph of Time to peak outflow versus Diameter',
     xlab = expression('Diameter (ft)'),
     ylab = expression('Time to Peak Outflow (s)'),

)
```

```r
# plot the first curve by calling plot() function
# First curve is plotted
 plot(t_proj, Q_inflow, type="l", col="blue", lty=1, lwd = 3, ylim=c(0,500), main = 'Graph of Flowrates
     xlab = expression('Time (s)'),
     ylab = expression('Flowrate, Q '*'('*ft^3*'/s)'))

 lines(t_proj, Q_outflow_4, col="red",lty=1,lwd = 1)


 lines(t_proj, Q_outflow_3, col="dark red", lty=1,lwd = 1)


 lines(t_proj, Q_outflow_2, col="black", lty=1,lwd = 1)


# Adding a legend inside box at the location (2,40) in graph coordinates.
# Note that the order of plots are maintained in the vectors of attributes.
legend(2,450,legend=c("Q_in","D = 4","D = 3","D = 2"), col=c("blue","red","dark red","black"),
                                   lty=c(1,1,1,1), ncol=1)
```

```r
# plot the first curve by calling plot() function
# First curve is plotted
t_thousands = t_to_empty_D3/1000
t_thousands2 = t_proj / 1000
 plot(t_thousands, Q_to_empty_D3, type="l", col="blue", lty=3, lwd = 1, ylim=c(0,500), main = 'Graph of
     xlab = expression('Time '*'x '*10^3*'(s)'),
     ylab = expression('Flowrate, Q '*' ('*ft^3*'/s)'))

 lines(t_thousands2, Q_inflow, col="red",lty=1,lwd = 1)

 legend(40,450,legend=c("D = 3","Q_in"), col=c("blue","red"),
                                   lty=c(3,1), ncol=1)
```

```r
plot(t_thousands,h_to_empty_D3,type='l',
     col='blue',
     main = 'Graph of Storage emptying versus Time',
     xlab = expression('Time '*'x '*10^3*'(s)'),
     ylab = expression('Height, h (ft)')

)
```

```r
Q_to_empty_D2 = Q_outflow_finder_toempty(cross_A[2],DT,1)
h_to_empty_D2 = Q_outflow_finder_toempty(cross_A[2],DT,2)
t_to_empty_D2 = Q_outflow_finder_toempty(cross_A[2],DT,3)
t_thousands = t_to_empty_D2/1000

 plot(t_thousands, Q_to_empty_D2, type="l", col="blue", lty=1, lwd = 1, main = 'Graph of Flowrate vs ti
     xlab = expression('Time (s)'),
     ylab = expression('Flowrate, Q '*'('*ft^3*'/s)'))
```

```r
 plot(t_thousands, h_to_empty_D2, type="l", col="blue", lty=1, lwd = 1, main = 'Graph of height vs time
     xlab = expression('Time (s)'),
     ylab = expression('Height, h (ft)'))
```