Introduction to the Finite Difference Method:
Filling and Draining a Cylindrical Tube

Lensyl Urbano

Tuesday 8$^{\text{th}}$ November, 2022

Chapter 1

Filling a Cylindrical Tube

Consider filling a cylinder with water.

The water flows in at a constant rate of 5 cm^3/s. The inflow rate (Q) can be written as the change in volume (V) over the change in time (t) (the Δ symbol represents change):

Inflow Rate

$$Q = \frac{\Delta V}{\Delta t} = 5\ cm^3/s \tag{1.1}$$

1.1 Filling the Tube Calculations and Equations

1.1.1 Conceptual Physics Approach

So, at this inflow rate, after 10 seconds there will be 50 cm^3 added to the cylinder.

$$\begin{aligned} V &= 5\ cm^3/s \cdot 10\ s \\ &= 50\ cm^3 \end{aligned}$$

In terms of the equation, the change in volume is equal to the inflow rate (Q) times the time period (t).

$$V = Q \cdot t \tag{1.2}$$

How high will the water have risen in the cylinder in those 10 seconds when 50 cm^3 of water was added? Well, we know that the

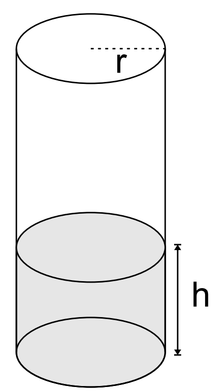

Figure 1.1: Cylinder with dimensions. r is the radius and h is the height of water in the tube.

volume of a cylinder is given by the equation:

$$V = \pi r^2 h \tag{1.3}$$

So, if we know the volume and the radius of the cylinder (r) we can solve this equation for the height (h):

Divide both sides by πr^2:

$$\frac{V}{\pi r^2} = \frac{\pi r^2 h}{\pi r^2} \tag{1.4}$$

To get:

$$\frac{V}{\pi r^2} = h \tag{1.5}$$

Which can be rewritten as:

$$h = \frac{V}{\pi r^2} \tag{1.6}$$

or:

$$h = \frac{1}{\pi r^2} V \tag{1.7}$$

Thus, for our given problem where the radius is 2.25 cm, and the volume of water added is 50 cm^3:

$$h = \frac{1}{\pi \cdot 2.25^2} \cdot 50 = 3.1 \text{ cm} \tag{1.8}$$

Now we can substitute for volume using Equation 1.2 to get:

$$h = \frac{1}{\pi r^2}\, Q \cdot t \tag{1.9}$$

Now, lets rewrite this equation so we just consider what happens over a small time period (call it a *time step* denoted by Δt). It's the change from moment to moment and results in a small change in height (Δh). So our final equation becomes:

$$\Delta h = \frac{1}{\pi r^2}\, Q \cdot \Delta t \tag{1.10}$$

Which we rearrange a little to get:

$$\boxed{\Delta h = \frac{\Delta t}{\pi r^2}\, Q} \tag{1.11}$$

Having calculated the change in the height of the water in the cylinder in a given time step (Δt), for each timestep we calculate the new height of water (h_{new}) as the old height plus the change:

$$\boxed{h_{new} = h_{old} + \Delta h} \tag{1.12}$$

We can now use these two equation to create a computer model that gives the height of water in the column over time.

1.1.2 Using Calculus to Find the Discrete Equations

Same problem–filling a cylinder–but using calculus to end up with the same equations in the end.

Start with the equation for the volume of a cylinder:

$$V = \pi r^2 h \tag{1.13}$$

There are two variables that change with time as the cylinder fills, the volume (V) and the height (h) since the radius (r) does not change. So, if we differentiate this equation with respect to time (implicit differentiation), we get:

$$\frac{dV}{dt} = \pi r^2 \frac{dh}{dt} \tag{1.14}$$

Solving for $\frac{dh}{dt}$ gives the **height change equation**:

$$\frac{dh}{dt} = \frac{1}{\pi r^2} \frac{dV}{dt} \tag{1.15}$$

The expression $\frac{dh}{dt}$ represents the instantaneous change in height with time: the rate at which height changes at any instant. To write a program to solve this equation we'll **discretize** the expression by using $\frac{\Delta h}{\Delta t}$:

$$\frac{\Delta h}{\Delta t} \approx \frac{dh}{dt} \tag{1.16}$$

The Δ means that we're taking the difference between two discrete value of h, so:

$$\Delta h = h_2 - h_1 \tag{1.17}$$

Since this is the rate of change over time it can be easier to think of the change in height as the difference between the new height and the old height over the short (Δt) time period.

$$\Delta h = h_{new} - h_{old} \tag{1.18}$$

So now we rewrite our height change equation (Eqn. 1.15) as:

$$\frac{\Delta h}{\Delta t} = \frac{1}{\pi r^2} \frac{dV}{dt} \tag{1.19}$$

which we can solve for the change in height (Δh):

$$\Delta h = \frac{\Delta t}{\pi r^2} \frac{dV}{dt} \tag{1.20}$$

Since the inflow rate (Q) is the change in volume over time, and it remains constant for our model, we can say:

Change in Height Equation

$$\boxed{\Delta h = \frac{\Delta t}{\pi r^2}\, Q} \tag{1.21}$$

Which is the same equation (Eqn. 1.11) we found when we took the conceptual approach in the previous section.

Now we substitute in the discrete change for Δh (Eqn. 1.18) to get:

$$h_{new} - h_{old} = \frac{\Delta t}{\pi r^2}\, Q \tag{1.22}$$

Which we can solve for the new height:

$$h_{new} = h_{old} + \frac{\Delta t}{\pi r^2}\, Q \tag{1.23}$$

Which is the same as:

Height Update Equation

$$\boxed{h_{new} = h_{old} + \Delta h} \tag{1.24}$$

We can use the **Change in Height** (Eqn. 1.21) and **Height Update** (Eqn. 1.18) equations to create a computer model of the height of the water in the cylinder as it fills it up.

1.2 Code

The following example code that solves this water-filling problem uses the ezGraph class (https://github.com/lurbano/ezGraph) which requires matplotlib and numpy. However, the code in the following section avoids the use of most imported modules, but does not graph.

Code with Graphical Output

Code 1.1: Model of a filling tube with graphical output (*water-filling.py*)

```
import numpy as np
import time
from ezGraph import *

# Finite Difference Model

# PARAMETERS
dt = 1.
nsteps = 20

r = 2.25      # radius (cm)
Qin = 5       # Volume inflow rate: (cubic cm / s)
h = 0         # Initial height (cm)

# GRAPH
graph = ezGraph(xmax=30, ymin=0, ymax=10, xLabel="Time (s)", yLabel="Height (cm)")
graph.add(0, h)                    # add initial values


# TIME LOOP
for t in range(1, nsteps):
    modelTime = t * dt

    dh = Qin * dt / (np.pi * r**2)      # find the change in height
    h = h + dh                          # update height

```

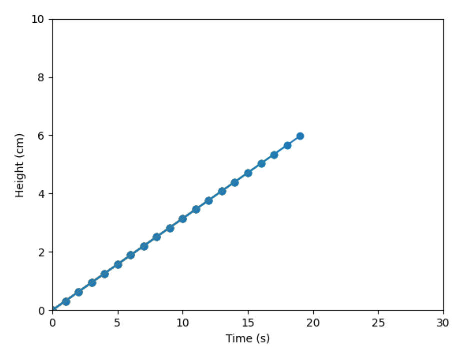

Figure 1.2: Model output: Graph of height of water in the column over time when filling the cylinder.

```
    print(modelTime, h)
    graph.add(modelTime, h)
    graph.wait(0.1)

# DRAW GRAPH
graph.keepOpen()
```

1.2.1 Code without Graphical Output

A stripped down version of the code with no graph and no external modules except ”math”.

Code 1.2: Model of a filling tube without graphing (*water-filling-fd-noGraph.py*)

```
import math
# Finite Difference Model

# PARAMETERS
dt = 1.
nsteps = 20

r = 2.25      # radius (cm)
Q = 5         # Volume inflow rate: (cubic cm / s )
h = 0         # Initial height (cm)

print(0, h)       # print initial values

# TIME LOOP
for t in range(1, nsteps):
    modelTime = t * dt

    dh = Q * dt / (math.pi * r**2)     # find the change in height
    h = h + dh                          # update height

    print(modelTime, h)
```

Which should produce a table of time and height output:

```
0 0
1.0 0.31438013450250935
2.0 0.6287602690050187
3.0 0.943140403507528
```

```
4.0  1.2575205380100374
5.0  1.5719006725125468
6.0  1.8862808070150563
7.0  2.2006609415175657
8.0  2.515041076020075
9.0  2.8294212105225847
10.0  3.143801345025094
11.0  3.4581814795276036
12.0  3.772561614030113
13.0  4.086941748532622
14.0  4.4013218830351315
15.0  4.715702017537641
16.0  5.03008215204015
17.0  5.34446228654266
18.0  5.658842421045169
19.0  5.973222555547679
```

1.3 Analytical Solutions using Calculus

The analytical solution to this problem will help confirm the accuracy of our model.+

1.3.1 Filling

As we saw in the section on using calculus (Section 1.1.2), we can start with the equation for the volume of a cylinder:

$$V = \pi r^2 h \tag{1.25}$$

And differentiate with respect to time to get:

$$\frac{dV}{dt} = \pi r^2 \frac{dh}{dt} \tag{1.26}$$

Assuming a constant inflow rate ($\frac{dV}{dt} = Q$):

$$Q = \pi r^2 \frac{dh}{dt} \tag{1.27}$$

And solve for $\frac{dh}{dt}$:

$$\frac{dh}{dt} = \frac{Q}{\pi r^2} \tag{1.28}$$

This we can separate:

$$dh = \frac{Q}{\pi r^2} dt \tag{1.29}$$

and integrate:

$$\int dh = \int \frac{Q}{\pi r^2} dt \tag{1.30}$$

to get:

$$h = \frac{Q}{\pi r^2} t + c \tag{1.31}$$

When $t = 0$, c can be shown to be the initial height ($c = h_i$) so:

$$h = \frac{Q}{\pi r^2} t + h_i \tag{1.32}$$

And since $\frac{Q}{\pi r^2}$ is constant, we can see that this term is the slope in a linear equation of the form.

$$y = mx + b \tag{1.33}$$

So the linear pattern produced by the filling model is correct (Figure 1.2).

Chapter 2

Draining

Consider a cylinder with water draining out of the bottom through a hole.

2.1 Numerical Solution

For draining, the outflow rate (change in volume over time, ($Q = \frac{dV}{dt}$) is not constant. The outflow rate is proportional to the height of water in the tube, since the higher the water level the greater pressure at the bottom of the tube and the faster the outflow rate.

$$\frac{dV}{dt} \propto h \tag{2.1}$$

Converting the proportionality statement to an equation requires us to introduce a constant (k). Also, recognizing that this will be an outflow rate means that the flow rate should be negative:

$$\frac{dV}{dt} = -k \cdot h \tag{2.2}$$

As we saw when we were filling the cylinder, the change in height of water in the tube is given by the **change in height** equation:

$$\boxed{\Delta h = \frac{\Delta t}{\pi r^2}\, Q} \tag{2.3}$$

where Q is $\frac{dV}{dt}$ so:

$$\Delta h = \frac{\Delta t}{\pi r^2}\, \frac{dV}{dt} \tag{2.4}$$

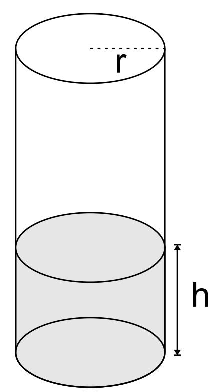

Figure 2.1: Cylinder with dimensions. r is the radius and h is the height of water in the tube.

So, let's substitute our flow rate equation (Eqn. 2.2) for $\frac{dV}{dt}$ to get:

$$\Delta h = \frac{\Delta t}{\pi r^2} \left(-k \cdot h\right) \tag{2.5}$$

which simplifies to:

$$\Delta h = -\frac{\Delta t}{\pi r^2}\, k \cdot h \tag{2.6}$$

Important to note for the computer model, is that the height (h) used in this equation is the old height from the previous timestep so:

$$\boxed{\Delta h = -\frac{\Delta t}{\pi r^2}\, k \cdot h_{old}} \tag{2.7}$$

and we can still update the new height using:

$$\boxed{h_{new} = h_{old} + \Delta h} \tag{2.8}$$

so, in our code we just need to change this line and set up a few different constants.

2.1.1 Code: Draining

This program is based off the filling code, but for this example we ignore the filling by setting the inflow rate to zero (**Line 12**). We're using an initial height of 50 cm ($h_0 = 50$), and set the constant k to be equal to one.

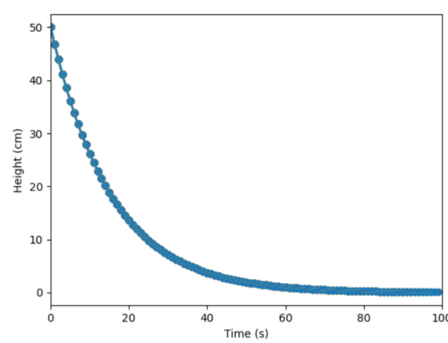

Figure 2.2: Model output: Graph of height of water in the column over time when draining the cylinder via a hole in the bottom.

Code 2.1: Model of water draining from a tube (*water-draining-fd.py*)

```
1  import numpy as np
2  import time
3  from ezGraph import *
4
5  # Finite Difference Model
6
7  # PARAMETERS
8  dt = 1
9  nsteps = 100
10
11 r = 2.25      # radius (cm)
```

```
Qin = 5          # Volume inflow rate (dV/dt): (cubic cm / s)
h = 0            # Initial height (cm)
k = 1.0          # outflow rate constant

# GRAPH
graph = ezGraph(xmax=100,
                xLabel="Time (s)", yLabel="Height (cm)")
graph.add(0, h)                 # add initial values


# TIME LOOP
for t in range(1, nsteps):
    modelTime = t * dt

    # Filling
    dh = Qin * dt / (np.pi * r**2)  # find the change in height
    h = h + dh                      # update height

    # Draining
    dVdt = -k * h
    dh = dVdt * dt / (np.pi * r**2)
    h = h + dh

    print(modelTime, h)
    graph.add(modelTime, h)
    graph.wait(0.1)

# DRAW GRAPH
graph.keepOpen()
```

The output from the model (Fig. 3.1) looks like an exponential decay curve, which is what we will find from the analytical solution (Eqn. 2.24).

2.2 Draining: Analytical Solution using Calculus

Experiments (which you may have done) show that if you're draining a cylinder by gravity the outflow rate of water is linearly proportional to the height of water in the tube.

$$\frac{dV}{dt} \propto h \tag{2.9}$$

Converting the proportionality statement to an equation requires us to introduce a constant (k):

$$\frac{dV}{dt} = kh \tag{2.10}$$

So in draining, the outflow rate ($\frac{dV}{dt}$) is not constant, it slows down as the height of water in the tube decreases.

Now, lets substitute the equation for the volume of a cylinder:

$$V = \pi r^2 h \tag{2.11}$$

into the draining equation (Eq. 2.10) to get:

$$\frac{d[\pi r^2 h]}{dt} = kh \tag{2.12}$$

we can extract π and r^2 from the differential because they are constant:

$$\pi r^2 \frac{dh}{dt} = kh \tag{2.13}$$

separating the variables gives:

$$\pi r^2 \frac{dh}{h} = k \cdot dt \tag{2.14}$$

and rearranging:

$$\frac{dh}{h} = \frac{k \cdot dt}{\pi r^2} \tag{2.15}$$

$$\frac{dh}{h} = \frac{k}{\pi r^2} dt \tag{2.16}$$

To simplify a little, lets consolidate the constants on the left hand side into one variable K:

$$K = \frac{k}{\pi r^2} \tag{2.17}$$

so:

$$\frac{dh}{h} = K \cdot dt \tag{2.18}$$

which we can integrate (remember K is a constant):

$$\int \frac{dh}{h} = K \int dt \tag{2.19}$$

$$\ln h = K \cdot t + c \tag{2.20}$$

we can solve for h by raising both sides by e to cancel the ln:

$$e^{\ln h} = e^{Kt+c} \tag{2.21}$$

$$h = e^{Kt+c} \tag{2.22}$$

Because of math, we can pull the constant out to get:

$$h = Ce^{Kt} \tag{2.23}$$

Where the constant is the initial value of the height (h_0):

$$h = h_0 \cdot e^{Kt} \tag{2.24}$$

This is an exponential function. If K is less than 1 ($K < 1$) then this is a decay curve.

Chapter 3

Model Calibration

We have a model that shows the general patterns we expect when filling and draining the tube: linear increase for filling at a constant rate, and exponential decay for draining due to gravity. But can these models reflect the actual thing?

Fortunately, we have some experimental data, thanks to my pre-Calculus class.

We'll start with the filling portion of the model.

3.1 Filling Calibration

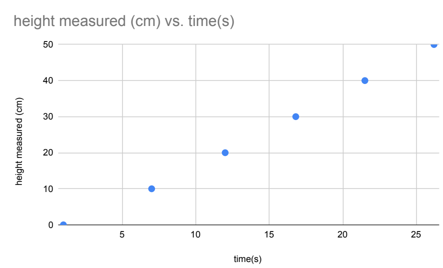

Figure 3.1: Experimental data from filling the cylinder. Data by Blas.

time (s)	height (cm)
1	0
7	10
12	20
16.8	30
21.5	40
26.2	50

Table 3.1: Experimental Data from filling the cylinder. Data by Blas.

Recall that, at the core of the model, is the change in height equation (Eqn. 1.11):

$$\Delta h = \frac{\Delta t}{\pi r^2}\, Q$$

The radius (r) is measured and we choose Δt as steps in the simulation, so the only unknown is the inflow rate (Q).

Therefore, to calibrate the model we adjust Q until the model output matches the experimental data.

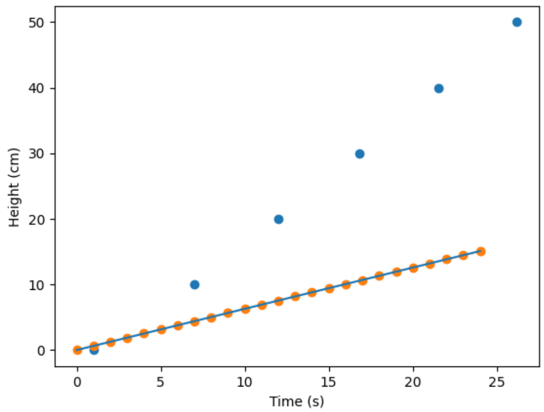

Figure 3.2: Comparison of measured (blue) and modeled (red) data. This version of the model uses $Q = 10$ cm^3/s.

In order to produce this graph, we use the *ezGraphMM* class, which allows us to plot the measured and modeled data separately. The full code is here:

Code 3.1: Model showing measured data and modeled output. (*water-filling-calibration.py*)

```
import numpy as np
import time
from ezGraph import *

def myAvg(lst):
    # sum
    s = 0
    n = 0
    for i in lst:
        s = s + i
        n += 1
    return s/n

# Finite Difference Model

# PARAMETERS
```

```python
dt = 1
nsteps = 30

r = 2.25      # radius (cm)
Qin = 30        # Volume inflow rate (dV/dt): (cubic cm / s)
h = 0         # Initial height (cm)
k = 0.0       # outflow rate constant

# EXPERIMENTAL DATA
x_measured = [1, 7, 12, 17, 22, 26]
y_measured = [0, 10, 20, 30, 40, 50]

# GRAPH
graph = ezGraphMM(xmin=0, xmax=100,
                  xLabel="Time (s)",
                  yLabel="Height (cm)",
                  x_measured = x_measured,
                  y_measured = y_measured)

graph.addModeled(0, h)                # add initial values


# TIME LOOP
for t in range(1, nsteps):
    modelTime = t * dt

    # Filling
    dh = Qin * dt / (np.pi * r**2)  # find the change in height
    h = h + dh                      # update height

    # Draining
    dVdt = -k * h
    dh = dVdt * dt / (np.pi * r**2)
    h = h + dh

    graph.addModeled(modelTime, h)
    graph.wait(0.01)

# DRAW GRAPH
graph.keepOpen()
```

3.2 How Good are Our Results: Finding the r^2 Value

To determine how good a match we have (instead of just eyeballing it) we'll calculate the regression coefficient (r^2) value using a series of assignments.

3.2.1 Residuals (a.k.a. Errors)

The *residual*, also sometimes called the *error* is the difference between the value the model predicts and the actual measured values. This is shown in Figure 3.3. The worse the model prediction, the greater the error.

We need to add up all the errors in such a way as the resulting value tells us how good the model fits the data.

3.2.2 Selecting time data

The first thing we need to do is have the model record its predictions at each point in time when there was a measurement taken (see Table 3.1).

The objective is to have three lists: one of times (x values) when height was measured; one of measured heights; and one of modeled heights. In the parlance of the code above (Code 3.1) we already have the time ("x_measured") and measured heights ("y_measured") data.

```
x_measured = [1, 7, 12, 17, 22, 26]
y_measured = [0, 10, 20, 30, 40, 50]
```

We now just need to collect the modeled data.

> **Assignment 1(water-array)**:
> Adapt the model so it records the water heights (in an array) at the same times as the measured data (you may round the measured times to whole integers to make it easier).

The code that produced Figure 3.3, uses $Q = 25$ and an initial height value of $h = 5$ which produces a list of modeled heights ("y_modeled") that look like this.

```
[6.57, 16.00, 23.86, 31.72, 39.58, 45.87]
```

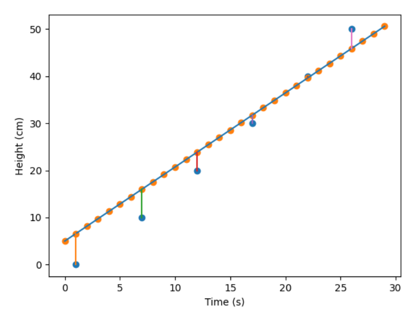

Figure 3.3: The error (or residual) is the difference between the measured and modeled value at a given point in time. These are shown as the vertical lines.

Hint for Assignment water-array:
This is a hint.