Demand Factor Analysis Using the Box Complex-Method

By Aidan Gill

1. Questions

Something many folks in the first world take for granted is the instant access to clean, safe drinking water at the simple turning of a faucet handle. This water is expected to be both cheap and as reliable as the rising and setting of the sun. Looking to our not so distant past however reveals a world riddled in disease, scarcity, and the unknown when it comes to water distribution. What changed?

Since the dawn of the computer age around the 1950's, methods for solving the complex non-linear equations associated with hydraulic modelling became widely available for use in distribution network analysis (Gill and Ormsbee, 2023). This allowed for the practical design of these systems and is the framework upon which modern network planning finds its roots.

While using these tools for design may be readily accomplished, the lack of predictable demand patterns for a system on any given day is a significant barrier in using hydraulic models for daily management.

Essentially, the inability to process live meter and tank data on the fly to back out household consumption (demand) throughout the system causes inefficiencies in basic daily processes. These processes include but are not limited to pumping operations, and emergency scenario response.

The goal of this study is to investigate an optimization technique known as the Box-Complex method to see if it will be able to back out system demands for multi-zonal systems that will aid in processing live data for operator planning and management. Python skills and optimization terminology learned in CE 610 will be employed in this research.

2. Methods

2.1. Data

It was the authors intention to use real data to test the algorithm in the Whitesburg, Kentucky water system. However, that data has not yet been made available and synthetic demands will be used for this report.

What the algorithm is doing here is testing demands within each of the pressure zones within a water distribution system that will cause the water levels in the tanks to match what they look like in reality (this will be our objective function). The limitations on the length of this paper force a not so thorough discussion on the relevant terms above but suffice it to say the following:

1) We can break a water distribution network into demand management areas (aka pressure zones) that contain the summated average values of my water demand (usually assumed values) as well as all my neighbors within a specific zone (figure 1).

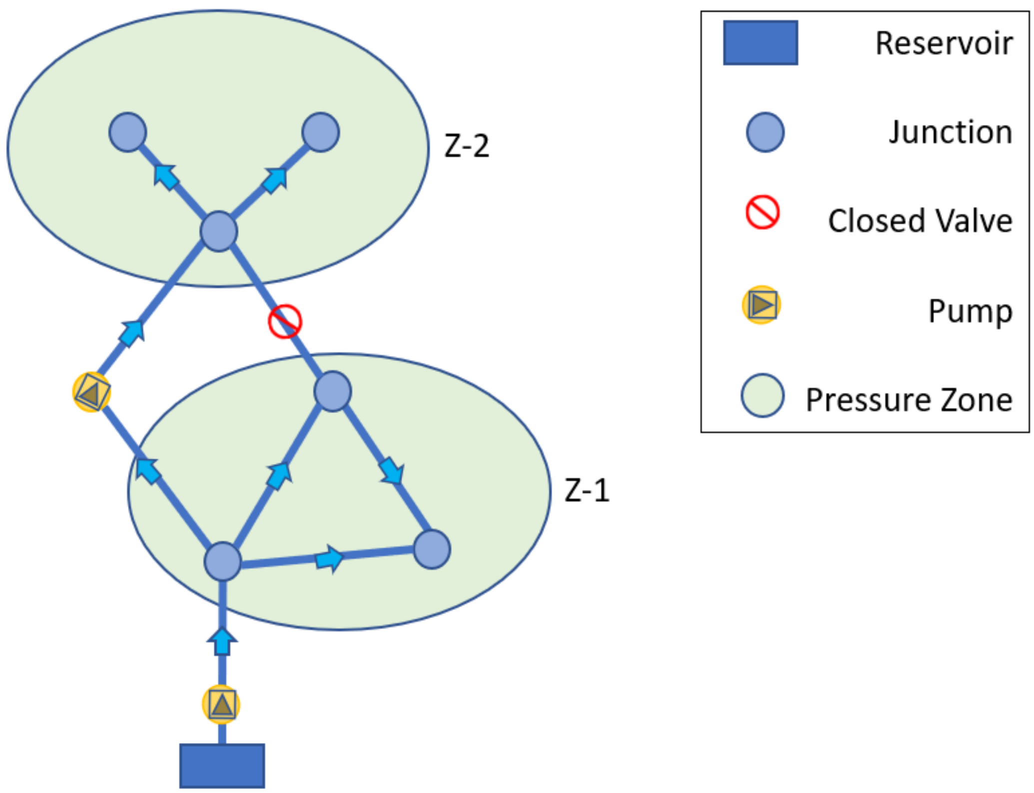

Figure-1: A Two Zone System with Junctions (Households)

2) Because master meters (selling points to other utilities), tank level sensors, and pumps are typically capable of sending live data, the only unknown in our mass balance is the consumption (demand) within each of the pressure zones:

$$\Sigma(DF_{n,t_i}) = Q_{plant,t_i} + Q_{tank,t_i} - Q_{sold,t_i}$$

Q_{sold,t_i} is flow in gpm sold to other systems at time t

Q_{tank,t_i} is flow in gpm (where flow out of the tank is taken as positive) at time t

Q_{plant,t_i} is the flow into the system from the water treatment plant (gpm)

$\Sigma(DF_{n,t_i})$ is the sum of all demands for each of the zones (in gpm) at time t

3) Given that we do not currently have real data to work with for the system, we will assign random demand factors for each zone within Whitesburg Kentucky Water System (figure 2), record the tank levels and see if the optimization algorithm can back these factors out.

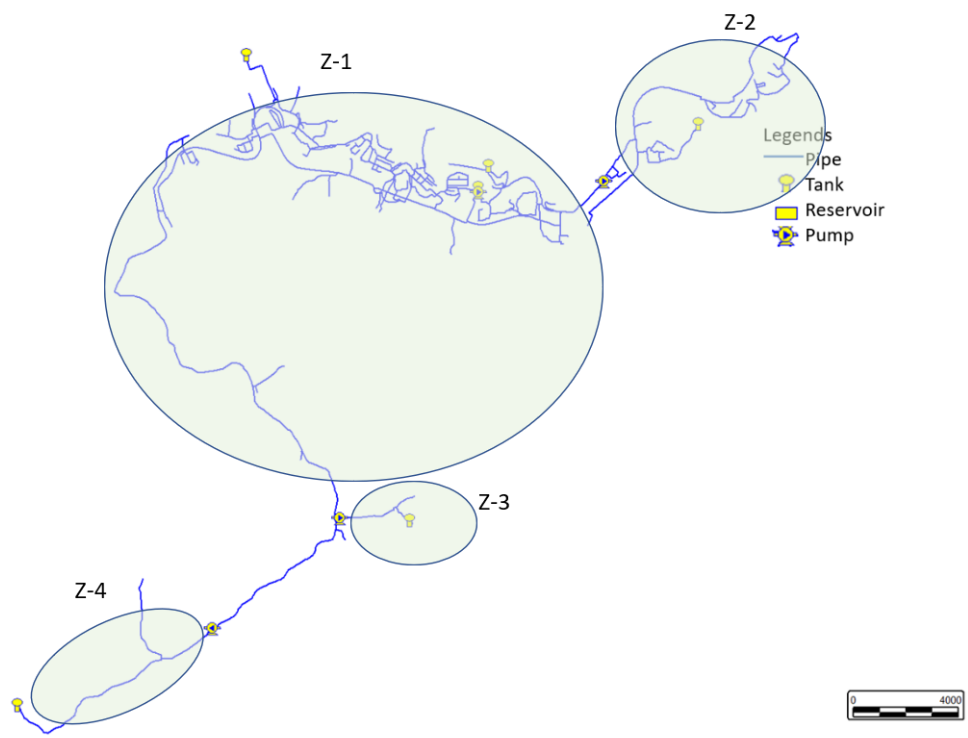

Figure-2: Whitesburg Water System with Delineated Pressure Zones

2.2. Models

The Box-Complex method in simple terms tests (n+1) points in n dimensional space (think triangles in 2-d, a tetrahedron in 3-d, etc). Each point contains "pseudo random" demand factors that satisfy the conservation of mass for the system at each time step (see the more thorough discussion in Gill and Ormsbee, 2023). The worst of these points is manipulated across the simplex and tests a new region. If the new point is better it is kept and the old point is thrown away! See the flow chart created for my thesis (figure 3).

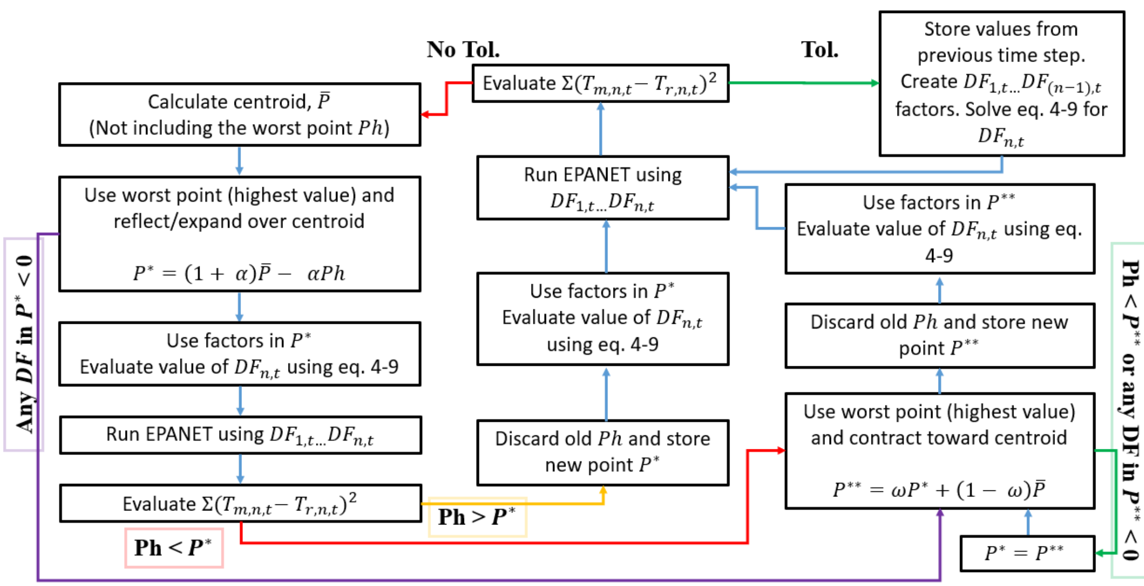

Figure-3: Logic of the Box-Complex Method

Synthetic demands have been created and may be seen in table 1. Accompanying these demands are the tank levels within the model that are a function of the demands. The code has been created within Jupyter notebook using python skills learned in class and seeks to optimize the following objective function:

$$f(DF_{z1,t}...DF_{zn,t}) = \Sigma(T_{m,zn,t} - T_{r,zn,t})^2$$

Where $DF_{z1,t}...DF_{zn,t}$ are the demands within each of the zones and $\Sigma(T_{m,zn,t} - T_{r,zn,t})^2$ is the mean squared error between the "real" tank levels and the model tank levels.

Time (Hour)	Zone	Junctions	Synthetic Demand Factor	Tank Level (at the end of the hour in feet)
0	1	226	1	1411.43
0	2	48	4	1468.59
0	3	3	10	1484.67
0	4	18	6	1655.82
1	1	226	0.5	1402.19
1	2	48	6	1468.79
1	3	3	8	1484.60
1	4	18	10	1655.01

Table-1: Synthetic Results Using Manufactured Demands

3. Results

As expected, the code was found capable of processing the demand factors (see table 2). Given a cursory understanding of demands in water distribution analysis and design this may seem insignificant. However, what this project shows is the capability of this algorithm to handle large systems and backout their respective zonal demands based on tank, pump, and meter information.

Translating this procedure to Python and confirming the ability of the Box-Complex method to work within this environment opens the possibility for utilities across the country to incorporate open-source tools like this in optimizing their daily operations.

Time (Hour)	Zone	Junctions	Synthetic Demand Factor	Tank Level (at the end of the hour in feet)	Calculated Demand Factor
0	1	226	1	1411.43	1.005
0	2	48	4	1468.59	3.98
0	3	3	10	1484.67	10.004
0	4	18	6	1655.82	5.98
1	1	226	0.5	1402.19	0.509
1	2	48	6	1468.79	6.015
1	3	3	8	1484.60	8.02
1	4	18	10	1655.01	9.84

Table-2: Results from Using the Box-Complex Method