Construction-Driven Execution Design Challenge: PUMP IT UP (Template)

Contents

[I. Introduction 2](#_Toc18424623)

[II. Hydraulic Calculations Explanation 2](#_Toc18424624)

[III. Excavation Calculations 2](#_Toc18424625)

[IV. Foundation Calculations 2](#_Toc18424626)

[V. Modularization (Pros and Cons) 2](#_Toc18424627)

[VI. Zero Based Execution (show calculations) 2](#_Toc18424628)

[VII. Risk Assessment 3](#_Toc18424629)

[VIII. Innovation Idea 3](#_Toc18424630)

## Introduction

The AquaDucks team is composed of three mechanical engineers who wanted to push the limits of their problem-solving abilities. Joseph Amar, the team lead, works for NASA as a co-op student at Johnson Space Center. Zach McBurney works as an engineering intern at a construction firm in League City, and Saul Pizano has significant experience with CAD modeling and building concepts, but seeks to hone his other engineering skills.

This task was extremely difficult to complete from a time perspective. Even spending nearly every free minute on the project was not enough; the challenge was almost two weeks underway when we learned of it, making every second count that much more. This was the greatest difficulty of the whole project: gaining a firm grasp of the challenge and its interconnected variables, developing techniques for optimizing these variables with respect to cost, and doing it all in about two weeks while balancing school and personal obligations.

The three most important/useful lessons gained through this process were:

1. We are always capable of accomplishing more than we initially think is possible.
2. Your solution can only take advantage of combinatorics if you know what that word means.
3. Heroic efforts are all well and good, but much more valuable is consistency: do a little bit at a time, regularly, until you cross the finish line.

## Excavation Calculations

Excavation calculations were performed by casting the pipeline onto a 2-dimensional xy-coordinate system. We are given the elevation at each junction (y-value) and the distance along the pipeline. From this information, calculating the x-values at each junction is trivial. The system is then divided up into 20,000 separate sections, each one covering ten feet of the pipeline. These sections are each assigned a soil type based on the given information, and these soil types are each built into dictionaries with the attributes maximum bearing pressure, maximum slope, foundation depth, and excavation cost. Figure 2 shows the derivation for iteratively calculating how much soil needs to be excavated to create a 500’x500’ flat section.

This amount of soil is given as a volume; this volume is used to determine the number of track-hoe-days necessary to excavate this amount of soil. Since this needs to be an integer, floor division was used, and a 1 added to the result. I.e., if a track hoe can move 100 [ft3/day] and the excavation calls for 450 [ft3] to be moved, then floor division (450 [ft3]//[100 ft3/day]) = 4 [track-hoe-days]. Adding 1 to the result means we end up with 5 [track-hoe-days], so one track hoe could finish it in five days, or five track hoes could do it in one day. A simple check of where the excavation occurs tells what the soil type is, determining whether it needs to be multiplied by a factor of three (for stable rock only). This section of code outputs two lists: the first[[1]](#footnote-1) is sorted by amount of material that needs to be removed from each location, and the second[[2]](#footnote-2) is sorted by the actual excavation cost, with soil type, track hoe usage, and dump truck usage. Dump truck usage is calculated by a method identical in concept to the track hoe method.

## Foundation Calculations

Calculating the foundation was easier than it initially seemed. Each pump was put into a dictionary with all its relevant data. An important part of this process was that the length and width of the baseplate were put into separate key-value pairs; this made programmatically determining the length/width of the pedestal much simpler. A section of code was written to iterate through each of the 19 junction-and-pump options developed in The Hydraulics Calculations sections. Some of the calculations[[3]](#footnote-3) for how the program was designed to compute the foundation size are shown in Figure 3. As mentioned, this was easier than expected. The reason for this was because the soil bearing pressure for all pumps and locations (in the list of 19 viable) was sufficient to support the pedestal without a footing, decreasing complexity and required excavation. However, the program as written still has the capability to determine footing size if necessary. A standard concrete unit-weight of 145 [lb/ft3] was used, and a floor division technique similar to the dump truck and track hoe method was used to determine the concrete-truck-days required.

Once foundation size was determined, the last piece was to excavate the necessary volume in order to safely pour the foundation on which the pump will sit. This was done by first determining the soil type at the pump location. According to calculations based on Figure 5, if the soil type changes at the junction, preference for the foundation location should be awarded to the material with a higher maximum slope *unless* that material is rock. By these parameters, digging into stable rock is never preferable and should be avoided.

## Hydraulic Calculations Explanation

The hydraulics portion of the design consists of the pipe thicknesses along with the pump choice and location. Using the AFT Fathom software, it was quickly established that with all else held equal, the maximum static pressure in the system would be minimized by placing the pump location further toward the outlet. However, this minimization technique has a built-in limiter: the pump cannot be placed so far down the line that the pressure at its inlet is negative. In fact, each pump has a criterion known as Net Positive Suction Head (NPSH), which is required at its inlet in order to avoid cavitation. The NPSH Available (NPSHA) scales linearly with the pressure in the line; so, place the pump too far down the line, and it will cavitate. Place it too close to the source, and it drives up the maximum pressure in the system, increasing the required wall thickness for the pipes, and therefore, their mass and cost.

Two major assumptions were made throughout this process:

1. Only the (5) different schedules of 30” pipe wall thickness available in AFT Fathom could be used in the various mill runs.
2. An increase in wall thickness (at the scales available in the aforementioned schedules) produces a negligible decrease in the static pressure of that line (Bernoulli’s equation).
3. Pipe schedule can only change at junctions, not between them.

As directed, only the “STD” schedule wall thickness was used in the AFT Fathom simulation. In this simulation, the pump was placed at each junction, 4 through 12, and the results exported. The first three junction are ignored because they correspond the first 30,000 feet of the pipeline, which has been excluded from eligibility for the pump placement. The NPSHA and maximum system pressure are shown in Figure 1.

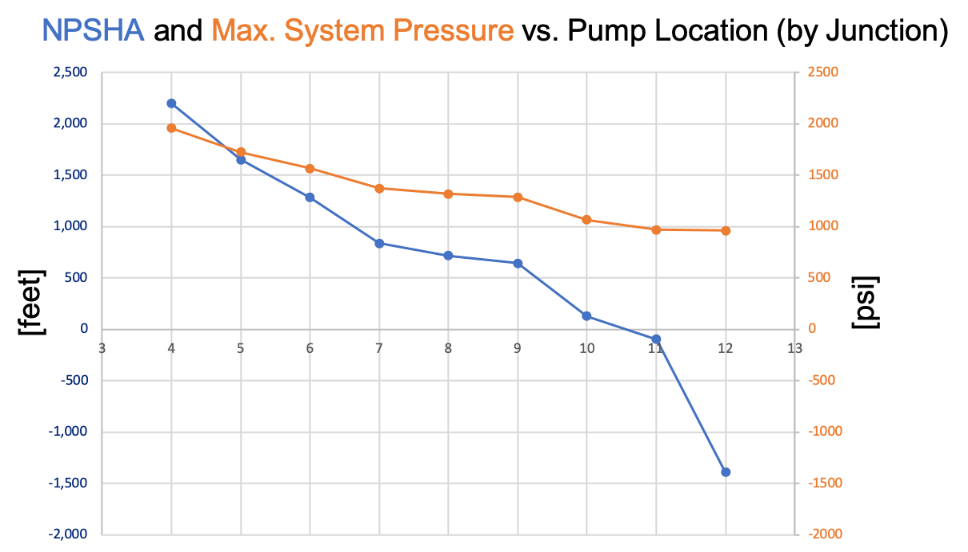


Figure 1. Visualization of rate of change of both NPSHA and System Pressure with respect to Pump Location by Junction. Results developed by AFT Fathom, all other variables held constant.

Each available pump has a different required NPSH value. By inputting these parameters into the program, we were able to determine which pumps are viable at each location. Because only one pump may be used in this initial evaluation, pumps not meeting the minimum required power of 6,545 [hp] were discarded from the iterations. This[[4]](#footnote-4) generated a list of 19 items: pumps A, B, and E were all viable at junctions 4, 5, 6, 7, 8, and 9. Only pump E was viable at junction 10, due to the pump’s low requirement for NPSHA.

Combinatorics were used in the minimization of the pipe schedules. Specifically, the technique known as “n choose k” was researched and applied. This created all the possible lists of 3 combinations of the 5 available pipe schedules. Each of these lists is ordered from lowest to highest wall thickness. For each section of piping (between junctions) the lowest thickness in the list is checked based on the calculation given in Section IIIb. Pipeline Mechanical Design of the guidelines. If this hoop stress calculation is successful (i.e., thickness is sufficient for maximum pressure in pipe section), the volume of the section is calculated based on the wall thickness[[5]](#footnote-5). If the hoop stress calculation fails, the next thickness in the list is tested. If all three thicknesses fail, the section is not added to the list. In the next code block, all sub-lists with less than 12 entries are discarded, because these are the lists where, for at least one section of pipe, none of the thicknesses were great enough to withstand the pressure. This leaves us with a list of tuples in the following format: (Total pipeline volume, (t1, t2, t3), Junction #). Two more nearly identical lists were generated from this data, with volume replaced by weight and cost in each list, respectively. Each of these lists is 231 items long.

This list tells us that the 14 cheapest piping options all occur at junction 10. The absolute cheapest option costs $ 59,189,978 (including both steel cost *and* pipeline construction), using pipe schedules 5, 10, and 20. The quantity of each schedule is specified in the Cost spreadsheet.

The combination of each of the previous three sections yields the following:

1. Pump E
2. Junction 10
3. Foundation dug on the ‘Firm Clay’ side
4. Pedestal 16’x16’, 4’ depth, no footing
5. 3 mill runs: pipe schedules 5, 10, and 20

Total Cost: $ 66,751,979

Calculations; iterative example; image

## Modularization (Pros and Cons)

This is the process in which we utilize a compact design of process equipment and condense the pipe, steel, and instruments so that it can be fabricated safely and with high quality control. The entire module is then moved to site and set in place with minimal efforts in the field. Show in your narrative what aspects of the design have the scope for modularization and list the pro-cons of modularizing the design.

## Zero Based Execution (show calculations)

This is your team’s primary opportunity to provide design innovation to the overall concept. This is Fluor’s term for revisiting the early decisions that were made on a project that may have inadvertently driven cost and schedule high. For example, in this case if the pipeline is operating 24 hours, thereby reducing the flow rate condition.

It was also decided that the pump cannot be placed within the first 30,000 feet of the start of the pipeline; if that decision was revisited, how it could positively impact the overall cost of the project. Show calculations for how it impacts the design (wall thickness, pipe diameter, pump size, etc) and how it leads to lower costs.

## Risk Assessment

Your team will be tasked with determining the risks of the project and quantifying them in terms of costs. Several sets of criteria will be given to the teams so that they can brainstorm the issues and provide mitigations against those risks. Now that you are familiar with aspects that entail design, costs of procuring and installing them, identify five risk areas and your plan to mitigate them. List your assumptions

## Innovation Idea

Any idea you have come up during design development that you think would improve safety, ease of construction, or cost/schedule.

1. In fluor.py, lines 99 - 148 [↑](#footnote-ref-1)
2. In fluor.py, lines 171 - 221 [↑](#footnote-ref-2)
3. Details on the computation of the foundation, its mass, interaction with pump and soil bearing pressure, concrete trucks, and cost is found in fluor.py, lines 193 - 306 [↑](#footnote-ref-3)
4. Found in fluor.py [↑](#footnote-ref-4)
5. Found in pipes.py [↑](#footnote-ref-5)