Expected Commute Time Between Classes

Page Count: 18

1 Introduction

At my school, it feels as if we are constantly taking long walks. Following the end of each fifty-minute class period, students have five minutes to commute to their next class. For some, these five minutes can feel like plenty or feel inadequate. This paper will investigate whether the five-minute commute time between classes is well-founded, determining whether it is insufficient or too much time for students to arrive at their next class.

This investigation aims to determine whether a five-minute commute time is justified by the layout and size of my school. In Part 1, we will calculate the expected commute time assuming that traveling from any class to another is equally likely. In Part 2, we will extend the analysis of Part 1 to the temporary 'one-way hallway' policy instituted during the COVID-19 pandemic and determine how much commute times increased (thus quantifying the inefficiency of the commutes). Throughout this investigation, we will assume that every commute from one classroom to another takes the shortest path possible.

This topic interests me as I find myself frequently rushing to my next class. While I find that the five minutes is insufficient, I would like to justify or challenge my opinion with mathematics. Testing the commute times can also be used by the school administration to determine if its current scheduling should be adapted. Furthermore, the topic lends itself to working with various maps and graphs and thus lends itself to lots of visualization, which I enjoy in mathematics.

Figure 1: School Map

2 Part 1: Expected commute time (simple case)

To begin the investigation, I must first clarify the layout of the school and the relevant data. The school has three floors: the main level, lower level, and upper level. Furthermore, some classes are held outside in the school's greenhouse (labeled G on the school map shown above in Figure 1). An important consideration is that due to safety reasons, the school must only be entered and exited through the main entrance (labeled ME). For the sake of clarity, I have labeled also the Auditorium (labeled as A) and my economics classroom (labeled as E). These labeled locations will be referred to many times throughout the exploration for both frequent commutes and also the longest commutes on the school's campus.

The mathematical tool we will be using for this analysis is graphs. A graph is formally recognized as a pair of sets, the first set being the set of the graph's vertices and the second being the set of the graph's edges or pairs of vertices that are connected. Informally, graphs can be seen as representing a mutual relationship between objects, which could be people knowing one another or cities with an airport line between them. [2] Similar to the airport example, this problem lends itself to a graph. By filling the school map with vertices, we can connect vertices with the rule that there is a direct pathway or hallway between the corresponding locations on the school map. While the definition of a graph is abstract, it can

Figure 2: Graph Representing School Map

be made much more concrete with a picture representing the vertices and the edges which connect them. This is shown in Figure 2 above.

The figure represents the graph which can describe commuting in the main level of the school. For each vertex, you may only arrive at that vertex by traveling on an edge. For example, if you would like to travel from vertex 17 to vertex 19, you must walk along the edges which are connected to vertex 18. This graph may be created just by placing vertices at key points (such as the main entrance, the intersections of hallways, the ends of hallways, etc.). However, I have split up each hallway into several vertices to represent classrooms that students may be stopping at. The graph was created using the website tool "Graphonline" and the vertices were best spaced to equally split up the lengths of each hallway shown on the school map. [2]

Using this graphical representation of the main level of the school, we would like to calculate the expected commute time. To do so, we must use the idea of the length of a path. Given any two vertices, a path is a sequence of connected and distinct vertices representing the journey between these two vertices in the graph. Furthermore, the length of a path is the number of edges in it. Say, for example, that we want to travel from vertex 17 to 19. A path connecting the two vertices could be Vertex 17, Vertex 18, and Vertex 19. This path has a length of 2 as there are 2 edges traveled on. Furthermore, the idea that the vertices are distinct becomes more intuitive– if you are walking between any two vertices, it is inefficient to visit the same portion of the hallway twice in the journey, as it is unnecessary.

Besides the edges connecting ME and G (which will be addressed in a later calculation), each of the vertices is equally spaced and thus represents the same distance traveled. By calculating the average length of a path for all vertices on this graph, we can thus calculate the expected time by multiplying by the expected amount of time traveling along one edge. Vertex G and the edge connecting vertex ME to G is a special case– while the commute to the greenhouse is daily for me (as my school attendance is taken there) many students may have never stepped foot in the building. The average commute time would be distorted with the addition of the greenhouse as it must be measured externally and is uncommon for most students. Thus, the calculation of the 'average commute time' will be split into two cases: one without the commute to the greenhouse and one with.

At this stage, we would like to find the average length of a path on our graph representing the main level (excluding the Greenhouse, vertex G). To do this, we can compute the sum of all lengths of all possible paths and then divide it by the number of total paths. We will start by identifying the denominator. From the figure, it can be seen that there is exactly one path connecting any two vertices. Furthermore, there are 36 vertices and 35 edges (since vertex G and the edge connecting it to vertex ME are removed for the time being). This means that we just need to find a way to pick two different vertices to determine the path that connects them. Since the vertices must be different, there will be $\binom{36}{2} = 630$ such ways.

Finding the numerator requires a bit more work. Since we know that every vertex is connected to every other vertex by exactly one path, we can go through each vertex and find its distance from all others (for consistency, the distance from a vertex to itself is 0). The process of this would require $36^2 = 1296$ calculations if to be done by hand. However, there are distance-finding algorithms and computer code that output the 'distance matrix' for a given graph. While the algorithms themselves are out of the scope of this exploration, we can discuss the importance of the distance matrix (which is created for me using the software on Graphonline).

The distance matrix provides all of the minimum distances from one vertex to another. Given a graph with n vertices, if we label the vertices $1, 2, \ldots, n$, the distance matrix of the graph shows the distance from a vertex $i \in \{1, 2, \ldots, n\}$ to $j \in \{1, 2, \ldots, n\}$ in the ith row and jth column of the matrix. The matrix is essentially a way to organize all of the minimum distances of vertices on a graph. Notice that for the sake of our graph, since there is exactly one path between each vertex the minimum distance is simply the length of the path from vertex i to vertex j (which we know exists, as all vertices are connected by some path). For clarity, the vertices labeled G, ME, A, and E will be numbered 1, 2, 15, and 37 respectively (to be found as rows or columns in the Matrix). However, for the calculation we aim to perform, the whereabouts of specific vertices will not matter.

The distance matrix given by Graphonline [2] is shown below. Notice that the matrix's entries are symmetric along the main diagonal (or the diagonal from the top left to the bottom right where the row and column number are the same). This is because the path going to and from a vertex is the same. Notice that the main diagonal only has entries of 0, as the distance from a vertex to itself was defined to be 0. For clarity, since vertex G (or as stated earlier, vertex 1) will not be included in this calculation, every vertex's index as indicated on the graph will be shifted down by 1 for the rows and columns of this matrix.

$$\begin{pmatrix}
0 & 1 & 2 & 3 & 4 & 5 & 6 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 11 & 12 & 13 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 \\
1 & 0 & 1 & 2 & 3 & 4 & 5 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 10 & 11 & 12 & 12 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\
2 & 1 & 0 & 1 & 2 & 3 & 4 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 9 & 10 & 11 & 11 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 \\
3 & 2 & 1 & 0 & 1 & 2 & 3 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 8 & 9 & 10 & 10 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 \\
4 & 3 & 2 & 1 & 0 & 1 & 2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 9 & 10 & 11 & 11 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 \\
5 & 4 & 3 & 2 & 1 & 0 & 1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 10 & 11 & 12 & 12 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\
6 & 5 & 4 & 3 & 2 & 1 & 0 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 11 & 12 & 13 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 \\
6 & 5 & 4 & 3 & 2 & 1 & 2 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 11 & 12 & 13 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 \\
7 & 6 & 5 & 4 & 3 & 2 & 3 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 12 & 13 & 14 & 14 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 \\
8 & 7 & 6 & 5 & 4 & 3 & 4 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 13 & 14 & 15 & 15 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 \\
9 & 8 & 7 & 6 & 5 & 4 & 5 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 14 & 15 & 16 & 16 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 \\
10 & 9 & 8 & 7 & 6 & 5 & 6 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 15 & 16 & 17 & 17 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 \\
11 & 10 & 9 & 8 & 7 & 6 & 7 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 16 & 17 & 18 & 18 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 \\
12 & 11 & 10 & 9 & 8 & 7 & 8 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 17 & 18 & 19 & 19 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 \\
4 & 3 & 2 & 1 & 2 & 3 & 4 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 7 & 8 & 9 & 9 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 \\
5 & 4 & 3 & 2 & 3 & 4 & 5 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 6 & 7 & 8 & 8 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\
6 & 5 & 4 & 3 & 4 & 5 & 6 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 5 & 6 & 7 & 7 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\
7 & 6 & 5 & 4 & 5 & 6 & 7 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 4 & 5 & 6 & 6 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 \\
8 & 7 & 6 & 5 & 6 & 7 & 8 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 3 & 4 & 5 & 5 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\
9 & 8 & 7 & 6 & 7 & 8 & 9 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 2 & 3 & 4 & 4 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
10 & 9 & 8 & 7 & 8 & 9 & 10 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 1 & 2 & 3 & 3 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 \\
11 & 10 & 9 & 8 & 9 & 10 & 11 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 2 & 4 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
12 & 11 & 10 & 9 & 10 & 11 & 12 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 3 & 4 & 1 & 5 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\
11 & 10 & 9 & 8 & 9 & 10 & 11 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 2 & 3 & 0 & 1 & 4 & 2 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
12 & 11 & 10 & 9 & 10 & 11 & 12 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 3 & 4 & 1 & 0 & 5 & 1 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
13 & 12 & 11 & 10 & 11 & 12 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 4 & 5 & 0 & 6 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 \\
13 & 12 & 11 & 10 & 11 & 12 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 4 & 5 & 2 & 1 & 6 & 0 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
13 & 12 & 11 & 10 & 11 & 12 & 13 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 4 & 5 & 2 & 1 & 6 & 2 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
14 & 13 & 12 & 11 & 12 & 13 & 14 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 10 & 9 & 8 & 7 & 6 & 5 & 4 & 5 & 6 & 3 & 2 & 7 & 3 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
15 & 14 & 13 & 12 & 13 & 14 & 15 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 11 & 10 & 9 & 8 & 7 & 6 & 5 & 6 & 7 & 4 & 3 & 8 & 4 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 \\
16 & 15 & 14 & 13 & 14 & 15 & 16 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 12 & 11 & 10 & 9 & 8 & 7 & 6 & 7 & 8 & 5 & 4 & 9 & 5 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 4 & 5 \\
17 & 16 & 15 & 14 & 15 & 16 & 17 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 13 & 12 & 11 & 10 & 9 & 8 & 7 & 8 & 9 & 6 & 5 & 10 & 6 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 3 & 4 \\
18 & 17 & 16 & 15 & 16 & 17 & 18 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 14 & 13 & 12 & 11 & 10 & 9 & 8 & 9 & 10 & 7 & 6 & 11 & 7 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 2 & 3 \\
19 & 18 & 17 & 16 & 17 & 18 & 19 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 15 & 14 & 13 & 12 & 11 & 10 & 9 & 10 & 11 & 8 & 7 & 12 & 8 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 1 & 2 \\
20 & 19 & 18 & 17 & 18 & 19 & 20 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 16 & 15 & 14 & 13 & 12 & 11 & 10 & 11 & 12 & 9 & 8 & 13 & 9 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & 1 \\
21 & 20 & 19 & 18 & 19 & 20 & 21 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 17 & 16 & 15 & 14 & 13 & 12 & 11 & 12 & 13 & 10 & 9 & 14 & 10 & 8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0
\end{pmatrix}$$

Now that we have the distance between each vertex, we would like to compute the sum of the lengths of all paths. This will be the sum of all of the matrix elements divided by 2 (as we are over counting paths by considering whether traveling to or from a vertex). Summing (using the Python code attached in the appendix) yields 5976. Thus, our average length will

(Path from ME to E) Walk Number	Time (minutes:seconds)
1	2:55.34
2	2:53.61
3	2:51.44
Average	**2:53.46**

Table 1: Time taken for walks from the main entrance to economics class

be $\frac{5976}{630} \approx 9.486$. Now, we must use actual data from my walking speed to use this number to find the average commute time between classes.

Ideally, I would have recorded the commute time for all 5976 possible routes to take in the school. Instead, I chose to record my commute time from Vertex ME (the main entrance) to Vertex E (my economics class). This is one of the longest possible walks within the school on the main level and thus means I gain the most accurate picture of the varying traffic conditions across the different hallways. Furthermore, taking the average of multiple of the same walk means that it is most resistant to outliers. The times that I recorded for the walk were 2:55.34, 2:53.61, 2:51.44 minutes. The sum of these three times is 8:48.39 for a total of 520.39 seconds. Thus, the average is 173.46 seconds or 2:53.46 minutes. This information is consolidated above in Table 1.

This time was split among 21 equally spaced edges. This means my average time per edge is $173.46/21 \approx 8.26$ seconds. Now, we must multiply our average edges traveled in each path by our average time per edge to yield the average commute time. Thus,

$$\frac{5976 \text{ total edges in all paths}}{630 \text{ paths}} \times 8.26 \frac{\text{seconds}}{\text{path}} \approx 78.352 \text{ seconds/path}$$

Converting to minutes, the average commute is only 1 minute and 18.35 seconds. This is significantly less than the 5 minutes allotted. Furthermore, the maximum distance traveled according to the distance matrix was 27 edges from Vertex E to Vertex A. This means the maximum travel time of $27 \times 8.26 = 223.02$ seconds = 3 minutes and 43.02 seconds.

(Path from ME to G) Walk Number	Time (minutes:seconds)
1	1:49.30
2	1:50.18
3	1:51.53
Average	**1:50.34**

Table 2: Time taken for walks from the main entrance to the greenhouse

I believe the low average may be due to the fact that most paths included in our calculation are close to one another. However, in reality classes in the same department are often bunched near one another, meaning you are much less likely to travel only one edge, which may be biasing the calculation to be smaller. Even still, the maximum travel time still leaves about a minute to spare.

To expand our model, we will now include the possibility of commuting to the greenhouse, or vertex G. After walking the commute from the greenhouse (Vertex G) to the main entrance (Vertex ME), the times I recorded were 1:49.30, 1:50.18, and 1:51.53, giving an average time of 1:50.34 or 110.34. This information is consolidated in Table 2.

At this point, we must adapt our methodology, since the edge traveling from Vertex ME to Vertex G has a different time-weighting than all other edges. Instead, we can use a clever workaround to calculate the new average time and longest commute. Notice that every commute starting from Vertex G must go through Vertex ME (as Vertex ME is the only connection to Vertex G). We can take the the first row of the distance matrix, representing all of the commute times from Vertex ME, convert them from number of edges into average times, and then add on our calculated average time walking from the main entrance to the greenhouse. Furthermore, this tells us that there are exactly 36 new paths after reintroducing vertex G. This data will be communicated in a table, as shown on the page below.

Path from Vertex ME to Vertex _	# Edges	Avg Time (in s)	Avg time from Vertex G to Vertex _ (in s)
ME	0	0.00	110.34
3	1	8.26	118.60
4	2	16.52	126.86
5	3	24.78	135.12
6	4	33.04	143.38
7	5	41.30	151.64
8	6	49.56	159.90
9	6	49.56	159.90
10	7	57.82	168.16
11	8	66.08	176.42
12	9	74.34	184.68
13	10	82.60	192.94
14	11	90.86	201.20
A	12	99.12	209.46
16	4	33.04	143.38
17	5	41.30	151.64
18	6	49.56	159.90
19	7	57.82	168.16
20	8	66.08	176.42
21	9	74.34	184.68
22	10	82.60	192.94
23	11	90.86	201.20
24	12	99.12	209.46
25	11	90.86	201.20
26	12	99.12	209.46
27	13	107.38	217.72
28	13	107.38	217.72
29	13	107.38	217.72
30	14	115.64	225.98
31	15	123.90	234.24
32	16	132.16	242.50
33	17	140.42	250.76
34	18	148.68	259.02
35	19	156.94	267.28
36	20	165.20	275.54
E	21	173.46	283.80

Important Quantities (in s)
Sum 6929.32
Average 192.48
Max 283.80

Table 3: Calculations now including the Greenhouse

The first column on the left Table (in Table 3) labels which path we are concerned with, with the second column stating how many edges each path has from the first row of the distance matrix. Essentially, the first column orders which path we are concerned with (i.e the path from Vertex ME to all of the other vertices). We then multiply the number of edges by 8.26 seconds, our average time for traveling each of the equally spaced edges. With this average time quantity, we add 110.34 seconds to indicate the additional time traveling from Vertex G to Vertex 1 and then to our desired vertex. From the table on the right, we get important quantities like the sum, average, and maximum time for commutes from Vertex G to all other vertices of our graph. In fact, the maximum time stated in the table is the commute from Vertex G to Vertex 36, which is a commute that I make every day. With an expected time of 283.30 seconds or 4 minutes and and 43.30 seconds, its clear why I feel like 5 minutes is never enough.

Using the important quantities stated in Table 3, we can now make broader calculations about the average commute time now including the greenhouse. The calculation will be

$$\frac{(5976 \times 8.26) + 6929.32 \text{ sec}}{630 + 36 \text{ paths}} = 84.521 \text{ seconds/path}$$

The logic from this calculation comes from thinking of the average as the total amount of time spent on commutes over the number of commutes. Since our total time increases by 6929.32 seconds and our number of commutes increases by 36, our new average is 84.521 seconds per path or 1 minute and 24.521 seconds. Comparing this to our previous average, including the greenhouse increases our average by 84.521 - 78.352 = 6.169 seconds per path. Based on this result, I began to conjecture that the long walks had 'biased me' into thinking that all walks were longer. However, this thought evolved into the belief that there might be a large deviation from the mean.

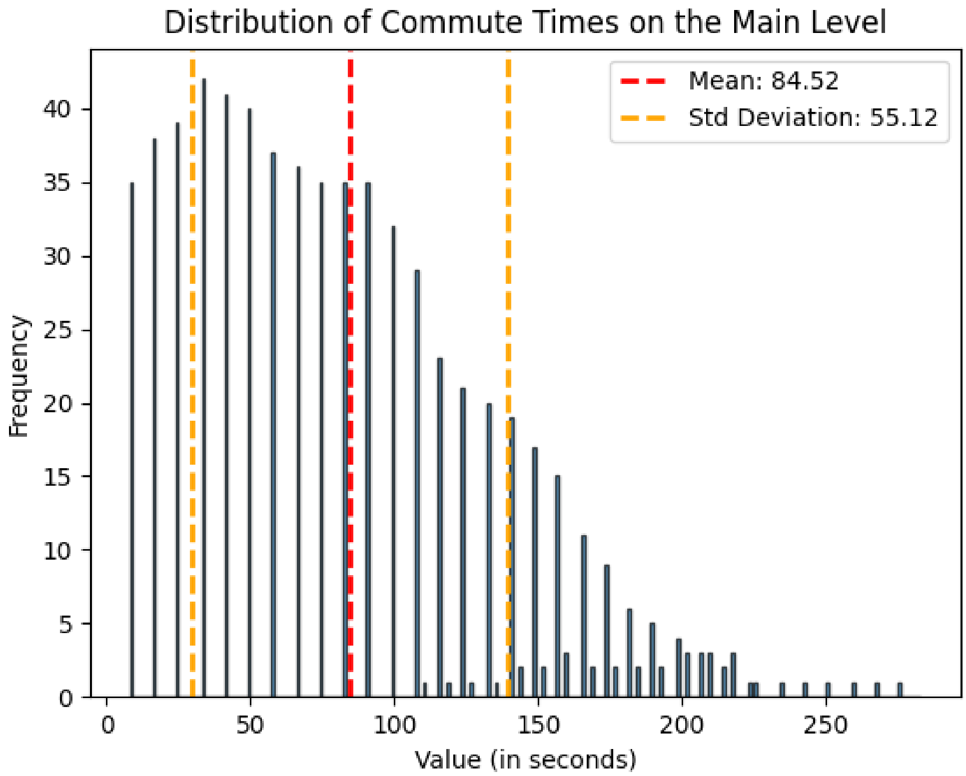


Figure 3: Distribution of Commute Times on the Main Level

To better understand the distribution of all commute times, I created a Python program to plot the frequency of each of the commute times (the bins being 1 second wide) and highlighted the mean and standard deviation.

Figure 3 finally puts the mean number into perspective. The commute times have a large standard deviation of 55.12. Simply put, the long walks are quite long and the short walks are quite short in comparison to the mean commute time. Furthermore, the impact of the greenhouse becomes much clearer with the visual– very few additional walks take place, but these walks populate the rightmost part of the graph (i.e the longest walks).

Figure 4: 2021 School Map (with One-Way Hallways)

3 Part 2: One-way hallways

In 2021, the school opened with restrictions in place to reduce the spread of COVID-19. The idea to reduce hallway traffic and close contact with large groups of students was to institute one way hallways, where students could more easily spread themselves six feet apart. Furthermore, during this time two openings were operated: the main entrance (labeled as ME in the map shown in Figure 4) and also the east entrance (labeled A). The idea was that students would enter the school in the morning from the side entrance and move across the red direction as shown in the map. Students could access the blue direction by going up or down the stairs, traveling along the second floor or basement respectively to travel along the blue direction. Furthermore, since the greenhouse was outside and clustering would not be an issue, the greenhouse could be accessed from both the main and east entrance and both directions of movement were permissible.

To model this mathematically, we must use more advanced tool than a conventional graph. This is because you can only travel from node to node in a certain direction– commutes are no longer symmetric, or the same there and back. This created the familiar scenario (to the annoyance of many students and teachers) where the classroom next to them would be a far longer walk to obey the one-way hallway rules.

The tool that we can use are weighted directed graphs. A directed graph is a pair of sets, the first set being the set of the graph's vertices and the second being the set of the graph's *directed* edges or *ordered* pairs of vertices that are connected. This is an important distinction– for a graph with two vertices v and w, (v, w) and (w, v) are two different edges. Visually, the first edge is an arrow pointing from v to w and the second is pointing w to v. In the case of our graph, this means that having a directed edge means that you can only travel along that direction. This will be made clearer when the picture of the graph is shown later in the exploration. Furthermore, a *weighted* graph is a graph where a 'weight' (some number, positive or negative) is assigned to each edge. In the previous section, we could think of each edge having weight one, meaning that when we travel across a path our distance matrix gave us the sum of all of the weights (or analogously, the amount of edges that we travel on). [1]

This problem intuitively lends itself to a directed graph, as we must distinguish the direction we are able to travel along for each vertex. However, the weights may not be as clear. The important simplification we are making is that a student is *only* attempting to travel to classes along the main level. While my school has a second floor and basement floor, the approach of the problem would become too large for an exploration if the details were matched perfectly. Thus, we assume for this section that the second and basement floors are used *only* for traveling backwards to access classrooms on the main level. However, to travel along the blue direction, you must go up or down a stairwell which only exists at the end at the hallways and requires a considerable amount of time to traverse. Thus, we would like it so that traveling along the blue direction from one stairwell back to a previous stairwell has an additional 'weight' (representing the additional amount of time spent using the stairs) than simply traversing the hallway in the red direction.

Another useful aspect is that now we do not need to treat the cases of including the

Figure 5: Directed Weighted Graph of the School

greenhouse and without the greenhouse separately. To reflect the greater amount of time spent commuting to the greenhouse, we can simply assign the edges going to and from the greenhouse a higher weight according to the increased commute time. It is also important to notice that with the addition of one-way hallways, our assumption that a student would always be traveling along the single ‘best’ path breaks down. This is because with the addition of the blue direction, students may have multiple paths with different total weights. Thus, in this section we will be considering the *best* path they can take, as there are multiple paths which do not visit vertices multiple times but could incur additional weight.

The new directed weighted graph shown in Figure 5 to model the scenario is shown above. Now, between every adjacent vertex that is indoors and on the main level, you can now see a directed edge with weight one. Notably, there are now large arrows going in the opposite (referred to as the blue direction previously) with various weights. The weights were calculated from my time of traveling on the stairs, which for simplicity I have rounded to a weight of one. Thus, to go backwards in a hallway, you must incur a weight of one going up the stairs, incur all of the weights of traveling backwards, and incur an additional weight going down the stairs (essentially, the 2 more than the number of edges between the two 'stairwell' vertices). Furthermore, the edge between Vertex ME and Vertex G as well as between Vertex A and Vertex G both do not have a direction and respectively have a weight of 12 and 3. based on the previously collected data Though the weights would be most accurate by including decimal places (for example, instead of having a weight of 12 we could have a weight of 110.34/8.26), including decimal places will make blur the analysis by unnecessarily opting for accuracy over a clearer understanding of the relative length of each commute.

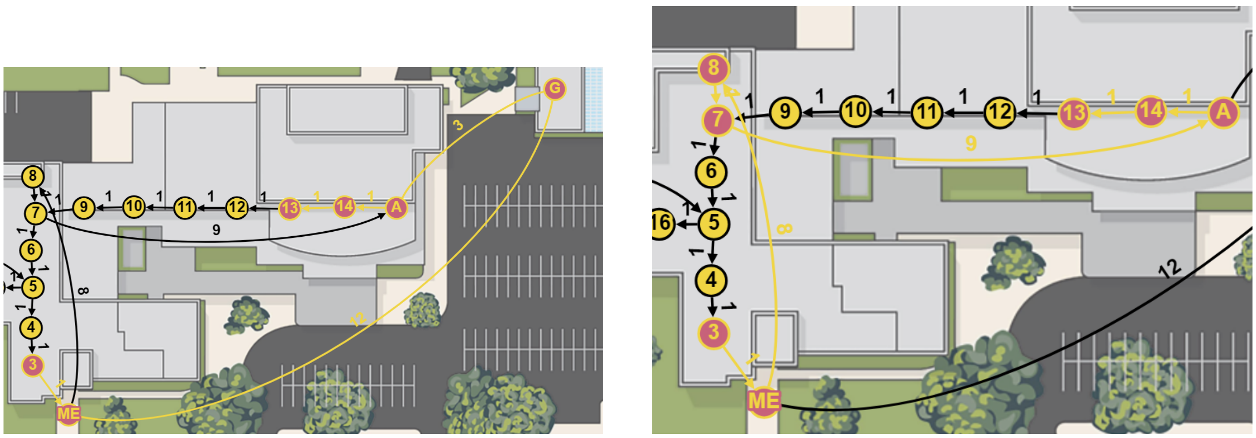


Figure 6: Two possible paths from Vertex 3 to Vertex 13

Furthermore, having access to the visual representation of the weighted directed graph now shows us the existence of multiple paths between two vertices, as shown in Figure 6. Take for example the two paths between Vertex 3 and Vertex 13. The first path(shown on the left) is $3 \to ME \to G \to A \to 14 \to 13$, incurring a total length of 18. The second path (shown on the right) is $3 \to ME \to 8 \to 7 \to A \to 14 \to 13$, incurring a length of 21. Thus, it is unintuitively faster to leave the building to go to another location within the building. This is one of many examples of how the one-way hallway policy changed student's intuitions about going from class to class.

Thankfully, the distance algorithm provided by GraphOnline only uses the minimum path distance. Thus, assuming the student is making optimal choices while still following the rule, here is the following distance matrix:

$$
\begin{pmatrix}
0 & 12 & 14 & 12 & 11 & 10 & 20 & 9 & 8 & 7 & 6 & 3 & 13 & 14 & 15 & 16 & 17 & 18 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 & 22 & 20 & 19 & 32 & 31 & 30 & 13 & 5 & 4 \\
12 & 0 & 13 & 11 & 10 & 9 & 8 & 21 & 20 & 19 & 18 & 15 & 12 & 13 & 14 & 15 & 16 & 17 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 21 & 19 & 18 & 31 & 30 & 29 & 12 & 17 & 16 \\
13 & 1 & 0 & 12 & 11 & 10 & 9 & 22 & 21 & 20 & 19 & 16 & 13 & 14 & 15 & 16 & 17 & 18 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 & 22 & 20 & 19 & 32 & 31 & 30 & 13 & 18 & 17 \\
15 & 3 & 2 & 0 & 13 & 12 & 11 & 24 & 23 & 22 & 21 & 18 & 1 & 2 & 3 & 4 & 5 & 6 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 10 & 8 & 7 & 20 & 19 & 18 & 1 & 20 & 19 \\
16 & 4 & 3 & 1 & 0 & 13 & 12 & 25 & 24 & 23 & 22 & 19 & 2 & 3 & 4 & 5 & 6 & 7 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 11 & 9 & 8 & 21 & 20 & 19 & 2 & 21 & 20 \\
12 & 5 & 4 & 2 & 1 & 0 & 13 & 15 & 14 & 13 & 12 & 9 & 3 & 4 & 5 & 6 & 7 & 8 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 12 & 10 & 9 & 22 & 21 & 20 & 3 & 11 & 10 \\
13 & 6 & 5 & 3 & 2 & 1 & 0 & 16 & 15 & 14 & 13 & 10 & 4 & 5 & 6 & 7 & 8 & 9 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 13 & 11 & 10 & 23 & 22 & 21 & 4 & 12 & 11 \\
13 & 6 & 5 & 3 & 2 & 1 & 14 & 0 & 15 & 14 & 13 & 10 & 4 & 5 & 6 & 7 & 8 & 9 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 13 & 11 & 10 & 23 & 22 & 21 & 4 & 12 & 11 \\
14 & 7 & 6 & 4 & 3 & 2 & 15 & 1 & 0 & 15 & 14 & 11 & 5 & 6 & 7 & 8 & 9 & 10 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 14 & 12 & 11 & 24 & 23 & 22 & 5 & 13 & 12 \\
15 & 8 & 7 & 5 & 4 & 3 & 16 & 2 & 1 & 0 & 15 & 12 & 6 & 7 & 8 & 9 & 10 & 11 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 15 & 13 & 12 & 25 & 24 & 23 & 6 & 14 & 13 \\
6 & 9 & 8 & 6 & 5 & 4 & 17 & 3 & 2 & 1 & 0 & 3 & 7 & 8 & 9 & 10 & 11 & 12 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 16 & 14 & 13 & 26 & 25 & 24 & 7 & 1 & 2 \\
3 & 12 & 11 & 9 & 8 & 7 & 20 & 6 & 5 & 4 & 3 & 0 & 10 & 11 & 12 & 13 & 14 & 15 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 27 & 19 & 17 & 16 & 29 & 28 & 27 & 10 & 2 & 1 \\
30 & 18 & 17 & 15 & 28 & 27 & 26 & 39 & 38 & 37 & 36 & 33 & 0 & 1 & 2 & 3 & 4 & 5 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 9 & 7 & 6 & 19 & 18 & 17 & 16 & 35 & 34 \\
29 & 17 & 16 & 14 & 27 & 26 & 25 & 38 & 37 & 36 & 35 & 32 & 15 & 0 & 1 & 2 & 3 & 4 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 8 & 6 & 5 & 18 & 17 & 16 & 15 & 34 & 33 \\
28 & 16 & 15 & 13 & 26 & 25 & 24 & 37 & 36 & 35 & 34 & 31 & 14 & 15 & 0 & 1 & 2 & 3 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 7 & 5 & 4 & 17 & 16 & 15 & 14 & 33 & 32 \\
27 & 15 & 14 & 12 & 25 & 24 & 23 & 36 & 35 & 34 & 33 & 30 & 13 & 14 & 15 & 0 & 1 & 2 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 6 & 4 & 3 & 16 & 15 & 14 & 13 & 32 & 31 \\
26 & 14 & 13 & 11 & 24 & 23 & 22 & 35 & 34 & 33 & 32 & 29 & 12 & 13 & 14 & 15 & 0 & 1 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 5 & 3 & 2 & 15 & 14 & 13 & 12 & 31 & 30 \\
25 & 13 & 12 & 10 & 23 & 22 & 21 & 34 & 33 & 32 & 31 & 28 & 11 & 12 & 13 & 14 & 15 & 0 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 4 & 2 & 1 & 14 & 13 & 12 & 11 & 30 & 29 \\
36 & 24 & 23 & 21 & 34 & 33 & 32 & 45 & 44 & 43 & 42 & 39 & 22 & 23 & 24 & 25 & 26 & 27 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 1 & 13 & 12 & 11 & 10 & 9 & 22 & 41 & 40 \\
47 & 35 & 34 & 32 & 45 & 44 & 43 & 56 & 55 & 54 & 53 & 50 & 33 & 34 & 35 & 36 & 37 & 38 & 11 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 12 & 24 & 23 & 22 & 21 & 20 & 33 & 52 & 51 \\
46 & 34 & 33 & 31 & 44 & 43 & 42 & 55 & 54 & 53 & 52 & 49 & 32 & 33 & 34 & 35 & 36 & 37 & 10 & 11 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 11 & 23 & 22 & 21 & 20 & 19 & 32 & 51 & 50 \\
45 & 33 & 32 & 30 & 43 & 42 & 41 & 54 & 53 & 52 & 51 & 48 & 31 & 32 & 33 & 34 & 35 & 36 & 9 & 10 & 11 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 10 & 22 & 21 & 20 & 19 & 18 & 31 & 50 & 49 \\
44 & 32 & 31 & 29 & 42 & 41 & 40 & 53 & 52 & 51 & 50 & 47 & 30 & 31 & 32 & 33 & 34 & 35 & 8 & 9 & 10 & 11 & 0 & 1 & 2 & 3 & 4 & 5 & 9 & 21 & 20 & 19 & 18 & 17 & 30 & 49 & 48 \\
43 & 31 & 30 & 28 & 41 & 40 & 39 & 52 & 51 & 50 & 49 & 46 & 29 & 30 & 31 & 32 & 33 & 34 & 7 & 8 & 9 & 10 & 11 & 0 & 1 & 2 & 3 & 4 & 8 & 20 & 19 & 18 & 17 & 16 & 29 & 48 & 47 \\
52 & 40 & 39 & 37 & 50 & 49 & 48 & 61 & 60 & 59 & 58 & 55 & 38 & 39 & 40 & 41 & 42 & 43 & 16 & 17 & 18 & 19 & 20 & 9 & 0 & 1 & 2 & 3 & 17 & 29 & 28 & 27 & 26 & 25 & 38 & 57 & 56 \\
51 & 39 & 38 & 36 & 49 & 48 & 47 & 60 & 59 & 58 & 57 & 54 & 37 & 38 & 39 & 40 & 41 & 42 & 15 & 16 & 17 & 18 & 19 & 8 & 9 & 0 & 1 & 2 & 16 & 28 & 27 & 26 & 25 & 24 & 37 & 56 & 55 \\
50 & 38 & 37 & 35 & 48 & 47 & 46 & 59 & 58 & 57 & 56 & 53 & 36 & 37 & 38 & 39 & 40 & 41 & 14 & 15 & 16 & 17 & 18 & 7 & 8 & 9 & 0 & 1 & 15 & 27 & 26 & 25 & 24 & 23 & 36 & 55 & 54 \\
49 & 37 & 36 & 34 & 47 & 46 & 45 & 58 & 57 & 56 & 55 & 52 & 35 & 36 & 37 & 38 & 39 & 40 & 13 & 14 & 15 & 16 & 17 & 6 & 7 & 8 & 9 & 0 & 14 & 26 & 25 & 24 & 23 & 22 & 35 & 54 & 53 \\
35 & 23 & 22 & 20 & 33 & 32 & 31 & 44 & 43 & 42 & 41 & 38 & 21 & 22 & 23 & 24 & 25 & 26 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 0 & 12 & 11 & 10 & 9 & 8 & 21 & 40 & 39 \\
37 & 25 & 24 & 22 & 35 & 34 & 33 & 46 & 45 & 44 & 43 & 40 & 23 & 24 & 25 & 26 & 27 & 28 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 2 & 0 & 13 & 12 & 11 & 10 & 23 & 42 & 41 \\
24 & 12 & 11 & 9 & 22 & 21 & 20 & 33 & 32 & 31 & 30 & 27 & 10 & 11 & 12 & 13 & 14 & 15 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 3 & 1 & 0 & 13 & 12 & 11 & 10 & 29 & 28 \\
25 & 13 & 12 & 10 & 23 & 22 & 21 & 34 & 33 & 32 & 31 & 28 & 11 & 12 & 13 & 14 & 15 & 16 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 4 & 2 & 1 & 0 & 13 & 12 & 11 & 30 & 29 \\
26 & 14 & 13 & 11 & 24 & 23 & 22 & 35 & 34 & 33 & 32 & 29 & 12 & 13 & 14 & 15 & 16 & 17 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 5 & 3 & 2 & 1 & 0 & 13 & 12 & 31 & 30 \\
27 & 15 & 14 & 12 & 25 & 24 & 23 & 36 & 35 & 34 & 33 & 30 & 13 & 14 & 15 & 16 & 17 & 18 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 6 & 4 & 3 & 2 & 1 & 0 & 13 & 32 & 31 \\
14 & 2 & 1 & 13 & 12 & 11 & 10 & 23 & 22 & 21 & 20 & 17 & 14 & 15 & 16 & 17 & 18 & 19 & 22 & 23 & 24 & 25 & 26 & 27 & 28 & 29 & 30 & 31 & 23 & 21 & 20 & 33 & 32 & 31 & 0 & 19 & 18 \\
5 & 10 & 9 & 7 & 6 & 5 & 18 & 4 & 3 & 2 & 1 & 2 & 8 & 9 & 10 & 11 & 12 & 13 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 17 & 15 & 14 & 27 & 26 & 25 & 8 & 0 & 1 \\
4 & 11 & 10 & 8 & 7 & 6 & 19 & 5 & 4 & 3 & 2 & 1 & 9 & 10 & 11 & 12 & 13 & 14 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 & 26 & 18 & 16 & 15 & 28 & 27 & 26 & 9 & 1 & 0
\end{pmatrix}
$$

Upon inspection, this matrix has much larger quantities due to the weights (which were previously considered as 1). The maximum edge count is 61, and is obtained from walking from vertex 34 to vertex 9 (the actual path will be shown in the Appendix). The path count is now $37^2 - 37 = 1332$, as we are excluding paths from a vertex to itself. The sum of the weights of all paths is 25942, meaning the average weight of a commute is $25942/1332 \approx 19.476$. Scaling these paths by the average time to commute an edge of weight 1, we can list the following data in Table 4.

Table 4: Important Quantities for One-Way Hallways Graph

Max: $61 \times 8.26 = 503.86$ seconds = 8 min 23.86 seconds
Average: $19.476 \times 8.26 \approx 160.87$ seconds = 2 minutes 40.87 seconds

Even though I only experienced the one-way hallways briefly three years ago, seeing these quantities bring back the memories of how difficult it was to navigate the hallways and arrive on class on time. It is clear that the administration has made a conscious effort to ensure every commute is below the 5 minute time allotted without the one-way hallways. Thus, seeing commutes that could total to over 8 minutes shows how much more difficult and inefficient commutes were with this rule. I believe the most interesting result was that for certain commutes, going outside would be the optimal path.

4 Conclusion

4.1 Results

In Part 1 and Part 2, we created a graph model of my school to determine the average and maximum commute times. For both, the average commute time was well under 5 minutes. For part 1, the maximum commute time was still under 5 minutes (albeit barely so), while in part 2 the maximum commute time was well over 8 minutes. Thus, one can argue that in the status quo, 5 minutes is sufficient time between classes. Furthermore, the time allotted should not be reduced, as the maximum commute time would not be included if the time were to be reduced to 4 minutes. For Part 2, arguments could go either way for whether the time between classes should be increased or decreased. However, with the additional context that teachers were more accommodating for students who were lost or confused by the format, I believe that the 5 minutes time was still okay even during the one-way hallway period. Nonetheless, this investigation quantitatively justified the 5 minute commute time by considering all possible commutes for both scenarios.

Conducting this investigation changed my biases about commuting from class to class. While my commutes may feel longer on average due to my frequent commutes to the greenhouse, the average student will feel no such issue while trying to get to class. With this investigation, I have gained a broader perspective of the student body at my school and challenged my preconceived notions.

4.2 Evaluation

There are various assumptions made during the investigation which when removed could facilitate a more accurate picture of commuting times at my school. The most important simplification made is that all commutes would be made on the main floor, while my school has 3 floors. While including the other floors in Part 1 would intuitively increase the average time (from commuting to and on the stairs), this could have an interesting effect on the calculation for Part 2 as there would me more commutes along the direction of the one-way hallway. Furthermore, a more rigorous data collection process, especially at different times per day, could increase the likelihood of the most accurate picture of the foot traffic in the hallways (especially during maximum traffic, where the maximum commute is likely to be increased). Finally, the visuals for the graph could have been more accurate by actually measuring the distance of an edge inside the school, ensuring that the graph precisely lines up with the map of the school.

This investigation could be extended to gain a better understanding of foot traffic in school environments. One extension could be manipulating the configuration of a set amount of vertices and edges and making an argument for the best school layout. Another extension could be to record a more average picture of a students walking speed, as this investigation was conducted only using my average pace. Furthermore, the ideas present here can be linked to 'flow models' present in research involving differential equations. The movement and the traffic flow of students is important to gaining the most accurate picture, and only through more advanced tools like differential equations can the problem be most accurately represented.

5 Bibliography

References

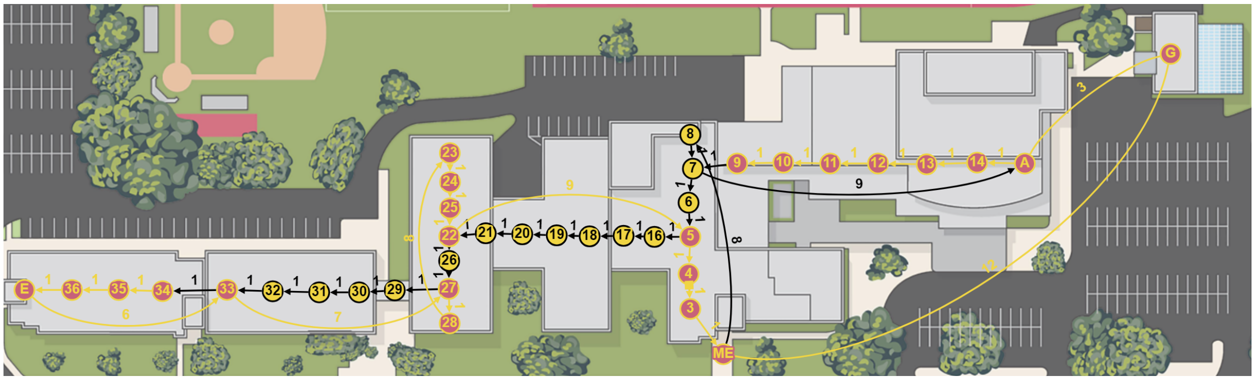
[1] Bumbacea, Radu. "Chapter 1: Introduction and Chapter 9: Directed Graphs." Graphs: An Introduction, XYZ Press, LLC, Plano, TX, 2020.

[2] "Online Graphing Tool." graphonline.ru/en/. Accessed 10 Jan. 2024.

6 Appendix

```python
import matplotlib.pyplot as plt
import numpy as np

distMatrix = [...]
#list is removed for purpose of fitting in appendix

upper_diagonaldistMatrix = [distMatrix[i * 36 + j] for i in range(36) for j in range(i
    + 1, 36)]

greenhousesTimes = [110.34, 118.60, 126.86, 135.12, 143.38, 151.64, 159.90, 159.90,
    168.16, 176.42, 184.68, 192.94, 201.20,
209.46, 143.38, 151.64, 159.90, 168.16, 176.42, 184.68, 192.94, 201.20, 209.46, 201.20,
    209.46, 217.72, 217.72, 217.72, 225.98,
234.24, 242.50, 250.76, 259.02, 267.28, 275.54, 283.80]
average_value = np.mean(upper_diagonaldistMatrix)
std_deviation = np.std(upper_diagonaldistMatrix)

timesNoGreenhouse = result = [x * 8.26 for x in upper_diagonaldistMatrix]

allTimes = timesNoGreenhouse + greenhousesTimes

average_value = np.mean(allTimes)
std_deviation = np.std(allTimes)
plt.hist(allTimes, bins=range(int(min(allTimes)), int(max(allTimes)) + 1),
    edgecolor='black', alpha=0.7)
plt.axvline(average_value, color='red', linestyle='dashed', linewidth=2, label=f'Mean:
    {average_value:.2f}')
plt.axvline(average_value + std_deviation, color='orange', linestyle='dashed',
    linewidth=2, label=f'Std Deviation: {std_deviation:.2f}')
plt.axvline(average_value - std_deviation, color='orange', linestyle='dashed',
    linewidth=2)
plt.title('Distribution of Commute Times on the Main Level')
plt.xlabel('Value (in seconds)')
plt.ylabel('Frequency')
plt.legend()
plt.show()
```

This was the code that I used to collect average value, standard deviation, and plot the distribution of the commute times. Similar techniques were used for the directed and weighted graph data.



This was the longest walk of a weight of 61 edges from Vertex 34 to Vertex 9.