

Cell Phone Tower Algorithm

Phillip Marais

April 2022

1 Introduction

The purpose of this document is to explain my algorithm in a graphical representation. This was requested by **Mr. Anshul Malhotra**, current CEO of WIM Technologies.

2 Description of Program

The program reads in the input of cell phone towers from a text file located in the same directory. It then creates an Arraylist of objects of cell phone towers that acts as the source of information. I then use an algorithm (That I created by myself) to identify where the frequencies should be assigned. After they are assigned the program prints out the arraylist of cell phone towers. The way my algorithm works is to maximise the distance of towers with the same frequency, while trying to minimise the towers with the same frequency.

3 Description of algorithm

- 1-It starts by taking the first unassigned tower in the arraylist. It then calculates the farthest tower (in kilometers) and then it assigns it the same frequencies.
- 2-Afterwards it calculates the closest tower and assigns it and its farthest tower the next frequency. It repeats this part (number of times depend on the number of frequencies left) and assigns those towers the remaining frequencies.
- 3-If there are more unassigned towers go step 1.

4 Graphical Representation

Here the Cell phone towers are represented as blocks. A table is provided to describe the latitude and longitude of each tower. The graphs represented are not entirely up to scale, but do represent a real world example.

Cell Phone Tower Table		
Tower Name	Longitude	Latitude
A	-0.03098	51.53657
B	-0.02554	51.53833
C	-0.02448	51.53721
D	-0.02415	51.5445
E	-0.02204	51.54735
F	-0.02234	51.53328D
G	-0.02206	51.53948
H	-0.01561	51.53862
I	-0.01273	51.54337
J	-0.01272	51.54202
K	-0.01216	51.5407

Let the list of frequencies given be [**110, 111, 112, 113**]. We start the algorithm by finding the first tower in the arraylist with no set frequency. In the example the first tower will be A. Then we calculate it farthest tower which is K and assign them both the same frequency.[Figure 1]

Now it finds the closest towers that do not have a frequency set. The amount of towers we find and set depend on the remaining frequencies in the given list of frequencies. Thus when we use 110 we have 111 - 113 left, so we find 3 of the closest towers and set them to the remaining frequencies.(B - 111, C - 112, G - 133)[Figure 2]

It then finds the farthest tower each of the closest towers and assign them the same frequency.[Figure 3]

If there are still more towers left unassigned in the arraylist it will repeat the process until all towers have been assigned.[Figure 4]

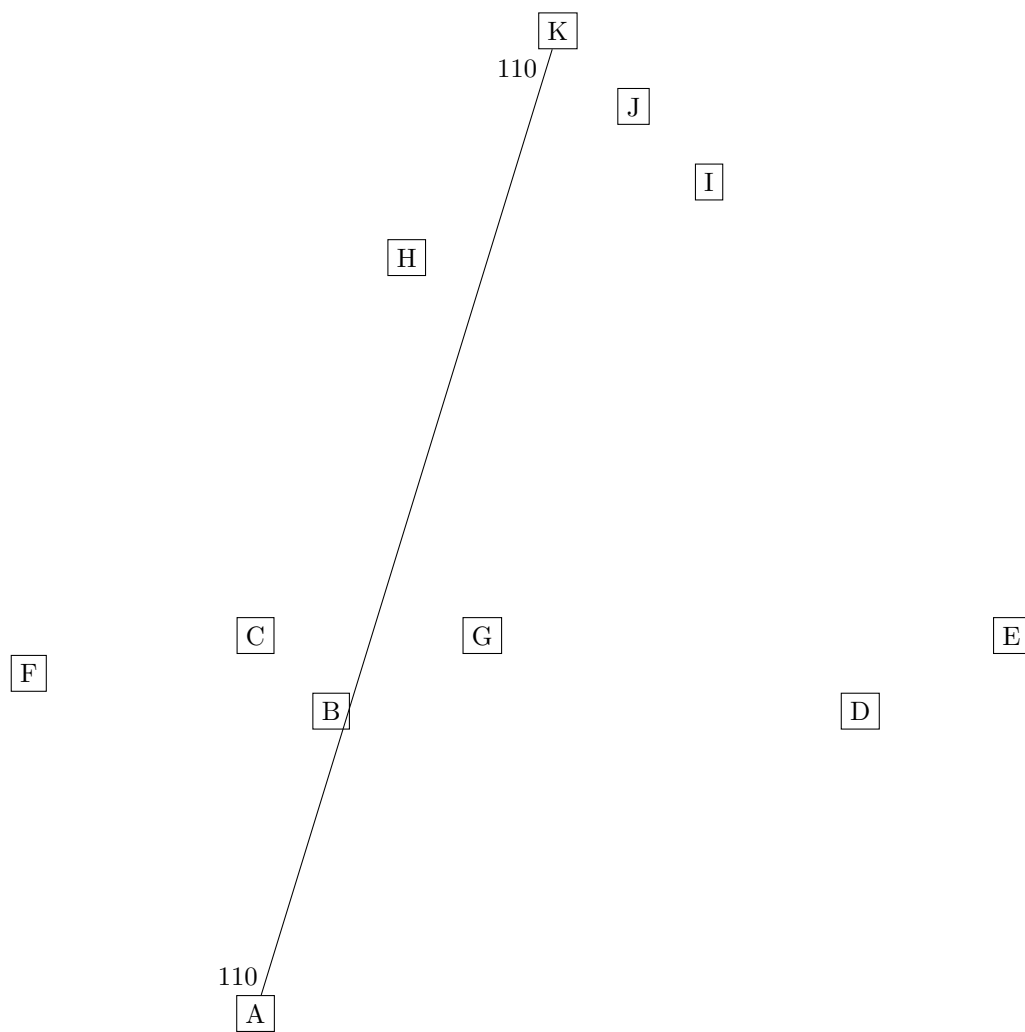


Figure 1

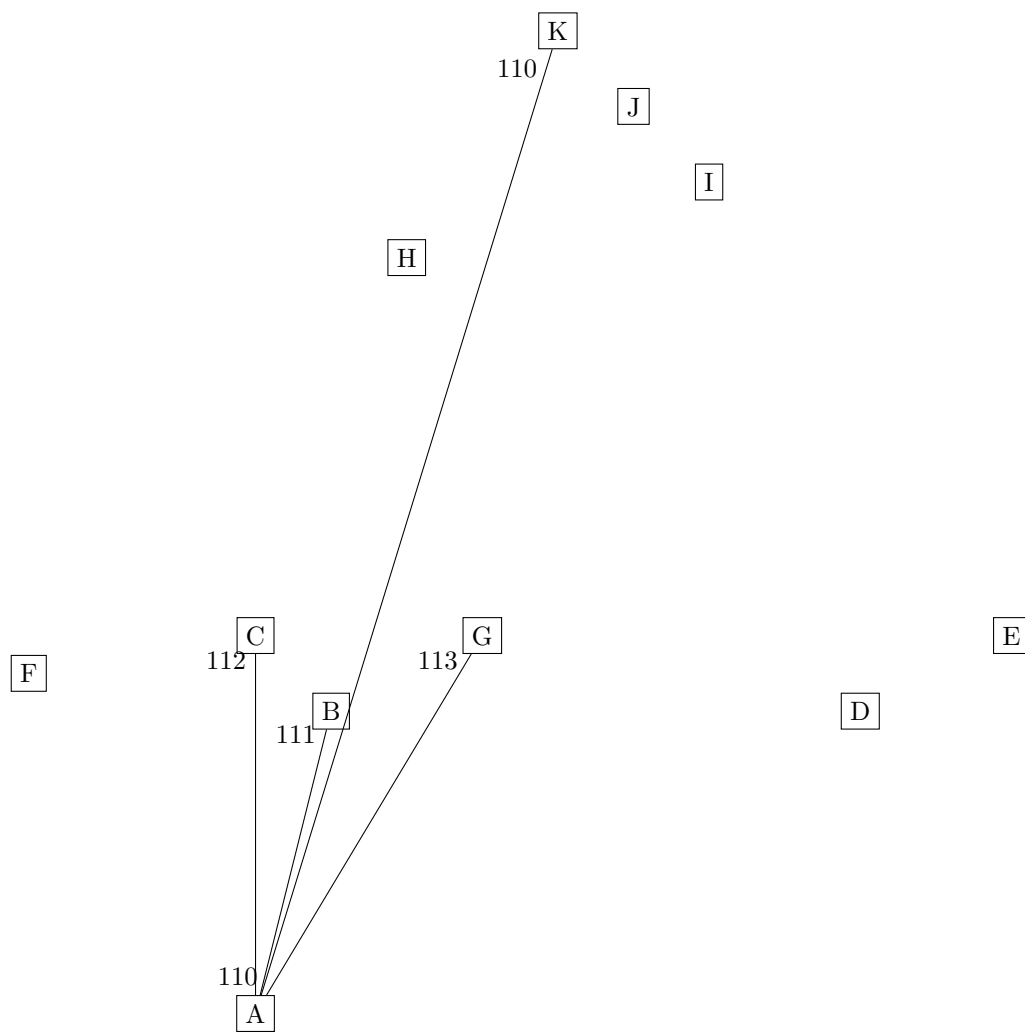


Figure 2

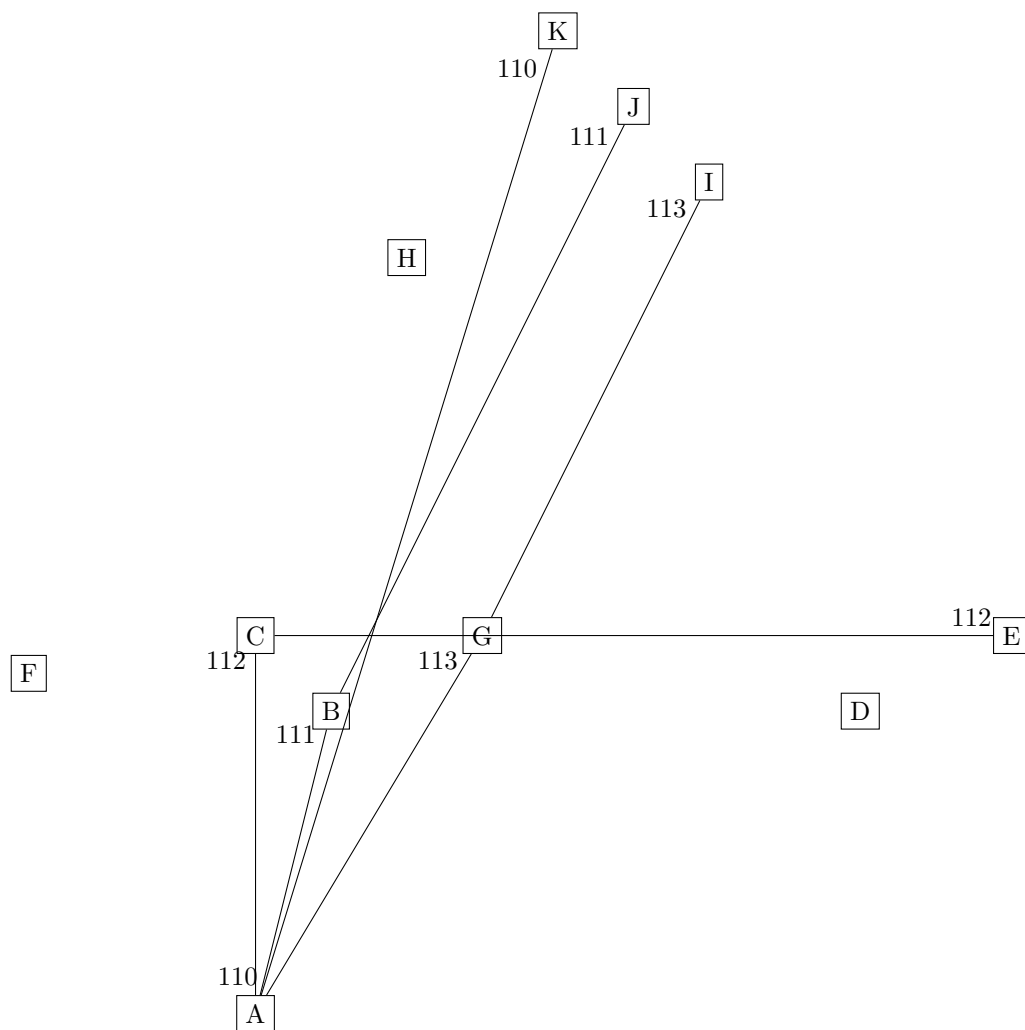


Figure 3

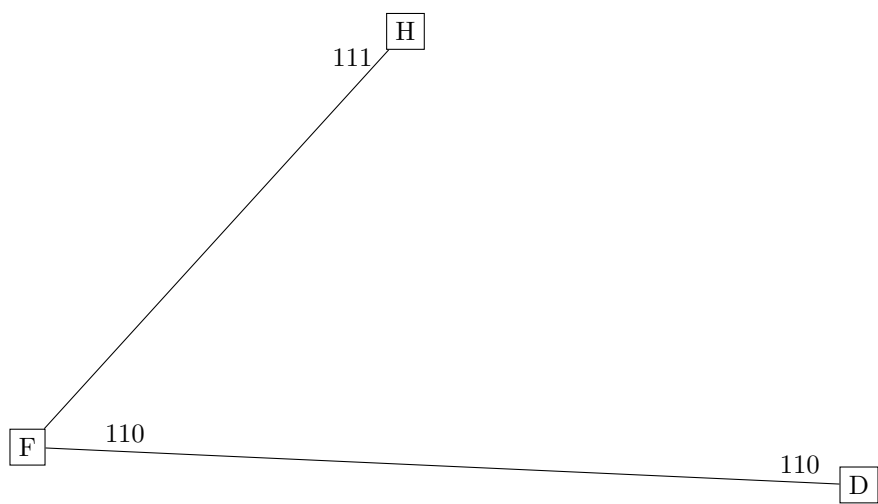


Figure 4