Empirical Competitive Analysis of Online Algorithms With Advice for the Online Facility Location Problem

Jayson Isaiah Tan *, Luis Miguel Senatin **, and Jhoirene B. Clemente ***

University of the Philippines Diliman, M339+H94, Roxas Ave, Diliman, Quezon City, Metro Manila, Philippines,
jttan7@up.edu.ph *, lmsenatin@up.edu.ph **, jbclemente@up.edu.ph ***

Abstract. In this study, the Online Facility Location Problem is examined using historical data on 911 EMS calls in Montgomery County. K-means clustering and the offline solution to a past year's data are both explored as solutions to the online facility location problem and compared to existing online solutions both with and without advice, and an approximation of the offline solution. In the given data set, the proposed solutions were found to outperform the existing solutions, with the most optimistic solution having a competitive ratio of 1.005036. The proposed solutions also show better performance in terms of computation time and memory costs.

Keywords: Facility location problem, online algorithms, empirical analysis

1 Introduction

Traditional algorithms operate with full knowledge of the input data before producing an output. These algorithms are also known as *offline* algorithms. However, such approaches are often impractical in real-world scenarios where complete input information is unavailable. Instead, *online* problems present input data incrementally, with new elements—called *requests* or arriving sequentially. Algorithms designed for online problems must make decisions without access to the entire input[1].

For offline problems, the performance is measured through time and space complexity, but for online problems, these cannot be directly applied since requests are tied to time. To evaluate the quality of an online algorithm, the *competitive ratio* is used instead. [1]. This compares the output of the online algorithm with the solution from an *optimal offline algorithm* which knows the entire input and can generate an optimal solution. The closer the approximation of the online algorithm is to the optimal offline solution, the more competitive the algorithm is [1].

Formally, a c-competiticve online algorithm is defined as follows.

J. Caro et al. (eds.), *Proceedings of the Workshop on Computation: Theory and Practice (WCTP 2025)*, Atlantis Highlights in Computer Sciences 24,
https://doi.org/10.2991/978-94-6239-638-8_22

Definition 1 (c-competitive Online Algorithms [4]). *An online algorithm ALG is c-competitive if there is a constant a such that for all finite input sequences I, $ALG(I) \leq c * OPT(I) + a$, ALG attains a competitive ratio c.*

This means that for each input I, a c-competitive algorithm will incur a cost within a factor c of the offline optimal cost, up to the additive constant a. It should be noted that the competitive ratio is always at least 1, and the smaller it is, the better ALG performs with respect to OPT, the optimal algorithm that produces the minimum cost for each feasible input and output[4].

The Facility Location Problem takes as input a collection of demand points, a cost function that defines the expense of serving a demand point from a facility, and the cost associated with opening a facility. The objective is to determine a set of facilities such that each demand point is assigned to exactly one facility. The goal is to minimize the combined cost of establishing the facilities and the service costs for meeting the demands. Formally, the problem is defined as follows:

Definition 2 (Facility Location Problem).
Input: *Set of Demands $U = \{u_1, u_2, \dots u_n\}$, Facility opening cost c, Service cost function $cost(u, f)$*
Output: *Set of Facilities $F = \{f_1, f_2, \dots, f_k\}$*
Goal: *Minimize total cost of serving all demands and opening facilities, $totalcost = c * k + \sum_{t=1}^{n} cost(u_t, f_t)$ where f_t is the closest facility to u_t.*

The Facility Location Problem is known to be NP-hard [7], which implies that no polynomial-time algorithm is currently known to solve it optimally unless P = NP. This complexity arises from the need to balance two conflicting objectives: minimizing the cost of opening facilities and minimizing the cost of assigning demand points to these facilities. Finding the optimal combination of facility locations and assignments requires exploring an exponential number of possible configurations, making the problem computationally intractable for large instances. Consequently, approximation algorithms and heuristics are often employed to achieve near-optimal solutions efficiently.

The online variant of FLP (OFLP) was introduced by Meyerson[13]. In the online variant, the inputs, also referred to as demands, arrive one at a time and are serviced in order of their arrival. For their proposal, both uniform and non-uniform facility costs and both a randomized and adversarial way of receiving requests were considered. Facilities and requests are in a metric space where the cost of serving requests is the distance from a request to the facility it is assigned to. Their algorithm is randomized in choosing whether to open a new facility or use an existing facility.

We will focus on the simplest variant of the problem where we expect a uniform cost of opening a facility, service cost will be based on a distance function in the metric space, the facilities are incapacitated, meaning that facilities can service an unlimited amount of demands. We also assume an oblivious adversary feeding input to the algorithm, meaning demands do not arrive in such a way that would maximize the cost of the objective function. Based on the survey paper by Markarian [12], the current best algorithms have competitive ratios that are functions of the input size. This means that the error grows alongside input size, and this error can still be mitigated. The results are summarized in Table 1.

When it comes to real-world applications of the OFLP, background information and context is usually available to help solve these kinds of problems. For example: in disaster response for typhoons, data is available on where the disaster may take place, the population density in the area, intensity of the disaster, etc.. Knowing this, the introduction of advice on OFLP can improve the performance of algorithms and provide more meaningful insights on practical solutions to OFLPs in real-world problems.

This study empirically evaluates a number of online algorithms, including ones with advice for the online facility location problem using real-time 911 call datasets. First, Meyerson's algorithm will be compared to the existing human-made solution as seen in the dataset, an online algorithm with advice, and a 1.52-approximation algorithm for the offline solution to the facility location problem. This will show how impactful introducing advice may be to an online algorithm. Existing online algorithms with advice are also compared against each other, as well as to two new online algorithms made for this study. In comparing online algorithms with advice, we not only look at how close the solution gets to the offline solution, but also the efficiency and practicality of generating the advice, since the goal is for these solutions to be applied to real-world scenarios.

2 Related Work

Empirical competitive analysis was demonstrated in the paper of Kasilag et al.[10], but on a different problem. In their study, the effect of the prediction error on the competitive ratio of their online algorithm is analyzed. It was shown that as ML prediction error increases, solution quality decreases; the magnitude of error is directly proportional to the size of the input; and for some tolerable errors, the solution quality is improved by their proposed algorithm, but when exceeding the tolerance, using the best existing online algorithm has a higher solution quality[10].

A variation of the OFLP is studied by Almanza et al.[2], where some information on the problem is known in advance. For their study, advice came in the form of multiple sets of potential facility locations to open, referred to as S. By predicting the optimal facility locations, the output set of facilities to open will be closer to the solution produced by an optimal offline algorithm. Their algorithms exhibit good theoretical guarantees as well as good practical performance[2]. Three algorithms are outlined in the paper. For the purposes of this research, the TAKEHEED algorithm of Almanza et al. is noted by the researchers. This is an online randomized algorithm whose expected cost is $O(log\ (|S|) * OPT(S))$, where $OPT(S)$ is the best solution that can be obtained using only facilities in S. For this study, their TAKEHEED algorithm is evaluated both with offline advice, meaning reliable and correct advice coming from the offline solution for the inputted demands, and random advice, meaning advice that has the potential to be incorrect. Almanza et al. also used the 1.52 approximation algorithm for offline facility location by Mahdian et al. [11], referred to as THEBASELINE in their work. The Max Planck Institute implementation is used [9], and will similarly be adapted for this study. Additionally, the amount of data evaluated at a single moment is notable since while the overall size of the dataset is large and spans around a year, the input instance consists of the activities for only one day.

Markarian [12] made a survey of recent findings related to the online facility location problem. Included in this survey is a summary of the performances of different algorithms and their classifications. Additionally, the fields in which the problem has been applied were noted, such as leasing, clustering and urban planning. Markarian [12] notes that the only other survey related to the online facility location problem is one by Fotakis in 2011, but unlike Fotakis, she aims to "target all variations and online settings for OFL, discuss their scenarios from real-world applications, and shed light on the arising open problems."

The portion of the survey most relevant to the researchers is the analysis of Classical OFL, encompassing the metric and non-metric variations of the problem. Through Markarian's summary of different algorithms [12], a narrative of the work done on the problem is effectively made, illustrating the progression of different algorithms.

The researchers have based the following table illustrating the different algorithms tackling the metric OFLP, their authors, result, and approach on this work. The work of Almanza et al. [2] was also added to this table as it also works on the metric OFLP, but stands out due to utilizing advice. For the following table, n is equal to the number of facilities, and S contains multiple sets of potential facility locations to open.

The different variants of OFLP discussed by Markarian [12] include non-metric OFLP, connected OFL; where facilities need to be connected, OFL with deadlines; where facilities are only opened momentarily and disappear once demands are assigned to them, and OFL with delays; where demands can wait before they are assigned to a facility for a cost. Other variants also include OFL in leasing framework, and OFL in dynamic environments. For the metric setting, extensions of the problem include considering penalties, commodities, and multiple facilities. For the non-metric setting, extensions of the problem include considering service installation costs, service-quality costs, and multiple facilities. A number of open problems were also discussed.

3 Empirical Analysis

This research implements an empirical approach to evaluating the performance of different methods of solving the facility location problem as presented through the 911 calls dataset. The methods tested are as follows:

1. Offline
   - THEBASELINE, a 1.52 approximation of the offline solution [2][11][9]

2. Online without advice

   – Meyerson's Algorithm, with an oblivious adversary [13]

3. Online with advice

   – TAKEHEED with offline advice [2]
   – TAKEHEED with random advice [2]
   – *K-means clustering of previous year as facilities for current year*
   – *OfflinePrev (facilities from the previous year used for the current year)*

The last two items (also in italics), K-means clustering and OfflinePrev, are solutions unique to this study, though the application of all of these methods to the 911 calls dataset has also not been done before.

K-means clustering is an unsupervised machine learning algorithm in which k points called means or cluster centroids are initialized. Then, each item is categorized to its closest mean and the mean's coordinates are updated to be the averages of the items categorized in that cluster so far. This process is repeated for a given number of iterations and at the end, we have our clusters[8]. For this problem, the cluster centroids represent the facilities. For a given facility cost, the optimal number of facilities would have to be known beforehand. K-means clustering is applied in this study by finding the k clusters to be set as facilities from the demands from 2016, as this would be known data. These facilities are then used to service demands that arrive in 2017, with each demand simply being assigned to the nearest facility, with no new facilities being built.

OfflinePrev is an algorithm that utilizes a set of facilities from the offline solution of a previous year and assigns incoming demands to their nearest facility.

For the purposes of this analysis, only the EMS calls from 2017 were evaluated as demands, and only the EMS calls from 2016 were used as advice.

Additionally, a representation of the operating cost of the actual dataset was calculated by finding the sum of all distances between each call and the station that serviced them, and adding the total facility cost. Although the dataset for the EMS stations in Montgomery County listed 60 stations [15], only 49 stations were found throughout the dataset, and only 33 were utilized in 2017. As other solutions only build facilities when they will be used, the total facility cost was calculated with 33 facilities.

All scenarios and solutions were tested with facility costs from 500 to 5000 in intervals of 500. Meyerson's algorithm was run 10 times, with its results being averaged out to get a more consistent idea of its performance. Each run of K-means clustering was done with 200 iterations, with $k = 11, 26, 33$ to more accurately compare to different solutions as facility cost increases. Haversine distance was used as the distance function for this study.

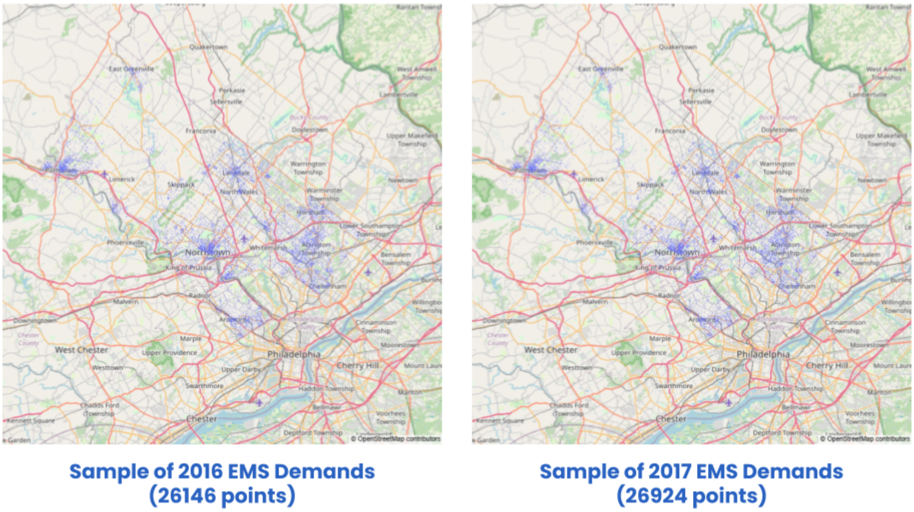

Fig. 1. Side by side of the plot of demands for 2016 and 2017 on the map.

Both the solution utilizing K-means clustering and OfflinePrev are inspired by the observed visual similarity in distribution of EMS calls between 2016 and 2017 as seen in Figure 1. Based on Definition 2, given the same facility opening cost and service cost function, only the difference between two sets of demands will cause a difference in their optimal sets of facilities. Thus, since the distribution of demands in 2016 is observed to be similar to the distribution of demands in 2017, it is hypothesized that the optimal location of facilities in 2016 is similar to the optimal location of facilities in 2017.

3.1 Preprocessing of the dataset

The 911 calls dataset[14] was processed as follows. First, the EMS calls for each year were extracted from the dataset. The coordinates for the locations of the EMS stations within Montgomery County were also obtained from the county data center. [15] From this, the distance between each call and the station that serviced it was found. The sum of distances was also taken as part of the operation cost in order to compare the actual locations of the stations with THEBASELINE, and solution through Meyerson's algorithm. Due to computing constraints, in order to use the implementation for THEBASELINE provided by Almanza et al. [2], a sample equal to 50% of the dataset for both 2016 and 2017 was taken in order to be able to compare all solutions with THEBASELINE.

3.2 Comparison of Costs

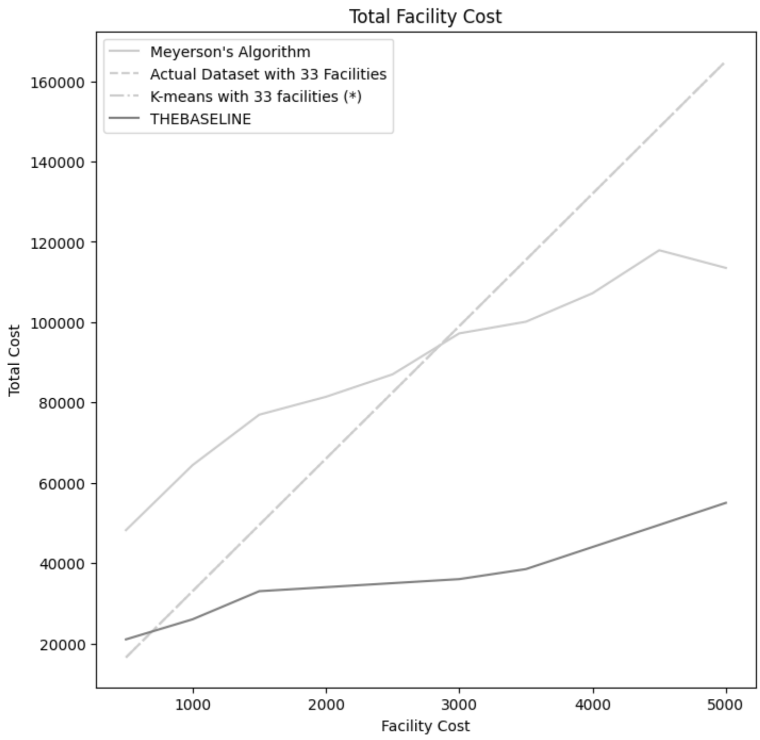

Fig. 2. Line plot showing the total cost of setting up facilities for Meyerson's algorithm, actual dataset with 33 facilities, K-means clustering with 33 facilities, and THEBASELINE

Starting with Figure 2, it can be observed that Meyerson's algorithm has a higher total facility cost compared to both the actual dataset (hereon referred to as Actual) and K-means clustering with 33 facilities (hereon referred to as K-means_33) at 0 to 2500 cost, and lower total facility cost at 3000 facility cost onwards. Actual and K-means_33 occupy the same line as both solutions will always create 33 facilities. Through both Meyerson's algorithm and THEBASELINE, it can be observed that it is more optimal to create less facilities as the facility cost increases, though Meyerson's algorithm creates far more than what is optimal as observed in the much lower total facility cost of THEBASELINE.

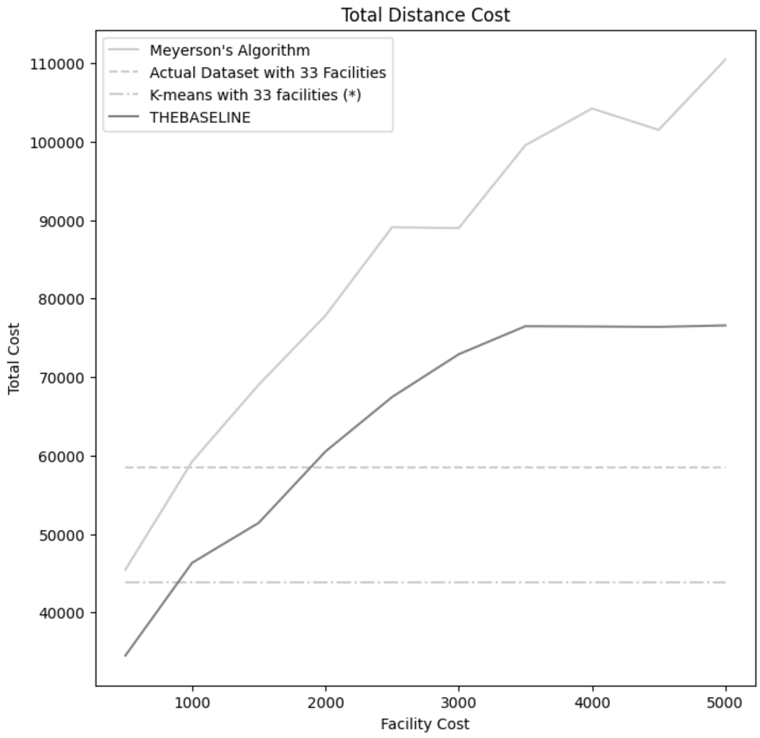

Fig. 3. Line plot showing the total cost of distances for Meyerson's algorithm, actual dataset with 33 facilities, K-means clustering with 33 facilities, and THEBASELINE

While at 3000 facility cost, Meyerson's algorithm creates close to 33 facilities on average, it can be observed in Figure 3 that the total distance cost of Meyerson's algorithm is much higher than both Actual_33 and K-means_33. This shows us that even when creating the same number of facilities, the way that Meyerson's algorithm places facilities is highly inefficient compared to K-means_33 and even the human-made solution reflected in Actual_33 from costs 1000 onwards. Additionally, Meyerson's algorithm seems to follow a similar trend to THEBASELINE, albeit much less efficiently.

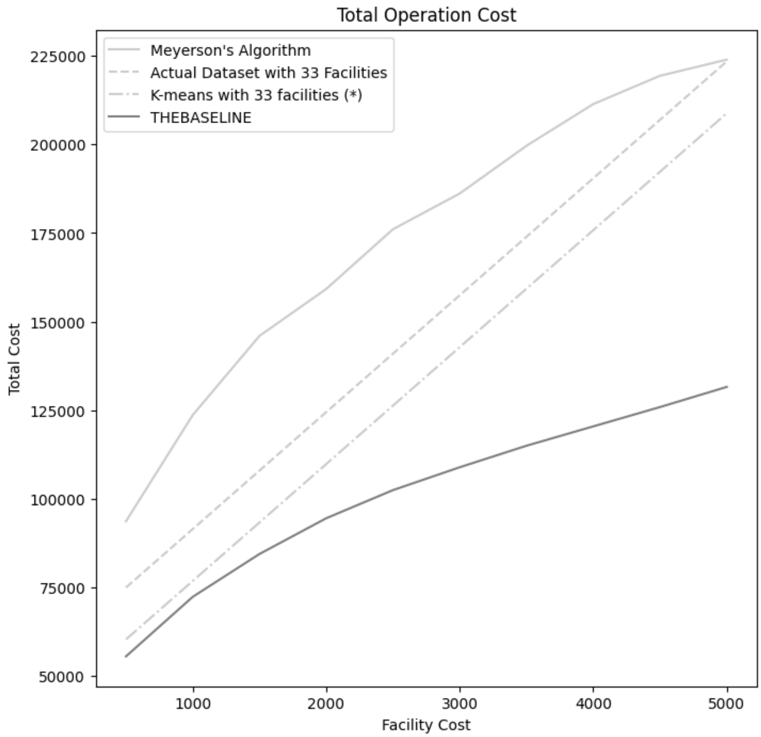

Fig. 4. Line plot showing the total operation costs for Meyerson's algorithm, actual dataset with 33 facilities, K-means clustering with 33 facilities, and THEBASELINE

The culmination of this inefficiency in both deciding the number of facilities to set up and where to place these facilities can be seen in Figure 4, where Meyerson's algorithm consistently has the highest total operation cost, never beating even Actual_33. In Figure 4, it can also be observed that K-means_33 is more efficient compared to Actual_33, with its total operation cost actually being quite close to THEBASELINE at lower facility costs, where it creates a similar number of facilities: 42 facilities at 500 cost and 26 facilities at 1000 cost. This already shows the potential of utilizing K-means clustering as a solution to OFLPs.

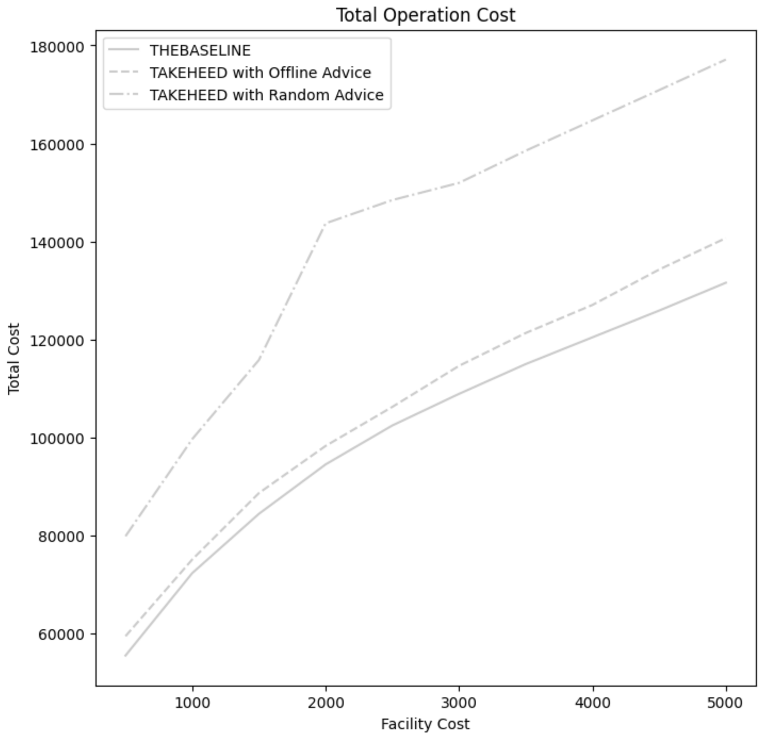

Fig. 5. Line plot showing the total operation costs for TAKEHEED with offline advice, TAKEHEED with random advice, and THEBASELINE

In Figure 5, the total operation cost of the two TAKEHEED variations[2] are compared to THEBASELINE. Here, the lack of robustness to bad advice TAKEHEED has can be observed, which is already noted by Almanza et al. in their study[2]. Additionally, it can be observed that TAKEHEED with offline advice performs very well, able to stay near THEBASELINE, with error increasing slightly as the facility cost increases.

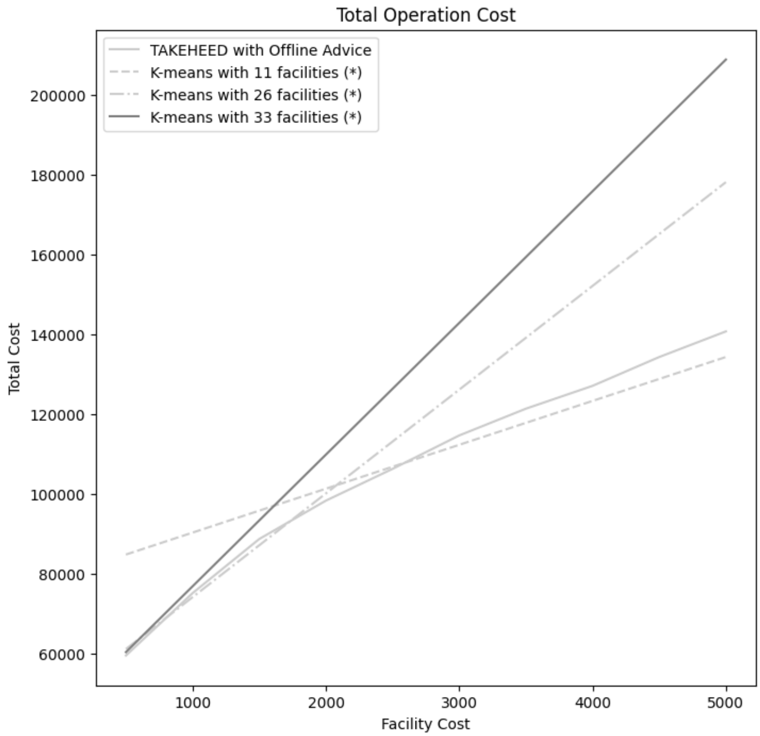

Fig. 6. Line plot showing the total operation costs for TAKEHEED with offline advice and K-means clustering with 11, 26, and 33 facilities

Given the observed efficiency of TAKEHEED with offline advice relative to THEBASELINE, this is then further compared to the K-means clustering solution with 11, 26, and 33 facilities. In Figure 6, it can be observed that there are sections and points of the curve created by TAKEHEED with offline advice that overlap or are near to sections of the K-means clustering solutions. At 1000 facility cost, K-means_26 almost overlaps with TAKEHEED, which is quite notable as TAKEHEED also creates 26 facilities at this facility cost. At 3500 facility cost onwards, it can be observed that K-means_11 even performs better than TAKEHEED, which also creates 11 facilities, meaning that K-means_11 is able to more optimally place its 11 facilities relative to the demands. From here, it can be seen that given a set of previous demands that is close to the set of upcoming demands and an appropriate k relative to the facility cost, a solution utilizing K-means clustering on past data can perform better than TAKEHEED with optimal advice.

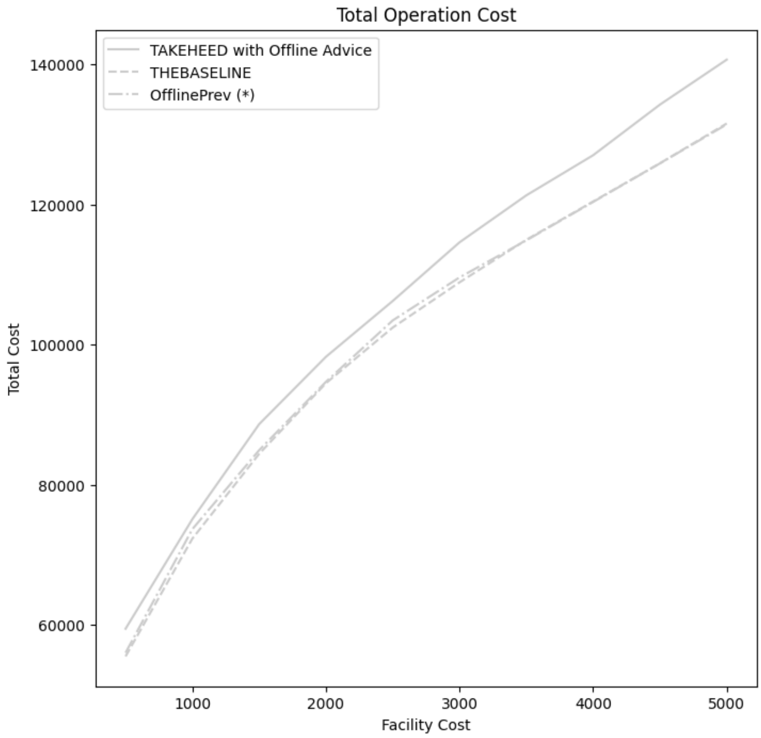

Fig. 7. Line plot showing the total operation costs for TAKEHEED with offline advice, THEBASELINE, and OfflinePrev

Furthermore, the potential of leveraging the patterns in data can be seen most optimistically in Figure 7. It can be observed here that OfflinePrev performs not just better than TAKEHEED with offline advice, but performs very closely to THEBASELINE, even performing marginally better at the highest facility costs. As OfflinePrev is just a solution that uses the offline solution of the previous year as the solution for the current year, the efficiency of this algorithm relies heavily on the distribution of demands between two time periods being similar. Fittingly, it can be observed in Figure 1 that this is the case, as the demands from 2016 and 2017 seem to be almost identical based on the plot.

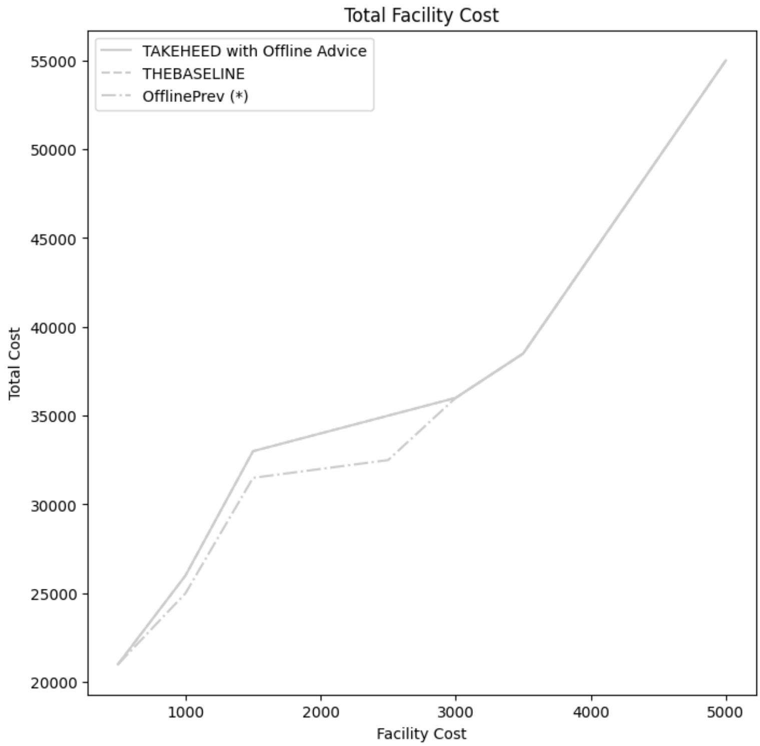

Fig. 8. Line plot showing the total cost of setting up facilities for TAKEHEED with offline advice, THEBASELINE, and OfflinePrev

To further analyze the performance of OfflinePrev relative to the two other algorithms, it can be observed that in Figure 8, that OfflinePrev largely follows the amount of facilities made by THEBASELINE and TAKEHEED, which occupy the same line. OfflinePrev actually sets up slightly less facilities at 1000 to 3000 facility cost, and sets up the same number of facilities afterwards.

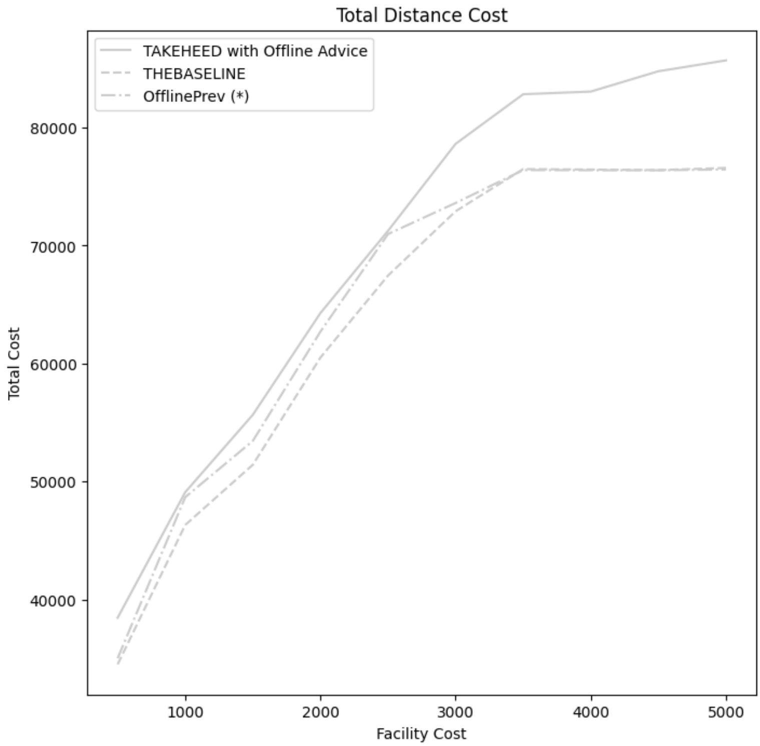

Fig. 9. Line plot showing the total cost of distances for TAKEHEED with offline advice, THEBASELINE, and OfflinePrev

Despite the difference in the number of facilities, it can be observed in Figure 9 that OfflinePrev is still able to consistently reduce the total distance cost compared to TAKEHEED with offline advice, despite setting up less facilities than TAKEHEED at 1000 to 3000 facility cost. This is notable as we would expect that with less facilities, demands would have a larger minimum distance from their nearest facility. It can also be observed that the similar total operation cost between OfflinePrev and THEBASELINE despite having a different number of facilities can be attributed to the higher total distance cost of OfflinePrev being sufficiently offset by the lowered total cost of setting up facilities. Thus, even if OfflinePrev does not follow all of the choices as TAKEHEED does, as seen in Figures 8,9, and 7, OfflinePrev can still solve the objective function in such a way that outperforms TAKEHEED with offline advice.

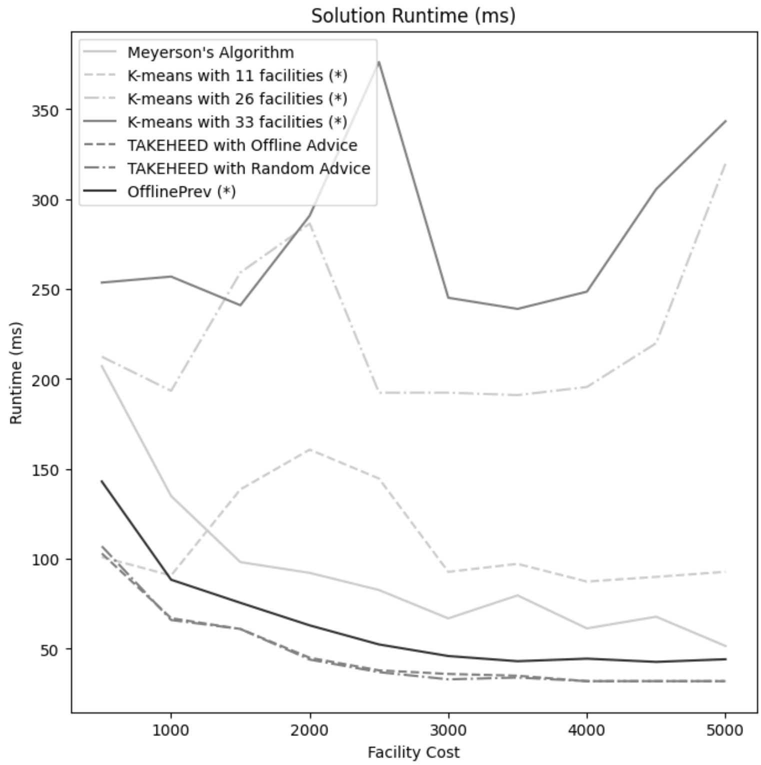

Fig. 10. Line plot showing the solution runtime for Meyerson's algorithm, K-means clustering with 11, 26, and 33 facilities, TAKEHEED with offline advice, TAKEHEED with random advice, and OfflinePrev

In evaluating the runtime of the solutions, it can be observed in Figure 10 that both the TAKEHEED variations have the shortest runtime on average. This is especially notable given the performance of TAKEHEED with offline advice. The K-means clustering solutions have increasingly large and varying runtime due to the inefficient implementation of the function that assigns a demand to its nearest facility. Despite this, it should be noted that the runtime of these solutions is still a fraction of a second.

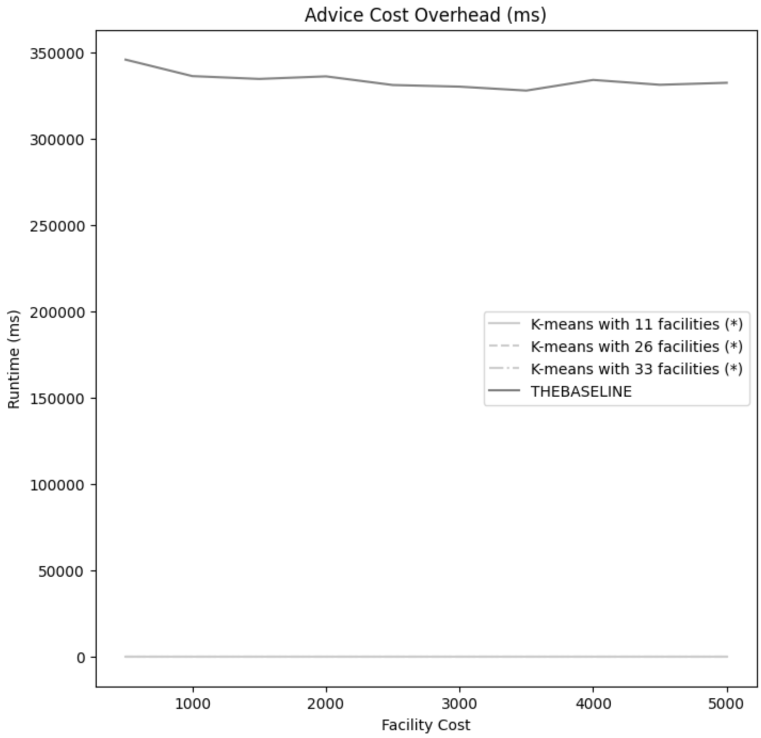

Fig. 11. Line plot showing the runtime for advice generation through THEBASELINE and K-means clustering with 11, 26, and 33 facilities

This is especially important when the time it actually took to generate the advice for the online algorithms that utilized advice is taken into account. In Figure 11, the large disparity between the running time of THEBASELINE versus the time to compute the clusters for K-means clustering can be observed. It takes roughly 350000 times longer to generate the offline solution for the dataset compared to conducting K-means clustering with a reasonable amount of facilities as k.

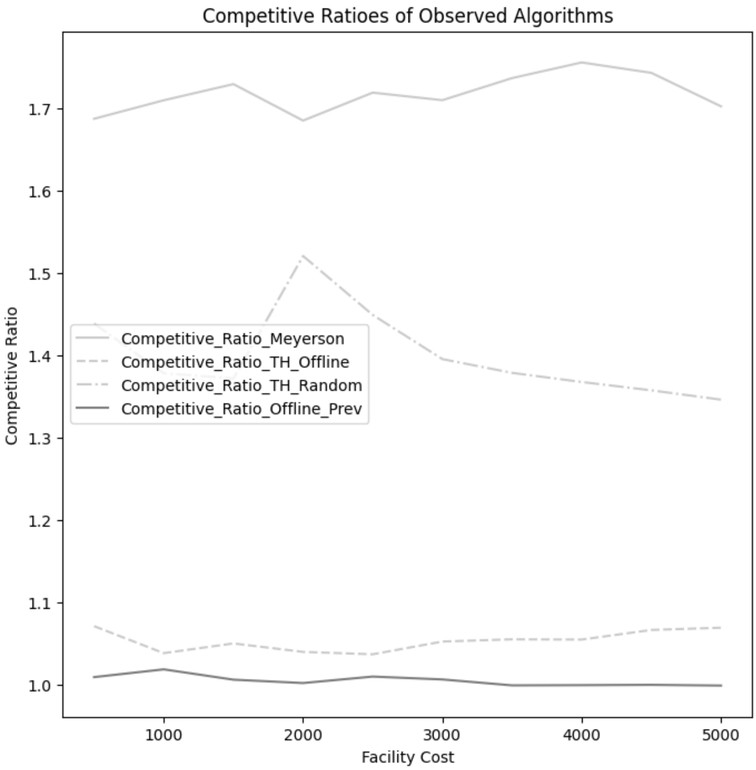

Fig. 12. Line plot showing the competitive ratioes of Meyerson, TAKEHEED with random and offline advice, and OfflinePrev

When the competitive ratios are calculated using THEBASELINE, it can be observed in Figure 12 that OfflinePrev and TAKEHEED with random advice are the two best performers overall, with OfflinePrev being slightly better than TAKEHEED. Meyerson's algorithm and TAKEHEED with random advice have much worse competitive ratios, and lack consistency in performance.

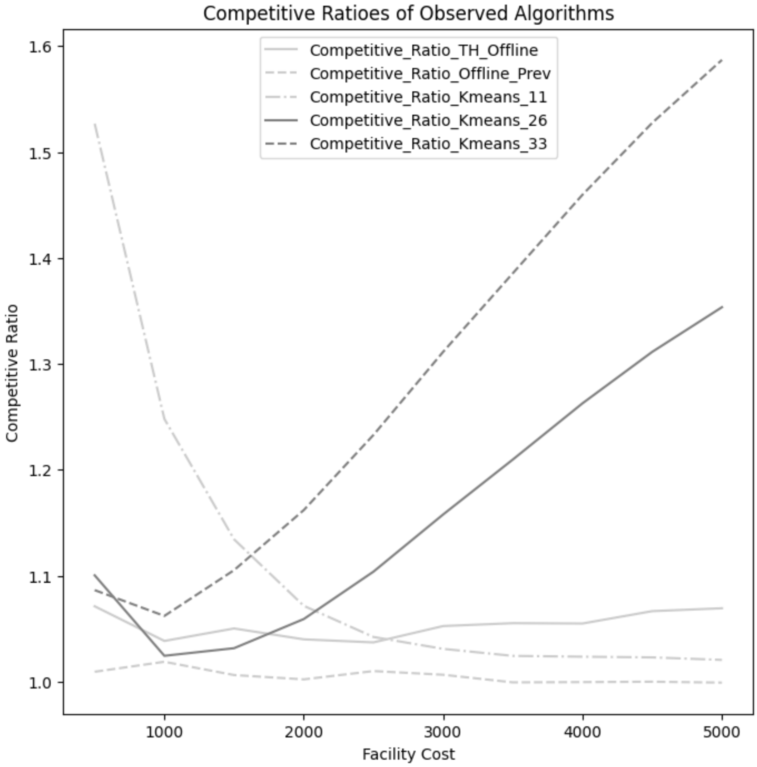

Fig. 13. Line plot showing the competitive ratioes of the k-means clustering with 11, 26, and 33 facilities, TAKEHEED with offline advice, and OfflinePrev

Algorithm/Solution	**Average Competitive Ratio**
Meyerson's Algorithm[13]	1.708781
Actual Dataset with 33 facilities	1.446904
TAKEHEED with offline advice[2]	1.053476
TAKEHEED with random advice[2]	1.400235
OfflinePrev	1.005036
K-means with 11 facilities	1.109851
K-means with 26 facilities	1.170167
K-means with 33 facilities	1.287749

Table 2. Table containing the average competitive ratio of each algorithm

Additionally, when observing the competitive ratios of the K-means clustering solutions and comparing them to the 2 best performing algorithms in Figure 13 , it can be noted that there is a facility cost where the competitive ratio of the

K-means clustering solution is closer to 1 (THEBASELINE) than TAKEHEED with offline advice is. However, all of the K-means clustering solutions perform worse than OfflinePrev.

It can also be observed that the performance of K-means clustering improves as it gets closer to the optimal number of facilities (the optimal k) at the given facility cost. This is seen in how the competitive ratio of K-means with 11 facilities improves as the facility cost increases, while for K-means with 26 and 33 facilities, their lowest point in found at facility cost 1000, with the competitive ratio increasing as facility cost increases.

The average competitive ratio of each algorithm can be seen in Table 2.

3.3 Notable Observations

As previously stated, the experiments conducted used a sample equal to 50% of the EMS calls in 2017. This is still a notably large amount of demands when compared to other studies such as Almanza et al.'s wherein the input instances were much smaller due to them being divided per day[2]. Interestingly, the researchers also found that the implementation used for Meyerson's algorithm was able to handle an input instance using 100% of the EMS calls in 2017, whereas the implementation of the 1.52-approximation solution had a high memory cost, resulting in the input instance needing to be cut down to 50%. The importance of utilizing advice for online algorithms in order to get closer to the offline solution can be seen in how Meyerson's algorithm is unable to place a number of facilities efficiently compared to other solutions placing the same number of facilities. The design in Meyerson's algorithm to only place facilities at demand points may also contribute to this inefficiency. The performance of OfflinePrev is heavily influenced by how close the distribution of demands are between two years. Alongside the fact of how resource-intensive running THEBASELINE is, the application of OfflinePrev may ultimately still be limited. However, if a trend in data is observable, this may still be leveraged without the potentially high resource costs through a solution utilizing K-means clustering, which has shown to have the potential to be more efficient than the existing TAKEHEED-based algorithms provided by Almanza et al[2].

4 Conclusions

In conclusion, the impact of advice in an online solution for the facility location problem is clearly seen. On average, Meyerson performs 1.729 times worse than THEBASELINE. Meyerson's algorithm can also perform worse than the configuration of facilities at the time.

If the data between two years are similar or if the distribution of demands over a certain time frame can be expected to follow a similar pattern or trend, then calculating the offline solution of the previous year may give a very close result to getting the offline solution of the current year. This is seen optimistically with OfflinePrev, where on average, it performs 1.005 times worse than THEBASELINE[2].

If there are existing memory and time constraints that would make the calculation of THEBASELINE[2] or any similar approximation to the offline solution impractical, a solution utilizing k-means clustering may be a sufficient replacement provided an appropriate k can be used for the given facility cost, as seen with its close performance to TAKEHEED with offline advice[2].

As future work, the potential of k-means clustering can be explored by testing for the optimal k for a given facility cost. Additionally, given the observed dependence on the similarity between the data points across 2016 and 2017, a proper metric for the similarity between the distribution of two different sets of demands and the resulting facilities should be pursued in order to properly describe the relationship between the two beyond visual inspection. Lastly, the researchers also recommend that the study be repeated with road network distance in order to see if there will be any changes in the observations made.

References

1. Albers, S. Online algorithms: a survey. *Mathematical Programming.* **97** pp. 3-26 (2003)
2. Almanza, M., Chierichetti, F., Lattanzi, S., Panconesi, A. & Re, G. Online Facility Location with Multiple Advice. *Advances In Neural Information Processing Systems.* **34** pp. 4661-4673 (2021)
3. Alon, N., Awerbuch, B., Azar, Y., Buchbinder, N. & Naor, J. A general approach to online network optimization problems. *ACM Transactions On Algorithms (TALG).* **2**, 640-660 (2006)
4. Borodin, A. & El-Yaniv, R. Online Computation and Competitive Analysis. (Cambridge University Press,1998)
5. Boyar, J., Favrholdt, L., Kudahl, C., Larsen, K. & Mikkelsen, J. Online algorithms with advice: A survey. *ACM Computing Surveys (CSUR).* **50**, 1-34 (2017)
6. Fotakis, D. On the competitive ratio for online facility location. *Algorithmica.* **50**, 1-57 (2008)
7. Garey, M. & Johnson, D. Computers and Intractability: A Guide to the Theory of NP-Completeness. (W. H. Freeman,1979)
8. Kartik "K means Clustering – Introduction". (2025,5), https://www.geeksforgeeks.org/k-means-clustering-introduction/
9. Hoefer, M. Performance of heuristic and approximation algorithms for the uncapacitated facility location problem. (Max-Planck-Institut für Informatik,2002,12)
10. Kasilag, C., Rey, P. & Clemente, J. Empirical Competitive Analysis for the Online Assignment Problem with ML Predictions. *Workshop On Computation: Theory And Practice (WCTP 2023).* pp. 38-54 (2024)
11. Mahdian, M., Ye, Y. & Zhang, J. Improved Approximation Algorithms for Metric Facility Location Problems". *Approximation Algorithms For Combinatorial Optimization.* pp. 229-242
12. Markarian, C. Online Algorithmic Study of Facility Location Problems: A Survey. *IEEE Access.* **12** pp. 77724-77738 (2024)
13. Meyerson, A. Online Facility Location. *Proceedings 42nd IEEE Symposium On Foundations Of Computer Science.* pp. 426-431 (2002)
14. Chirico, M. Emergency - 911 Calls. (Kaggle,2020), https://doi.org/10.34740/kaggle/dsv/1381403

15. Montgomery County. Montgomery County EMS Stations. (ArcGIS,2025), https://services1.arcgis.com/kOChldNuKsox8qZD/arcgis/rest/services/Montgomery_County_EMS_Station/FeatureServer, Last accessed February 7, 2025