2018

Scalable Heuristic for Locating Distribution Centers on Real Road Networks

Saeed Ghanbartehrani
Industrial and Systems Engineering Department, Ohio University, ghanbart@ohio.edu

Jose David Porter
School of Mechanical, Industrial and Manufacturing Engineering, Oregon State University, david.porter@oregonstate.edu

Scalable Heuristic for Locating Distribution Centers on Real Road Networks

Saeed Ghanbartehrani
Industrial and Systems Engineering Department
Ohio University
Athens, Ohio
ghanbart@ohio.edu

J. David Porter
School of Mechanical, Industrial and Manufacturing Engineering
Oregon State University
Corvallis, Oregon
david.porter@oregonstate.edu

***Abstract*—The median problem is a type of network location problem that aims at finding a node with the total minimum demand weighted distance to a set of demand points in a weighted graph. In this research, an algorithm for solving the median problem on real road networks is proposed. The proposed algorithm, referred to as the Multi-Threaded Dijkstra's (MTD) algorithm, is used to locate Walmart distribution centers on the 28-million node road network of the United States with the objective of minimizing the total demand weighted transportation cost. The resulting optimal location configuration of Walmart distribution centers improves the total transportation cost by 46%.**

Keywords—network location, median problem, real road network, clustering, distribution center location

I. Introduction

Location theory is a well-established and active research area. The two main factors in the facility location problems are customers and facilities serving them. The best location for a facility depends on the nature of the problem being studied, the problem's constraints, and the optimality criteria [1].

Determining the location of distribution centers plays a significant role on the efficiency, service quality, and economical sustainability of a distribution network. Many models have been proposed to optimally locate distribution centers. Mathematical programming algorithms, multi-criteria decision-making, heuristics, and simulation are among the most applied solution approaches [2].

In this paper, a scalable heuristic for locating distribution centers on real road networks is proposed. The proposed algorithm is used to locate 78 Walmart distribution centers on the continental United States road network graph (composed of over 28 million nodes) so that the total demand weighted transportation cost between each distribution center and the set of stores it serves is minimized.

II. Literature Review

Network location models have been applied to problems in location theory where a facility must be located on a network composed of nodes (or locations) and edges (or routes) [3]. The median problem (also known as the 1-median problem) is a type of network location problem introduced by Hakimi [4] whose objective is to locate a facility on a network so that the total demand weighted distance between a set of demand points and the facility is minimized. Hakimi [5] also formulated a generalized version of the median problem known as the *p*-median problem for locating p facilities on a network.

Kariv & Hakimi [6] proved that the *p*-median problem is NP-hard (on general graphs) and proposed an algorithm with time complexity $O(n^2p^2)$ for solving the *p*-median problem on tree networks (i.e., a connected graph with no cycles).

Many heuristics have been proposed for solving *p*-median problems. Heuristics based on simulated annealing [7] and genetic algorithm [8] have been tested on Beasley's benchmark [9]. Networks in Beasley's benchmark have between 100 to 900 nodes. Avella et al. [10] proposed an aggregation heuristic and tested it on a benchmark data set with up to 89,000 nodes. Rebreyend et al. [11] experimented with *p*-median problems on simplified real road networks with up to 67,000 nodes. The road network in this study is created based on the real road network of Sweden with 1.5 million nodes while the demand values were assigned based on population data. The effect of the density of road network on the quality of solutions was evaluated in this study, however, runtimes were not reported.

Rebereyend et al. [12] in a more recent study, compared several exact and heuristic *p*-median solution methods on Beasley's and Swedish benchmarks. The Swedish benchmark is a simplified road network with 1,938 nodes created based on the real road network for the country of Sweden which is processed and reduced in several stages. The exact solution method evaluated in this study is based on a Mixed Integer Programming (MIP) model which was solved with CPlex. The exact solution method was capable of solving problems on networks with up to 195 nodes. Heuristic solution methods based on genetic algorithm and simulated annealing solved *p*-median problems with up to 100 facilities in the Swedish benchmark with runtimes over 10 hours.

The literature review conducted in this research indicates that none of the proposed exact algorithms for median problems are scalable to real road networks. All reviewed methods perform pre-processing on the road network data to reduce the size of the network. The pre-processing stage is a timeconsuming process that is specific to each problem. Eliminating the pre-processing stage will not only increase the number of candidate nodes (every node on the road network is a candidate location for the facilities) and therefore improve the quality of the solutions, but also makes it possible to solve different location problems on the same network without having to perform time and compute intensive pre-processing stage for each problem. On the other hand, the most scalable methods identified in the literature on simplified road networks take over 10 hours to solve larger problems.

In this research, a scalable algorithm for the median problem on real road networks with millions of nodes is proposed. The proposed algorithm does not require time and computation intensive pre-processing or simplification of the road network data. The algorithm is used to locate 78 Walmart distribution centers with respect to 3,163 Walmart store locations in the continental United States so that the total demand weighted transportation cost between each distribution center and the set of stores it serves is minimized.

III. Methodology

There are three main phases in the proposed methodology: Data Preparation, Preliminary Evaluation, and Location.

In the data preparation phase, data acquired from different sources are analyzed and processed. Current locations of 78 Walmart distribution centers in the continental US, the locations 3,163 of Walmart stores in the continental US, shapes (geographic boundaries) and population of 3,592 urban areas in the continental US, and the US road network with over 28 million nodes are imported in a spatially enabled PostgreSQL database. In this step, every store is allocated to the closest distribution center using the shortest network distance found by A* algorithm. The demand for each store is also estimated base on the population of the urban area the store is located in.

In the preliminary evaluation step, total demand weighted transportation cost for the whole distribution network based on the shortest network distances and estimated demand values found in the previous step is calculated.

Finally, the optimal location for each of the Walmart distribution centers is found under the two following scenarios: (i) current allocation of stores to their closest distribution center and (ii) clustering stores based on proximity and allocating each cluster to one distribution center. A newly developed algorithm is used to locate the distribution centers on the US road network so that the total demand weighted transportation cost is minimized. The proposed facility algorithm, referred to as the multi-threaded Dijkstra's algorithm, is based on Dijkstra's shortest path algorithm and it can solve very large *p*-median problems on real road networks with millions of nodes in reasonable time.

A. *Multi-Threaded Dijkstra's Algorithm*

The multi-threaded Dijkstra's (MTD) is a graph search algorithm based on the bidirectional Dijkstra's shortest path algorithm. The MTD algorithm starts the search from all demand points and finds the node on the network that has the lowest total demand weighted distance to the demand points.

Given a road network graph G with k demand points (v_i), edge weights (c_{ij}), and demand w_i associated with the demand point v_i, the MTD algorithm finds the node in G with the minimum total weighted distance to all demand points. The steps taken by the MTD algorithm are as follows:

1. Set the distance property for all v_i nodes to $d_i(v_i) = 0$. The distance property for all other nodes v is set to $d_i(v) = \infty$.
2. Start from node v_i and add v_i to the open list i. The open list is a priority queue. Do this step for $i = 1$ to k.
3. Select the non-empty open list that contains the lowest top element. The top element of an open list is the node with the lowest d_i value. Expand the top element of the selected open list by calculating $d_i(v)$ for all unvisited adjacent nodes v using Equation (1). Mark the expanded node as visited from demand point i.

$$d_i(v) = d_i(s) + w_i \cdot c_{sv} \tag{1}$$

4. Add all unvisited nodes v adjacent to the expanded node and their corresponding $d_i(v)$ values to the same open list as children of the expanded node. If the same node already exists in the open list, keep the instance with the lower d_i property.
5. Remove the expanded node from the open list. In case all d_i properties of a node v are found, set the value of the objective function for the best solution found so far (i.e., μ) to $\sum_{i=1}^{k} d_i(v)$. Update μ when a lower value is found.
6. Repeat steps 3 through 5 until μ becomes less than all d_i values for the top elements in all open lists (referred to as the optimality criterion), or all open lists become empty. The optimality criterion is presented in Equation (2) in which d_i^{top} is the top element in the open list i.

$$\mu \leq \operatorname*{Min}_{i=1,\dots,k} d_i^{top} \tag{2}$$

Once the optimality criterion is satisfied, the MTD algorithm terminates the search which usually results in searching a fraction of the graph rather than a complete exhaustive search. This behavior improves the efficiency of the algorithm and runtime. The optimality criterion also guarantees that a better solution cannot occur in future iterations of the algorithm – if the search continues – which ensures the optimality of the solution found.

B. *Data Preparation*

The main data utilized in this study included Walmart store and distribution center openings from 1962 to 2006 [14]. The list of Walmart distribution centers and stores was created using several data sources including Walmart's website, Walmart's Environmental Protection Agency (EPA) reports, and Walmart's annual reports [15].

The store openings data includes the store opening date and the street address of 3,163 Walmart stores in the continental United States. The street addresses were converted to latitude and longitude coordinates using an online geocoding service [16]. The data for the distribution centers include the street address and the coordinates for 78 Walmart distribution centers in the continental US. A shape file containing the continental US national urban areas was downloaded from the US Census Bureau website [17].

The road network data for the continental US was retrieved from OpenStreetMap (OSM) [18] project. The OSM data for the continental US is a large 110 GB file.

The locations of all Walmart stores and distribution centers, as well as the urban areas shape file, were imported into the open source, relational spatial database PostgreSQL [20]. PostgreSQL offers a spatial extension known as PostGIS for spatial analysis [21]. The population of the urban area associated with each Walmart store was found using a query developed in PostGIS. In order to analyze the road network data in the PostgreSQL database, the open source routing library pgRouting [19] was installed on the PostgreSQL/PostGIS database management system. PgRouting library adds the most popular shortest path algorithms such as Johnson's, Floyd-Warshall, A*, and Dijkstra to PostgreSQL. To import the road network data in the database, a freeware named OSM2PO [20] was used.

The demand for each store was estimated proportional to the population of the urban area the store is located in. In case there were several stores in the same urban area, the demand was distributed evenly between all stores in that area.

$$Store\ demand = \frac{Total\ urban\ area\ population}{1000\ \times No.of\ stores\ in\ the\ urban\ area} \qquad (3)$$

Another important step in the data preparation phase was the allocation of Walmart stores to their closest distribution centers based on the network distance. Fig. 1 shows the allocation of all 3,163 Walmart stores to their closest distribution center on the map. Shortest network distance was used in this step to achieve a realistic evaluation of current total transportation cost in the preliminary evaluation phase.

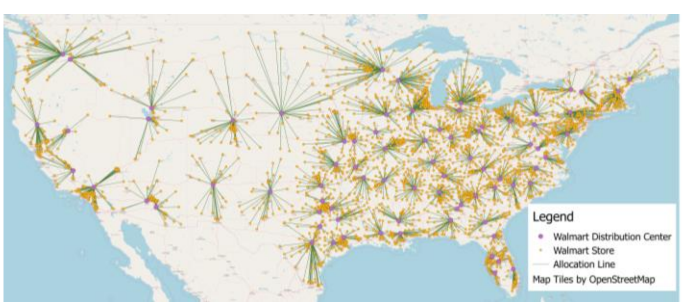

Fig. 1. Allocation of Walmart stores to their closest distribution center based on shortest network distance

The analysis of the results from this phase showed that the distance between some stores and their closest distribution center is about 1000 km (over 600 mi) which is likely to be caused by the distribution center locations file being incomplete. To address this issue, the stores farther from 129 km (80 mi) from the closest distribution center were excluded from the analysis. This resulted in a total of 1,771 stores being considered in the preliminary evaluation phase as well as scenarios 1 and 2 in the location phase.

All 3,163 Walmart store locations were used in scenario 3 of the location phase to locate 120-150 distribution centers.

C. *Preliminary Evaluation*

Once all the required data had been collected and organized, a preliminary evaluation of the current allocation of Walmart stores to their corresponding distribution centers was performed by calculating the total transportation cost (TTC).

$$TTC = \sum_{i=1}^{78} \sum_{\forall j \in S_i} d_{ij} p_j \qquad (4)$$

In Equation (4), d_{ij} is the shortest network distance (in kilometers) from distribution center i to its allocated store j, p_j is the estimated demand for store j, and S_i is the set of stores allocated to distribution center i.

The preliminary TTC based on the current allocation of stores to distribution centers was calculated as 9,684,166.

D. *Location*

In the location phase, the location of Walmart distribution centers was determined so that the total demand weighted transportation cost between distribution centers and stores is minimized. The three following scenarios were investigated with respect to the allocation of stores to distribution centers:

1) 1,771 selected stores were allocated to their closest distribution center based on the current location of the distribution centers. Then, each distribution center was located using the MTD algorithm to minimize the total demand weighted transportation cost between each distribution center and the stores allocated to it.
2) 1,771 selected stores were grouped into 78 clusters based on proximity. One distribution center was located to serve all stores in each cluster using the MTD algorithm to minimize the total demand weighted transportation cost in each cluster.
3) All 3,163 stores were grouped into 120-150 clusters based on proximity. One distribution center was located to serve all stores in each cluster using the MTD algorithm to minimize the total demand weighted transportation cost in each cluster.

The MTD algorithm was implemented in a solver software developed in Java based on the open source project GraphHopper [21]. GraphHopper is an open source, web-based routing engine developed in Java. Some of the most popular shortest path algorithms such as A* and Dijkstra's are already built into GraphHopper's routing engine.

GraphHopper is released under the Apache License which allows developers to "use the software for any purpose, distribute, modify, or distribute the modified version of software without the concern of royalties" [22].

In the second and third scenarios, a heuristic approach based on the clustering algorithm proposed by Klincewicz [23] is used to cluster stores based on proximity. Klincewicz's clustering algorithm was originally proposed for solving *p*-hub median problems. The *p*-hub median problem shares the same objective function with the *p*-median problem with an additional assumption that there can be a flow between hubs (facilities).

In Klincewicz's clustering algorithm, demand points (stores in the current problem) were first grouped into *p* clusters based on proximity. Then, the optimal location for the facility serving demand points in each cluster was determined.

In this study, two modifications were made to Klinewicz's method to apply it to *p*-median problems. First, all terms for inter-hub traffic were removed from the algorithm, which means that the clustering is only performed based on the flows between demand points and facilities. Also, instead of using the geometric center of mass to locate the facility serving each cluster, the MTD algorithm was used to locate the facilities at the network median of each cluster.

Following are the steps in the clustering algorithm:

- Demand points are sorted by their demand in descending order.
- The first *p* demand points are chosen as initial clusters.
- Each unassigned demand point is assigned to the closest cluster. The closest cluster is found by calculating the straight-line distance between the demand point and the geometric center of mass of each cluster and selecting the minimum value.
- In the exchange stage, S_{ik} which is the cost of re-assigning a demand point i to every other cluster k is calculated for all demand points. The exchange with the maximum positive S_{ik} value is performed and demand point i will be re-allocated to cluster k. This step is repeated until no positive S_{ik} is found.

Once the allocation of the demand points to facilities was determined by applying the clustering algorithm, the facility serving each cluster was located at the network median of the demand points in each cluster using the MTD algorithm.

Fig. 2. shows the resulting 78 clusters of Walmart stores on the map. Circles represent the store locations. The stores served by the same distribution center are enclosed in a polygon. Each polygon represents a cluster and displayed in an identical color.

The map data file of the continental United States is approximately 110 Gigabytes in size and its corresponding graph (i.e., the graph the solver software uses to solve location problems) has over 28 million nodes. To perform this analysis, an 8 core Intel Xeon (E5-2640 v3) 2.6 GHz Processors with 64 GB of memory, running Windows Server 2016 operating system and Java Run Environment 1.8 was used.

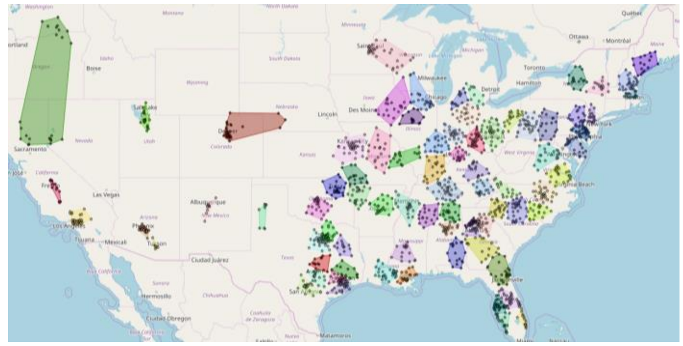

Fig. 2. 78 resulting clusters after applying the clustering algorithm on 1,771 Walmart stores

The 150 resulting clusters for 3,163 stores in one of the problem instances in scenario 3 is displayed in Fig. 3.

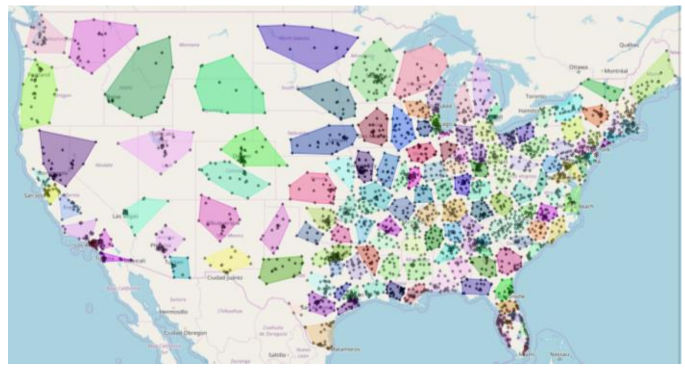

Fig. 3. 150 resulting clusters after applying the clustering algorithm on 3,163 Walmart stores

IV. Results

Table I summarizes the results of analyzing the weighted transportation costs of 78 Walmart distribution centers before and after relocating the distribution centers in scenarios 1 and 2. Locating 78 distribution centers took 51 minutes in scenario 1 and 45 minutes in scenario 2. Results of the preliminary evaluation are presented in the column labeled current in the table. The rows labeled minimum, average, and maximum reflect the lowest, average, and highest total weighted transportation cost among all 78 distribution centers respectively.

In scenario 1, after relocating Walmart distribution centers using the MTD algorithm, the TTC was reduced to 5,740,258.62. This represents an improvement of 41% in the TTC with respect to the current state. The minimum, average, and maximum weighted transportation cost for distribution centers also showed an improvement of 12%, 41% and 50% respectively compared to the current state.

TABLE I. Summary of Scenarios 1 and 2 Results

Weighted TSP Cost	Current	Scenario 1	Scenario 2
Minimum	1,136.17	1,004.35	12,348.41
Average	124,155.98	73,593.06	66,536.99
Maximum	890,163.57	444,563.13	508,797.9

Total	9,684,166.26	5,740,258.62	5,189,885.12

In scenario 2, the allocation of the stores to 78 distribution centers was determined by applying the clustering algorithm presented in section D while the distribution centers were located using the MTD algorithm. In this scenario, TTC was further decreased to 5,189,885.12 which represents a 46% improvement compared to the current state. The average and maximum weighted transportation cost for distribution centers also experienced improvements of 46% and 43% respectively compared to the current state, while the minimum distribution center weighted transportation cost increased significantly.

Fig. 4. and Fig. 5. depict two examples of recommended distribution center relocations in scenario 1. Circles represent the store locations, while diamond and star icons depict the current and proposed distribution center locations respectively.

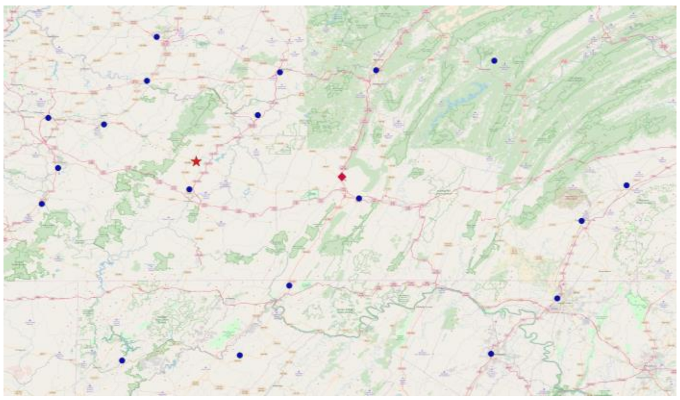

Fig. 4. Example of a distribution center relocation recommended by the MTD algorithm

Fig. 4. depicts a case in which the current location of the Walmart distribution center serves its allocated stores quite well although the distribution center location recommended by the MTD algorithm is 45 km (28 mi.) away from the current distribution center. In this case, the improvement in the weighted transportation cost that resulted from relocating the distribution center is 8%.

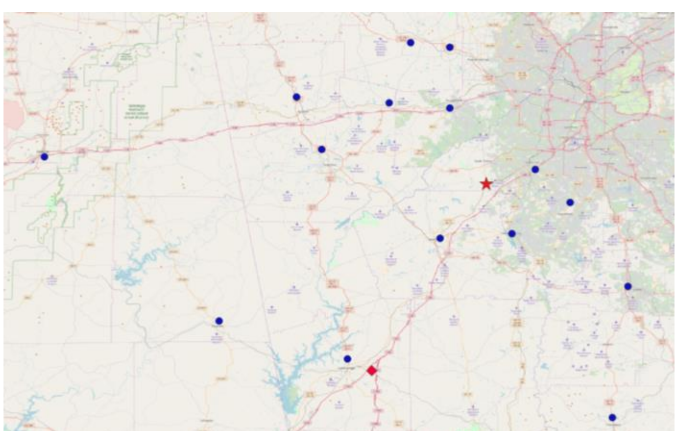

Fig. 5. Example of a distribution center relocation recommended by the MTD algorithm

In contrast, Fig. 5. depicts a case in which the current location of the Walmart distribution center is too far from stores with high demand. This situation can be improved by relocating the distribution center to or near the recommended location. In this case, the optimal distribution center location is 62 kilometers (39 mi.) from the current location. If the distribution center was relocated, it would translate into a 52% improvement in the weighted transportation cost.

The comparison of the results from scenarios 1 and 2 shows that TTC and average weighted transportation costs were improved by 10% in scenario 2 while both minimum and maximum distribution center weighted transportation costs increased.

The results for scenario 3 are presented in Table II. In this scenario, all 3,163 stores were allocated to 120-150 distribution centers and then each distribution center was located by the MTD algorithm. Number of distribution centers (DCs), total weighted transportation cost (TTC), average distance between stores and distribution centers in km (Avg Dist), average weighted distance (Avg W Dist), and runtime in minutes (Time) are presented in the table.

TABLE II. SUMMARY OF SCENARIOS 3 RESULTS

DCs	TTC	Avg Dist (km)	Avg W Dist	Time (min)
120	11,176,973.91	83.60	93,141.45	89.53
125	10,898,265.34	81.09	87,186.12	83.01
130	10,763,513.47	80.04	82,796.26	79.27
135	10,522,725.42	78.06	77,946.11	91.33
140	10,277,561.33	76.55	73,411.15	73.93
145	10,062,170.83	75.58	69,394.28	83.81
150	9,938,189.86	74.05	66,254.60	63.96

The results show that TTC decreases as the number of distribution centers increase which is an expected behavior in p-median problems. It is interesting that the TTC for the instance with 150 distribution centers is slightly (~3%) higher than the current TTC calculated in the preliminary evaluation phase for 1,771 stores. Also, average distance and average weighted distance decrease as the number of distribution centers increase. Average weighted transportation cost for the instance with 150 distribution centers in scenario 3 is comparable (.04% less) to scenario 2, although there are significantly more stores considered in scenario 3, which means that having 150 distribution centers will result in about the same level of service in terms of average weighted distance in scenario 2.

Runtimes for locating 120-150 distribution centers are between 64 to 91 minutes which is reasonable for a problem with 3,163 demand points on a network with over 28 million nodes.

V. CONCLUSIONS AND FUTURE WORK

In this research, a methodology for locating distribution centers based on the newly developed Multi-Threaded Dijkstra's (MTD) algorithm is proposed.

A case study involving 78 Walmart distribution centers and 3,163 stores was used to evaluate the performance of the proposed methodology. The demand for each store was estimated based on the population of urban area the store is located in. The real road network graph of the United States, composed of over 28 million nodes, was used as a basis for locating distribution centers. Three scenarios were investigated with regards to the allocation of stores to distribution centers: (i) allocating 1,771 selected store to their closest distribution center based on the network distance between the store location and the current location of the 78 distribution centers, (ii) clustering 1,771 selected stores based on proximity using a modified version of the clustering algorithm proposed by Klincewicz [23] and locating 78 distribution centers, and (iii) clustering all 3,163 stores and locating 120-150 distribution centers.

The results showed that the total transportation cost improved by 41% and 46% in scenarios 1 and 2 respectively, while the average weighted transportation cost decreased by 41% and 43% in scenarios 1 and 2 respectively. Choosing 150 distribution centers in scenario 3 resulted in the same average weighted distance as scenario 2, although the number of stores in scenario 3 is almost twice as scenario 2. The software implementation of the MTD algorithm was able to locate the 78 distribution centers in scenarios 1 and 2 in 51 and 42 minutes, and 81 minutes on average in scenario 3 which is reasonable considering the size of the network and problems.

The opportunities for future work are as follows:

- Walmart store and distribution center location data used in this research dates back to 2006. Performing the same analysis on more recent data can provide an insight on how the location of Walmart distribution centers have evolved in response to the competition.
- Demand values for the stores were estimated based on the population of urban areas. A more complex demand estimation model can be adopted to improve the accuracy of the analysis.
- More complex factors such as traffic and road type could be incorporated in addition to the distance to calculate the transportation cost.
- And finally, other objective functions such as minimizing maximum weighted distance, or minimizing maximum weighted travel time can be investigated and compared to the current analysis.

VI. REFERENCES

[1] G. Laporte, S. Nickel, and F. S. da Gama, *Location Science*. Springer, 2015.

[2] X. Xu and W. Liu, "Research on Logistics Distribution Center Location Model Based On GIS," *Int. J. Digit. Content Technol. Its Appl.*, vol. 7, no. 7, pp. 538–545, Apr. 2013.

[3] C. S. ReVelle, H. A. Eiselt, and M. S. Daskin, "A bibliography for some fundamental problem categories in discrete location science," *Eur. J. Oper. Res.*, vol. 184, no. 3, pp. 817–848, Feb. 2008.

[4] S. L. Hakimi, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," *Oper. Res.*, vol. 12, no. 3, pp. 450–459, May 1964.

[5] S. L. Hakimi, "Optimum Distribution of Switching Centers in a Communication Network and Some Related Graph Theoretic Problems," *Oper. Res.*, vol. 13, no. 3, pp. 462–475, May 1965.

[6] O. Kariv and S. L. Hakimi, "An Algorithmic Approach to Network Location Problems. I: The p-Centers," *SIAM J. Appl. Math.*, vol. 37, no. 3, pp. 513–538, Dec. 1979.

[7] A. Al-khedhairi, "Simulated Annealing Metaheuristic for Solving P - Median Problem," *Int. J. Contemp. Mathmatical Sci.*, vol. 3, no. 28, p. 10, 2008.

[8] E. S. Correa, M. T. A. Steiner, A. A. Freitas, and C. Carnieri, "A Genetic Algorithm for the P-median Problem," in *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, San Francisco, CA, USA, 2001, pp. 1268–1275.

[9] J. E. Beasley, "OR-Library: Distributing Test Problems by Electronic Mail," *J. Oper. Res. Soc.*, vol. 41, no. 11, pp. 1069–1072, 1990.

[10] P. Avella, M. Boccia, S. Salerno, and I. Vasilyev, "An aggregation heuristic for large scale p-median problem," *Comput. Oper. Res.*, vol. 39, no. 7, pp. 1625–1632, Jul. 2012.

[11] P. Rebreyend, M. Han, and J. Håkansson, "How Does Different Algorithm Work When Applied on the Different Road Networks When Optimal Location of Facilities Is Searched for in Rural Areas?," in *Web Information Systems Engineering – WISE 2013 Workshops*, 2013, pp. 284–291.

[12] P. Rebreyend, L. Lemarchand, and R. Euler, "A Computational Comparison of Different Algorithms for Very Large p-median Problems," in *Evolutionary Computation in Combinatorial Optimization*, G. Ochoa and F. Chicano, Eds. Springer International Publishing, 2015, pp. 13–24.

[13] T. J. Holmes, "Wal-Mart Data for 'Diffusion of Wal-Mart...,'" Mar-2010. [Online]. Available: http://users.econ.umn.edu/~holmes/data/WalMart/. [Accessed: 16-May-2016].

[14] T. J. Holmes, "The Diffusion of Wal-Mart and Economies of Density," *Econometrica*, vol. 79, no. 1, pp. 253–302, Jan. 2011.

[15] Dotsquare LLC., "Geocodio," *Geocodio*, 2014. [Online]. Available: https://geocod.io. [Accessed: 18-May-2016].

[16] G. D. US Census Bureau, "2010 TIGER/Line® Shapefiles," 2010. [Online]. Available: https://www.census.gov/cgi-bin/geo/shapefiles/index.php?year=2010&layergroup=Urban+Areas. [Accessed: 18-May-2017].

[17] OpenStreetMap, "OpenStreetMap Foundation," 21-Oct-2015. [Online]. Available: http://wiki.osmfoundation.org/wiki/Main_Page. [Accessed: 05-Nov-2015].

[18] pgRouting Community, "pgRouting Project — Open Source Routing Library," *pgRouting*. [Online]. Available: http://pgrouting.org/. [Accessed: 14-Apr-2018].

[19] C. Moeller, "osm2po - openstreetmap converter and routing engine for java," *OSM2PO*. [Online]. Available: http://osm2po.de/. [Accessed: 14-Apr-2018].

[20] P. Karich, "GraphHopper - OpenStreetMap Wiki," 12-Aug-2015. [Online]. Available: http://wiki.openstreetmap.org/wiki/GraphHopper. [Accessed: 06-Nov-2017].

[21] Apache Software Foundation, "Apache License," Jan-2004. [Online]. Available: http://www.apache.org/licenses/LICENSE-2.0.html. [Accessed: 07-Mar-2017].

[22] J. G. Klincewicz, "Heuristics for the p-hub location problem," *Eur. J. Oper. Res.*, vol. 53, no. 1, pp. 25–37, Jul. 1991.