UNIVERSITY THOUGHT
Publication in Natural Sciences, Vol.10, No.1, 2020, pp. 63-68.
DOI: https://doi.org/10.5937/univtho10-25351
Original Scientific Paper

COOPERATIVE MOBILE DATA COLLECTION IN SMART CITIES

IZZET FATIH SENTURK[1,⋆]

[1]Faculty of Engineering and Natural Sciences, Bursa Technical University, Bursa, Turkey

ABSTRACT

Smart cities are driven by huge amount of data collected from sensors deployed across the city. Sensors typically form a multi-hop network with a base station (BS) in order to send their data to the command and control center. However, sparse deployment of sensors can leave subsets of the network partitioned from the rest of the network. In such a case, isolated partitions cannot forward their data to the BS. Consequently, network coverage and data fidelity decline. A possible solution to link partitions and provide connectivity is employing mobile data collectors (MDCs). A smart vehicle supporting wireless communication can act as an MDC and carry data between sensors and the BS. Using a single MDC extends the average tour length. To minimize the maximum tour length, multiple MDCs can be employed. To identify sensors to be visited by each MDC, this paper clusters partitions as many as the number of MDCs and assigns an MDC for each cluster. Then two different cooperative data collection schemes are considered based on the availability of inter-MDC data exchange. If MDCs collaborate in data delivery, they meet at certain meeting points for data exchange. Such a cooperation avoids the requirement of visiting the BS for some MDCs and reduces tour lengths. On the other hand, MDCs closer to the BS can experience data loss due to buffer overflow given the higher volume of the accumulated data. Presented approaches are evaluated in terms of maximum tour length, data latency, and data loss. The smart city application is simulated with deployment of sensors on certain amenity types. Geographic data is obtained from a volunteered geographic information system and MDC mobility is restricted with the road network. Obtained results indicate that MDC cooperation decreases maximum tour length at the expense of increased rate of data loss and data latency.

Keywords: Wireless sensor networks, Caching, Buffer overflow, Latency, Energy consumption.

INTRODUCTION

The smart city phenomenon is a reaction against the continuous worldwide urbanization which leads to increased demand on limited resources in urban areas. Resource efficiency is a must for cities and gathering required data in a timely manner is essential to make better decisions regarding the management of urban infrastructure and services. The decision making part is handled by the the data analytics platform of the smart city framework by integrating big data and artificial intelligence. This framework is fueled with the data loop between sensors deployed across the city, urban infrastructure and services, and the data analytics platform. Wireless communication technology is the link in this loop which provides the most efficient solution for connectivity.

Despite availability of various wireless communication technologies, sensors are often restricted with limited transmission ranges to reduce energy consumption of the radio module. Therefore, sensors typically form a multi-hop Wireless Sensor Network (WSN) connected to the rest of the world with a base station (BS). However, smart city applications require sensor deployment across a large area and network connectivity can be partitioned due to the sparsity of the network. To link the partitions, various solutions can be pursued. Given the sensors are stationary, either additional nodes can be added between partitions to link them or a mobile data collector (MDC) can be used to carry data between partitions and the BS.

According to a recent study, linking partitions with additional nodes requires deploying nodes more than 94% of the network size (Senturk & Kebe, 2019) when low-rate wireless communication technologies are employed. To avoid the complexity of additional node deployment, this paper considers mobile data collection to sustain connectivity in smart city applications. An MDC can be a smart vehicle with wireless communication support. MDCs can be dedicated vehicles employed for mainly data collection. In such a case, mobility of the vehicle can be controlled and optimized based on considered metrics such as travel length or delay. It is also possible to use public transport vehicles following pre-determined paths. This approach eliminates the additional cost of vehicles in the expense of increased tour lengths and data latency due to the lack of mobility control. Worse, some regions will be likely out of network coverage. To ensure network connectivity, this paper assumes controlled mobility with dedicated vehicles. Nevertheless, the emergence of connected vehicles and the distributed ledger technology are promising to avoid the requirement of dedicated vehicles while ensuring data privacy. A sample demonstration of mobile data collection can be found in Figure 1.

Using a single MDC to link the whole network extends the resulting tour length. On the other hand, employing multiple MDCs can reduce the maximum tour length if the workload can be distributed uniformly between MDCs. To ensure a fair distribution of sensors to be visited between MDCs, this paper employs

⋆ Corresponding author:izzet.senturk@btu.edu.tr

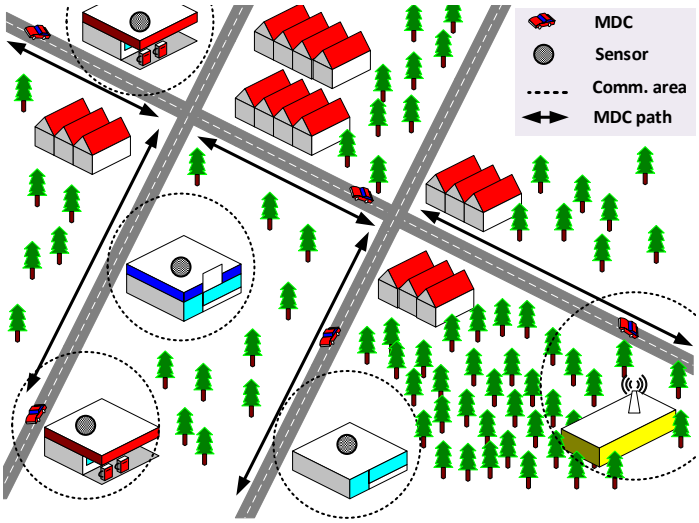

Figure 1. Mobile data collection with smart vehicles.

Un-weighted Pair-Group Method using Arithmetic Averages (UPGMA) clustering algorithm. The idea is grouping partitions according to the number of available MDCs. Subsequently, each MDC is assigned to a cluster. Each cluster can contain multiple partitions comprising several sensors. Visiting one of the sensors enables data collection from the whole partition considering the availability of multi-hop routing. Thus, it is sufficient to visit only a subset of sensors to collect data from the network. Visiting a sensor implies approaching close enough to establish wireless communication. Based on the distance between sensors, MDC can collect data from multiple sensors of different partitions at certain locations. It can be argued that total tour length of MDCs can be reduced by minimizing the number of locations to be visited. However, minimizing the locations to be visited to provide full network coverage is a complicated problem. This paper follows the Steiner Zone with Partitions (SZP) approach to identify locations to be visited in each cluster. SZP defines circular disks to represent communication areas assuming omnidirectional antennas and evaluates the degree of disk overlaps. SZP designates visiting points within the overlapping regions and favors overlaps with the highest degrees first.

When multiple MDCs are available, MDCs can cooperate in data delivery through inter-MDC data exchange. To exchange data, two or more MDCs meet at a certain meeting point. Based on the direction of the data exchange, this cooperation can be modeled as a tree where the *BS* is the root and data moves from leaf nodes to the root through parent nodes. Each parent node waits at the meeting point until collecting data from its children. The size of the data accumulated at parent nodes increases and MDCs closer to the *BS* can experience data loss due to buffer overflow. On the other hand, the tree structure obtained through inter-MDC data exchange avoids visiting the *BS* for most of the nodes and reduces tour lengths. To avoid extended tour lengths, this paper assumes availability of multiple MDCs and considers two different use-cases based on the availability of MDC cooperation in data exchange. Cooperative mobile data collection (C-MDC) assumes availability of inter-MDC data exchange. Individual mobile data collection (I-MDC) avoids this assumption and requires all MDCs to visit the *BS* to deliver their data. Limited cache size is assumed for MDCs. This paper evaluates both approaches in terms of maximum tour length, data latency and data loss through simulations considering metropolitan cities in Turkey.

RELATED WORK

Determining the order of visits among the given set of cities which yields the shortest tour is regarded as the traveling salesman problem (TSP). In TSP, the tour is circular and the salesman returns to the initial city. This paper considers a variation of the TSP problem known as TSP with neighborhoods (TSPN). TSPN seeks the shortest tour visiting a set of polygons. Unlike TSP which considers discrete points, the search space is continuous in TSPN. Considered problem is regarded as TSPN since we assume availability of wireless communication for sensors. Assuming omnidirectional antenna, regions to be visited can be represented with disks. For each sensor, a disk is defined centered at the location of the respective sensor. The radius of the disk is equal to the transmission range (R). Disks overlap if the distance between sensors is less than $2 \times R$. After defining disks, the problem of visiting sensors and returning to the *BS* with the shortest tour length becomes TSPN.

In the literature, various solutions are available to solve TSPN (Alatartsev et al., 2013a,b; Shuttleworth et al., 2008; Gulczynski et al., 2006). CIH (Alatartsev et al., 2013a) and C3-Opt (Alatartsev et al., 2013b) consider the TSPN problem as the combination of TSP and TPP (Touring a sequence of Polygons Problem). The idea is applying one of the tour improvement algorithms (e.g. 3-Opt (Frederick, 1958), Rubber-band algorithm (Pan et al., 2010)) as the TPP solver and then applying one of the available TSP solutions to the obtained set of points. (Shuttleworth et al., 2008) considers the problem of automated meter reading using radio frequency identification (RFID) technology. The goal is collecting readings from meters by approaching close enough while minimizing the tour length. The problem is formulated as the close enough TSP (CETSP). The problem defined in (Shuttleworth et al., 2008) is very similar to the considered problem since they also assume a road network with street segments. However, they employ a propriety software to solve the problem. Several heuristics are presented in (Gulczynski et al., 2006) for CETSP. This paper employs SZP to identify positions to be visited in the two-dimensional Euclidean space. SZP extends Steiner Zone approach presented in (Gulczynski et al., 2006). However, unlike Steiner Zone approach, SZP can handle partitions with multiple sensors.

Presented work is different from earlier studies due to the additional complexity of considered assumptions. First of all, this paper employs multiple MDCs to collect data. Therefore, the goal is not only minimizing the total tour length but also balancing the

load uniformly between MDCs. In the literature, assuming multiple salesmen is regarded as the multiple traveling salesmen problem (mTSP) (Bektas, 2006). In mTSP, a single depot node exists. Tours of all salesmen start and end at the depot. Presented problem is similar to mTSP since multiple MDCs are available with a single *BS*. This paper employs Un-weighted Pair-Group Method using Arithmetic Averages (UPGMA) clustering algorithm to divide the workload between MDCs. The idea is grouping partitions according to the number of available MDCs. In this work, two different data collection approaches are proposed based on the availability of cooperation between MDCs. *I-MDC* requires each MDC to visit *BS* in order to deliver its data. *C-MDC*, on the other hand, exploits inter-MDC cooperation in data delivery.

Another challenge is restricting mobility with the road network. Most of the earlier studies assume availability of a direct path between nodes. This paper models the road network as a weighted graph with directed edges for corresponding road segments. The road data is obtained from OSM (OpenStreetMap contributors, 2020). OSM is a volunteered geographic information (VGI) system. OSM data is collected using OSMnx (Boeing, 2017). Elevation data is obtained from Google Maps Elevation API (Platform, 2020).

DATA COLLECTION AND THE SIMULATION SETUP

This paper simulates data sampling from sensors deployed across a city and employs presented mobile data collection schemes to collect sampled data assuming a smart city application at the top. To simulate considered scenarios in a realistic manner, sensor locations and corresponding data generation rates are determined using OSM, a geographic information system. OSM follows a participatory approach to collect geospatial data. OSM models the real world using three basic data elements: node, way, and relation. Node denotes a point in the space. Way is an ordered list of nodes. Relation signifies how different elements interact. To designate sensor locations, Points of Interest (POIs) are obtained through OSM and sensors are deployed on certain POIs. This paper considers three POIs, namely school, hospital, and police station for sensor deployment. Data generation rates are as follows: 1, 2, and 3 sampling per second for hospitals, police stations, and schools respectively.

A POI can be represented as a node or as a polygon (i.e. way) to denote the boundary of the building. If the POI is represented as a polygon, multiple sensors are deployed to monitor the building. The actual sensor location is determined according to the drivable road network where MDCs can travel to ensure data collection from the sensor. Therefore, sensors are located at the closest road segment. A common transmission range is used for both sensors and MDCs. The transmission range is varied between 20 and 50 in the experiments. OSM also provides drivable road segments and mobility of MDCs is restricted with the road network. To calculate latency in a realistic manner, velocity of the MDCs are set dynamically according to the speed limit of the crossed road segment. OSM provides speed limits. If the maximum speed limit is not available on OSM for the considered road segment, a default limit of 50 km/h is used.

We assume availability of multi-hop routing between sensors. Thus, sensors can form a connected component (i.e. partition). Multi-hop communication enables data collection from the whole network upon visiting one of the sensors in the partition. Identifying the minimum set of sensors to be visited which ensures full network coverage is a complicated problem and this paper employs Steiner Zone with Partitions (SZP) approach as discussed earlier. SZP provides visiting points for data collection. Visiting points are not necessarily sensor locations but coordinates in the two-dimensional Euclidean space given the availability of wireless communication. On the other hand, visiting points are always part of the drivable road network. Based on the employed transmission range, the average number of visiting points vary as shown in Table 1.

Table 1. The number of points visited by MDCs with respect to the employed transmission range. The average number for 30 cities is reported. The average sensor count is 103,73.

Transmission range	The number of visiting points
20	38,77
30	34,00
40	28,37
50	24,20

In the experiments, the number of MDCs is set to 3. Sensors are clustered using UPGMA (Wikipedia, 2020c) and each MDC is assigned to a cluster. The first sensor in the list of sensors is regarded as the *BS*. As mentioned earlier, two different approaches are considered for mobile data collection. I-MDC does not assume MDC cooperation and requires MDCs to visit the *BS* to deliver their data. In C-MDC, MDCs meet at certain meeting points to forward their data to the next MDC. MDC meeting points are determined according to cluster size of the respective MDCs. For two MDCs MDC_a and MDC_b assigned to clusters $cluster_a$ and $cluster_b$ respectively, two visiting points vp_a and vp_b are selected from respective clusters such that the distance between visiting points are the minimum. If the size of $cluster_a$ is smaller, MDC_a visits MDC_b at vp_b, and vice-versa.

After designating clusters along with the set of visiting points, data collection paths are computed for each MDC. Visiting a set of points and returning to the starting point with a path minimizing the objective function is regarded as the Traveling Salesman Problem (Wikipedia, 2020b). In this paper, we aim to minimize the path length of MDCs. Given the availability of multiple MDCs, the goal can be defined as minimizing the maximum path length. Inter-MDC data exchange can minimize the maximum path length by avoiding the requirement of visiting the *BS* for certain MDCs. On the other hand, cooperation in data delivery can increase latency and data loss due to the additional waiting

times and the extended size of the accumulated data. Data latency is computed based on the path length, velocity of the MDCs and the waiting time for other MDCs. Waiting time and the size of the accumulated data are computed by modeling the inter-MDC cooperation as a tree where the *BS* is the root. Each sensor sampling is assumed be sent in packets of 4 bytes. A buffer size of 1 MB is assumed for MDCs. TSP is solved using OR-Tools library (OR-Tools, 2020).

This paper considers 30 metropolitan cities of Turkey (Wikipedia, 2020a). Obtained spatial data is limited within the bounding box of 1 km from city centers. For statistical significance, the average results are reported.

EXPERIMENTAL EVALUATION

Performance metrics

- *Maximum path length*: As the name suggests, this metric reports the length of the longest path among MDCs. Shorter path length reduces the cost of mobility. If the MDC is battery operated this metric suggests the network lifetime.
- *Latency*: This metric denotes the time required to complete the path. Two main sources of delay are considered; the duration of mobility and the waiting time for other MDCs.
- *Total overflow*: This metric indicates the ratio of data loss due to buffer overflow.

Results

Experiment results in terms of maximum path length are shown in Figure 2. According to the obtained results, C-MDC outperforms I-MDC thanks to inter-MDC data exchange. The results indicate that MDC cooperation in data delivery alleviates the cost of mobility by reducing the length of the longest path up to 12 per cent. The decline in the maximum path length can be attributed to the lack of requirement of visiting the *BS* in order to deliver data. Mobile data collection scheme imposes extended paths for MDCs assigned to partitions with sensors deployed far from the *BS*. Cooperation in data delivery, on the other hand, enables forwarding collected data to the next MDC instead of visiting the *BS*. This paper considers an application area of 1 km and it can be argued that the performance gap between two approaches can increase with an extended application area.

It can also be noticed from Figure 2 that the maximum path length declines when the employed transmission range is extended. The decline in the maximum path length can be attributed to the decreased number of visiting points when the transmission range is increased as indicated in Table 1.

Figure 3 portrays experiment results in terms of data latency. Two main sources of latency are considered in the experiments, namely mobility delay and waiting delay. Mobility delay denotes the duration which the data is moved on MDCs until reaching the *BS*. Mobility delay is based on the MDC velocity and the length of the travel path. Considering the performance of C-MDC in terms

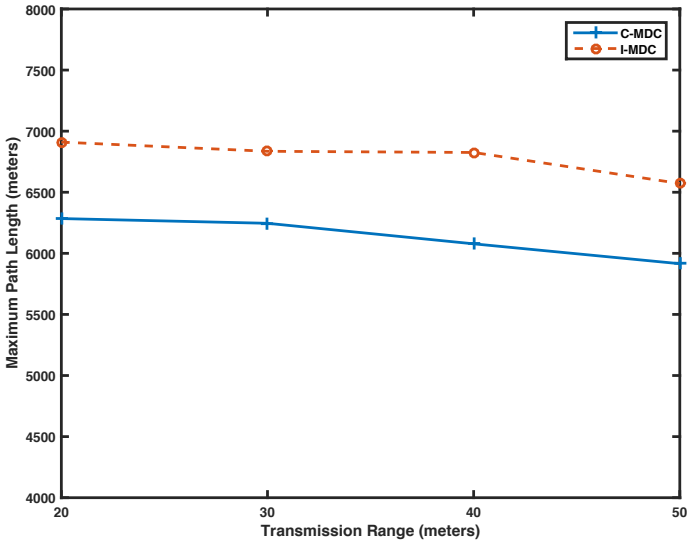

Figure 2. Experiment results in terms of maximum path length with respect to the employed transmission range.

of maximum path length, one can expected reduced mobility delay as well for C-MDC. However, obtained results indicate increased data latency up to 68 per cent for C-MDC compared to I-MDC. These results suggest the overwhelming impact of waiting delay in data latency.

Recall that the inter-MDC cooperation leads to a tree structure in data collection. The *BS* acts as the root and the direction of data transfer is from child nodes to their parents. Parent nodes wait at the meeting points to collect data until all of the child nodes arrive. Consequently, waiting delay can be considerably high to dominate the latency results. Note that the waiting time is zero for I-MDC since MDCs do not have to wait each other and rather forward the data to the *BS* directly.

According to obtained results, transmission range has limited impact on latency. For C-MDC, the latency declines slightly when the transmission range is extended. On the other hand, the latency is almost constant for I-MDC.

The results of the experiments to assess the ratio of total data loss due to buffer overflow are given in Figure 4. For both of the approaches, the rate of the total data loss is less than 3 per cent. However, C-MDC leads to increased data loss up to 75 per cent compared to I-MDC. Note that the size of the sampled data increases if the data collection is delayed. The main source of delay for I-MDC is mobility delay. As denoted in Figure 2, I-MDC leads to increased maximum path length which implies extended mobility delay. On the other hand, as illustrated in Figure 3, waiting delay dominates the overall latency for C-MDC. Consequently, I-MDC outperforms C-MDC by reducing data loss.

Another factor which impairs the performance of C-MDC is the cooperation in data delivery and the resulting tree structure. In I-MDC, MDCs carry data exclusively from their respective clusters. On the other hand, C-MDC leads to a tree structure and imposes parent MDCs to carry the data of their children. Given the

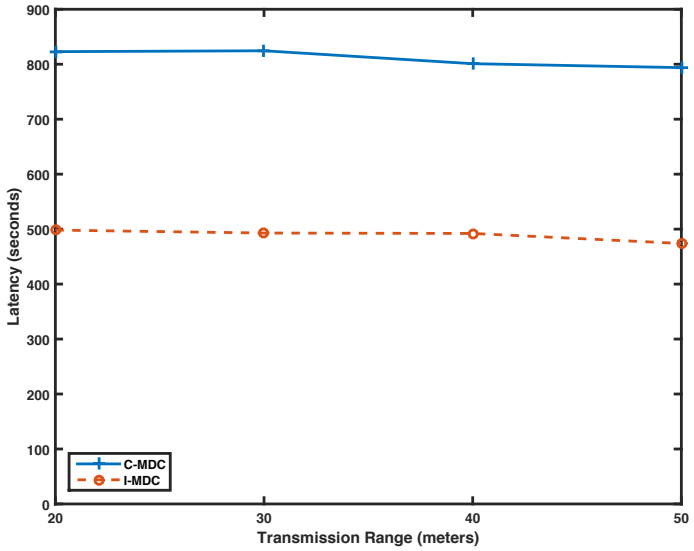

Figure 3. Experiment results in terms of latency with respect to employed transmission range.

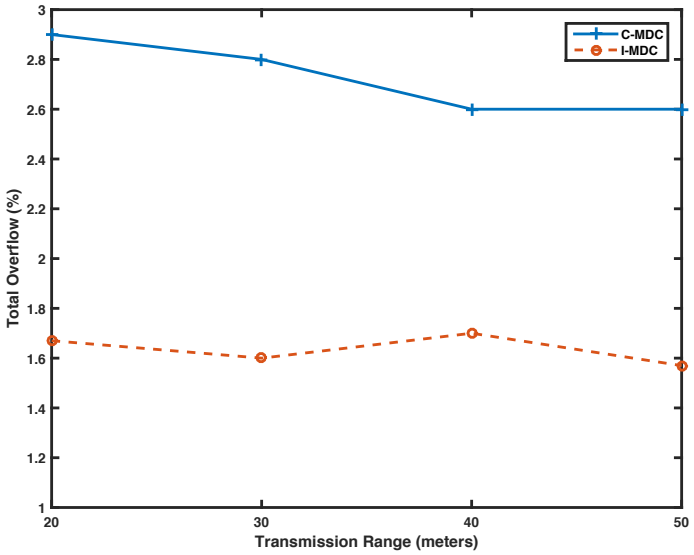

Figure 4. Experiment results in terms of data loss with respect to employed transmission range.

large volume of the accumulated data for MDCs closer to the *BS*, data loss is more likely for C-MDC compared to I-MDC.

The results indicate that increased transmission range can alleviate data loss for C-MDC. The rate of the data loss fluctuates for I-MDC for varying transmission ranges.

CONCLUSION

This paper presents two approaches for mobile data collection considering a smart city application. The idea is employing smart vehicles to collect sensor data in a sparse wireless sensor network deployed at city scale. One of the approaches, C-MDC, exploits cooperation of the vehicles in data delivery to the *BS*. C-MDC leads to a tree structure modeling the data collection. In this tree, the *BS* acts as the root and the data is forwarded from children to parents. This scheme avoids the requirement of interacting with the *BS* directly as in the other approach, I-MDC. To analyze the trade-offs, both approaches are evaluated in terms of maximum path length, data latency, and the rate of data loss. The results show that C-MDC decreases maximum tour length at the expense of increased rate of data loss and data latency. To simulate the smart city application in a realistic manner, sensor locations and data generation rates as well as the mobility path of the vehicles are determined according to geospatial data obtained from a volunteered geographic information system.

ACKNOWLEDGEMENT

This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK) under Grant No. EEEAG-117E050.

REFERENCES

Alatartsev, S., Augustine, M., & Ortmeier, F. 2013a

Alatartsev, S., Mersheeva, V., Augustine, M., & Ortmeier, F. 2013b

Bektas, T. 2006, The multiple traveling salesman problem: an overview of formulations and solution procedures, Omega, 34(3), pp. 209-219. doi.org/10.1016/j.omega.2004.10.004

Boeing, G. 2017, OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks, Computers, Environment and Urban Systems, 65, pp. 126-139. doi.org/10.1016/j.compenvurbsys.2017.05.004

Frederick, B. 1958, An algorithm for solving travelling-salesman and related network optimization problems, Operations Research, 6(6), pp. 897-897.

Gulczynski, D. J., Heath, J. W., & Price, C. C. 2006, The Close Enough Traveling Salesman Problem: A Discussion of Several Heuristics. Boston, MA: Springer US, pp. 271-283. doi.org/10.1007/978-0-387-39934-8$_1$6

OpenStreetMap contributors. 2020, Planet dump retrieved from https://planet.osm.org, https://www.openstreetmap.org. Accessed January 20, 2020.

OR-Tools, G. 2020, The OR-Tools Suite, https://developers.google.com/optimization. Accessed January 20, 2020.

Pan, X., Li, F., & Klette, R. 2010, Approximate shortest path algorithms for sequences of pairwise disjoint simple polygons (Citeseer).

Platform, G. M. 2020, Elevation API, `https://developers.google.com/maps/documentation/elevation/start`. Accessed January 20, 2020

Senturk, I. F. & Kebe, G. Y. 2019, in A New Approach to Simulating Node Deployment for Smart City Applications Using Geospatial Data, International Symposium on Networks, Computers and Communications (ISNCC'19), June 18-20, 2019, Istanbul, Turkey, pp. 1-5.

Shuttleworth, R., Golden, B. L., Smith, S., & Wasil, E. 2008, Advances in Meter Reading: Heuristic Solution of the Close Enough

Traveling Salesman Problem over a Street Network. Boston, MA: Springer US, pp. 487-501. doi.org/10.1007/978-0-387-77778-8$_2$2

Wikipedia. 2020a, Metropolitan Municipalities in Turkey, https://en.wikipedia.org/wiki/Metropolitan_municipalities_in_Turkey. Accessed January 20, 2020.

Wikipedia. 2020b, Travelling Salesman Problem, https://en.wikipedia.org/wiki/Travelling_salesman_problem. Accessed January 20, 2020.

Wikipedia. 2020c, UPGMA (Unweighted Pair Group Method with Arithmetic Mean), https://en.wikipedia.org/wiki/UPGMA. Accessed January 20, 2020.