IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 19 October 2023, accepted 2 December 2023, date of publication 13 December 2023, date of current version 20 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3342188

RESEARCH ARTICLE

HiCARE: Hierarchical Clustering Algorithm for Road Service Routing Enhancement

SAMI ALSHAMMARI [1,2], SEJUN SONG [2], (Senior Member, IEEE), AND BAEK-YOUNG CHOI [2], (Senior Member, IEEE)

[1] Faculty of Computing and Information Technology, Northern Border University, Rafha 76321, Saudi Arabia
[2] School of Computing and Engineering, University of Missouri–Kansas City, Kansas City, MO 64110, USA

Corresponding author: Sami Alshammari (sami.alshammari@nbu.edu.sa)

ABSTRACT The potholes in the road cause substantial monetary and physical damage to the ongoing traffic. Also, it requires extensive maintenance, especially in areas where the temperature may go down below freezing point. One of the major causes of potholes is the rain or running water that is accumulated in the cracks and later on due to low temperature, changes into ice. Then, ice forms a larger volume for the same amount of water after expanding and causes cracks to expand and at some point become a pothole. They are also the cause of traffic congestion. Therefore, pothole repair needs urgent attention and is one of the routine road maintenance tasks. Kansas City is one of the cities with established social data networks for residents to request road services to mitigate the problem. Although the policymakers have not ignored the issue and rudimentary patching policies are in place, unfortunately, that does not provide efficient road maintenance routes. This paper proposes a Hierarchical Clustering Algorithm for Road Service Routing Enhancement (HiCARE). HiCARE is a practical framework, that makes use of open data in order to optimize the route for maintenance. The optimization considers the time and date of reported potholes, locations, traffic situations, weather conditions, type of patch or other repairs needed, and crew availability, to mention some. This research has characterized spatiotemporal pothole density by analyzing the past 16 years of pothole data from the Open Data KC 311 in Kansas City. HiCARE enhances the NP-hard Traveling Salesperson Problem (TSP) by classifying potholes into layers of clusters. HiCARE finds a cluster's shortest possible pothole route by identifying each cluster group's entrance and exit pothole points. Moreover, it modifies the final routes to skip any local minima. The empirical research and analysis indicate that HiCARE significantly reduces the traversing distance and is faster in computation time when compared to typical TSP heuristic algorithms for daily resolution scheduling.

INDEX TERMS Clustering, NP-hard problem, heuristic algorithms, maintenance engineering, roads maintenance, optimization methods, potholes, urban areas, processor scheduling, traveling salesman problems.

I. INTRODUCTION

Seasonal weather changes, mainly rains have drastic effects on the transportation infrastructure, especially the roads. Due to drastic changes in the weather and heavy traffic, the road pavements experience various surface distortions, including but not limited to potholes, cracking, groove, and depression. water from different sources leaches through crevices and cracks in the surface pavement. Mainly in winter, the water changes into ice when the temperature drops below the freezing point, causing expansion in the volume of the water and damaging/softening the pavements of the road. Traffic wear crushes the softened surface and each freeze-thaw cycle renders wider cracks and creates **potholes**. Road crevices and potholes can create unsafe driving conditions and result in thousands of dollars of damage per vehicle [25]. Furthermore, pothole repair is among the most costly road maintenance service. For instance, street repair workers in Kansas City, Missouri (KCMO) fixed over 116,000 linear feet of road and 20,000 potholes in 2019, as of 2023,

The associate editor coordinating the review of this manuscript and approving it for publication was Wentao Fan.

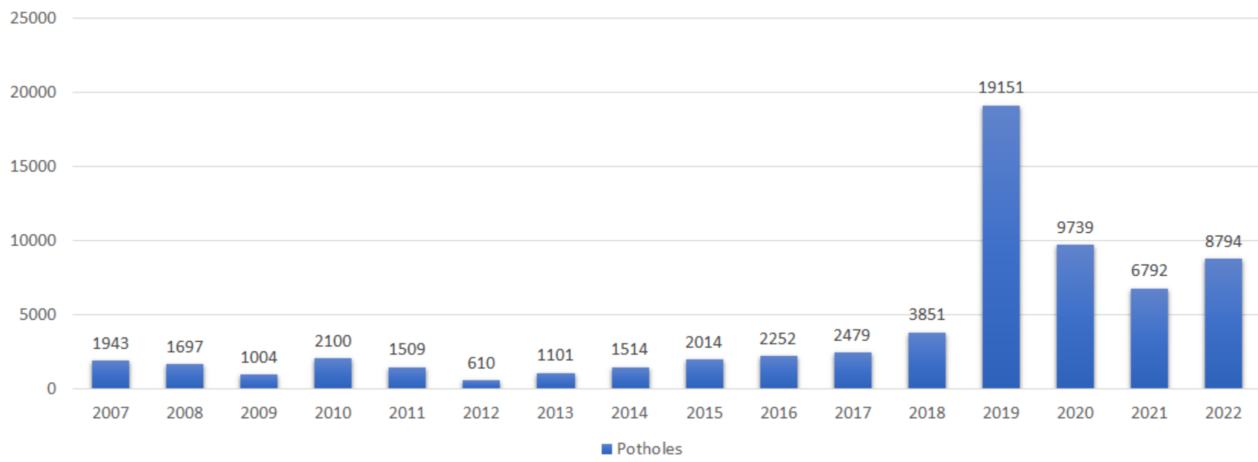


FIGURE 1. Service requests by years.

KCMO aims to resurface another 300 lane miles of streets with a nearly $40 million budget [1]. In order to manage patching operations in suburban neighborhoods, KCMO spends millions of dollars each year to fund street repair work [23]. Several cities have set up social data networks for locals to report their area's potholes to be fixed in a timely manner, for example, KCMO collects pothole reports directly from residents through different channels. These channels include emails, an online Open Data KC portal [23], and a non-emergency dial-up number(311). Every pothole reported to 311 appears on a maintenance list. Once the reports are received, the repair plan is laid out according to the nature of the complaint. It is worth mentioning that the plan is carried out without optimized route planning. This may lead to a substantial waste of time and cost, especially given the limited availability of trucks, trailers, and crew. Before moving ahead, it is essential to know what are the different kinds of road damage besides potholes. As discussed in [15], the damages can be categorized as detailed below.

- Cracking can be defined as damage that results from either improper construction material being used or when the traffic exceeds the maximum load capacity defined. Cracking can be considered as a low(gap width is specified as less than 6 mm) or high severity (gap width is specified as more than 19 mm). Furthermore, cracks can be identified from their different types i.e., fatigue cracking, block cracking, edge cracking, longitudinal cracking, reflection cracking at joints, and transverse cracking.
- Patching and potholes: Patching refers to those sections of the pavement, which have been identified as damaged and are usually bigger than 0.1 square meters. The damaged area is then cleaned and refilled with material and fixed after the original construction. Potholes are bowl-shaped damaged areas with a minimum dimension of 150 mm. The maximum depth of a pothole may reach below the pavement surface. If the depth is greater than 50 mm then it is considered a high-severity pothole and may cause severe damage to the ongoing vehicle.
- Surface deformation is normally caused by ongoing moving vehicles braking and accelerating motion. Also, it can be caused by heavy traffic or overloaded vehicles and may produce wavy-shaped horizontal or vertical deformation on the road. The main cause of deformation is poor quality material, or when the road quality is not properly adjusted for high-load vehicles e.g., trucks. There are two types of damage: rutting and shoving.
- Surface defect is reflected in the reduction in pavement surface friction. It can be due to many reasons such as the decomposition of granules, asphalt mud floating, or the loss of bonding between the construction materials. There are three types of this defect: bleeding, polish aggregate, and raveling.
- Miscellaneous distress does not belong to the above-mentioned categories and is mainly caused by segmental difference

Having so many categories of road damages and their types it is evident that road maintenance is a rigorous activity that needs large-scale management and availability of inventory and logistics. In a report published by Kentucky Transportation Center [3], it is clear that road maintenance needs a proper mechanism to be carried out effectively. The project report aims to help different counties to develop maintenance plans for crews. Furthermore, plans include an inventory of routine maintenance activities, and their frequency, and also outline how to address special maintenance projects. Additionally, it lays the groundwork for the Cabinet to develop a method for prioritizing maintenance and operational needs.

This research work proposes a novel approach, HiCARE (**Hi**erarchical **C**lustering **A**lgorithm for **R**oad Service Routing **E**nhancement) to address such issues. HiCARE is a practical framework for improving pothole resolution planning. It enhances the NP-hard Traveling Salesperson Problem

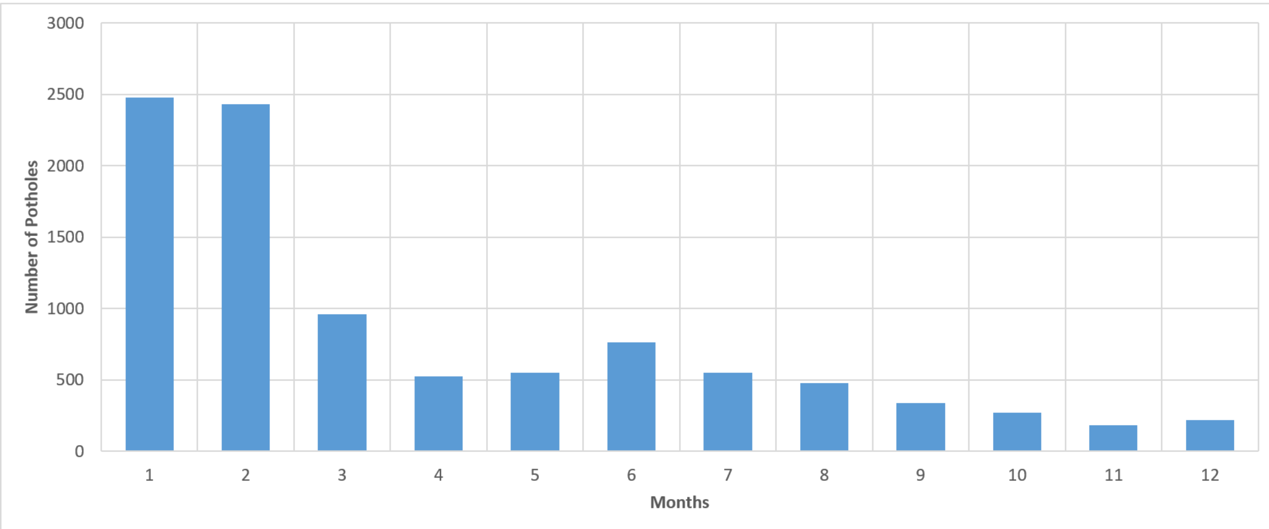


FIGURE 2. Service requests for Year 2020 where the COVID-19 caused state-wide lockdown started in March 2020.

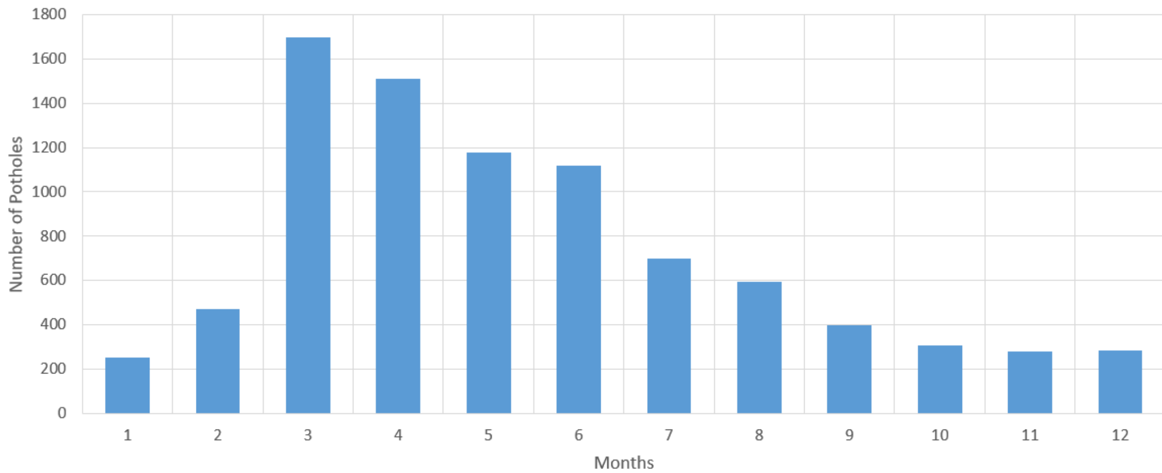


FIGURE 3. Service requests for year 2022.

(TSP) [18] [20] [4] [5] [6] [7] algorithms by classifying potholes into layers of clusters. HiCARE calculates the shortest possible pothole routes within and among the identified clusters using each group's entrance and exit points. Then it modifies the final routes to skip any local minima. Our analysis indicates that HiCARE significantly reduces the traversing distance and is faster in computation time than typical TSP heuristic algorithms, for daily resolution scheduling. Any road maintenance planning could benefit from applying HiCARE to get the most efficient maintenance routes.

In order to make it a robust and practical tool, we have used social network data for over a decade. Furthermore, we characterized spatiotemporal pothole density by analyzing the data from 2007 to 2022, of pothole data from the Open Data KC 311 in Kansas City [23]. The data contains pothole locations, damaged lanes, weather patterns, traffic circumstances, patch type, crew availability, etc. The data was properly cleaned and processed to identify many trends in the data of spatiotemporal potholes and related characteristics concerning their densities.

In Section II, we briefed and outlined the research gap that motivated us to conduct this research work. Section III provides a detailed review of the related work in this research work. Furthermore, in Section IV pothole data analysis and characterization are provided. The proposed HiCARE algorithm is detailed in Section V, along with its design. The results of our experiments are provided in Section VI. Finally, we conclude the research work in Section VII.

II. RESEARCH MOTIVATION

The literature has revealed that the majority of the study has been carried out to identify potholes or transportation improvement using various different techniques. Wu et al. [9] has proposed a random forest algorithm in order to detect potholes, with an accuracy of 95.7%. Basavaraju et al. [10],

TABLE 1. Open data KC pothole service reports by source.

Year	Report Source	Report count
2020	Phone	2777
	Web	4009
	Email	1206
	Others	1746
2019	Phone	7916
	Web	8584
	Email	2203
	Others	457
2018	Phone	2143
	Web	1292
	Email	348
	Others	81
2017	Phone	1590
	Web	585
	Email	192
	Others	271
2016	Phone	1371
	Web	652
	Email	192
	Others	37
2015	Phone	1235
	Web	634
	Email	124
	Others	23

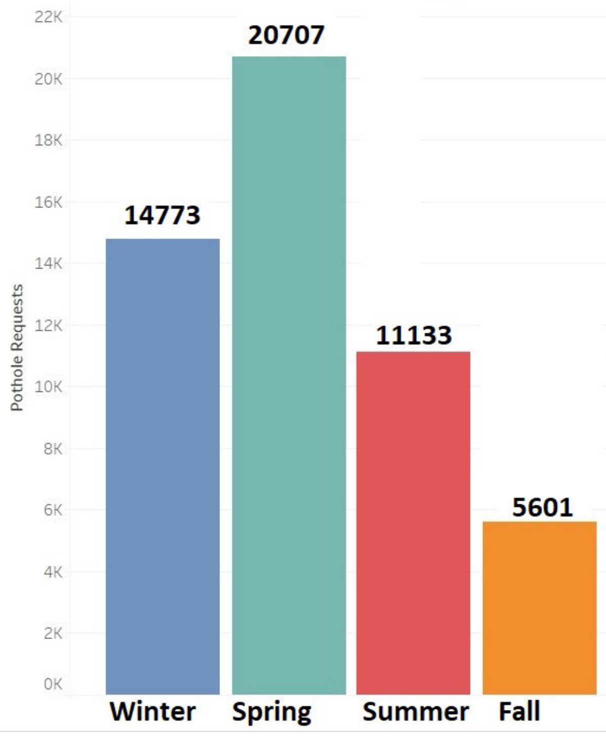


FIGURE 4. Service requests from 2007 to 2020 combined by seasons.

have used multilayer perception with an attained accuracy of 92.12% using a machine learning approach to assess road surface anomaly. Furthermore, Using a deep-learning approach i.e., Convolution Neural Networks (CNN) for road surface monitoring and detection of potholes yielded 93.0% results as highlighted by Varona et al. in [11]. Moreover, Using CNN Baldini et al. [12], obtained 97.2% accuracy. In this paper, they focused on road anomaly identification with the help of accelerometers and gyroscopes. Apart from the CNN, Luo et al. used Recurrent Neural Network, in their work [13], to obtain an astonishing 99.26% accuracy. They also addressed many different other techniques to detect road anomalies. While discussing the road anomalies, Hassan et al. [17], in their research work created their own dataset by mounting devices on twelve different vehicles and published their dataset. They focused more on classification and mainly used Support Vector Machine(SVM), Convolution Neural Networks(CNN), Random Forest(RF) and Nave Bayes(NB) as a low-cost road maintenance solution and provide comfortable routes for passengers. Furthermore, they concluded that the best results were provided by the SVM with the True Positive Rate (TPR) of 95.2%. The research work considered cat eyes, manholes, potholes, and speed bumps. Their method has one great quality, the road anomalies are detected automatically by the mounted devices and the users do not have to report them explicitly. These works show a promising future for the usage of machine learning and modern technologies in road anomalies detection which may also include potholes, cracks, and other deformities which may be considered serious road hazards. A review of such techniques can be found in [14]. However, there seems to be a solid gap between the detection of anomalies and dealing with such road hazards considering the high cost of repairs, road congestion, traffic flow, availability of resources, and logistical support. As mentioned earlier, the proposed system focuses mainly on optimizing routes in order to reduce the repair time and make the roads safer.

III. RELATED WORK

As discussed in the II, a vast majority of research work is taking place in the improvement of transport infrastructure with a main focus on identifying potholes, cracks, and road roughness using Internet of Things (IoT) sensors and visual identifications through machine learning techniques. However, the area of road repair routes optimization has been clearly ignored. Any promising findings on this topic will significantly benefit road services departments by reducing maintenance time along with cost and labor. It will be equally helpful for the road maintenance departments, where resources are limited and optimization of limited sources will be of great benefit in such cases. A few studies focus on finding optimal road network maintenance schedules. In a research work [15] specifically for Taiwan, the authors have considered road safety. Heavy rains in Taiwan cause road cracks and damage to the infrastructure. After that, the responsible department has to make inspection plans for damage and repair assessments. They have adopted an approach that is based on resource constrained project scheduling problem (RCPSP) framework, which is unlike Vehicle Routing Problem (VRP). They have encouraging results and reduction in total repair time and provide optimal decisions. Furthermore, the research work optimizes the repair sequence and transport activities. The authors further

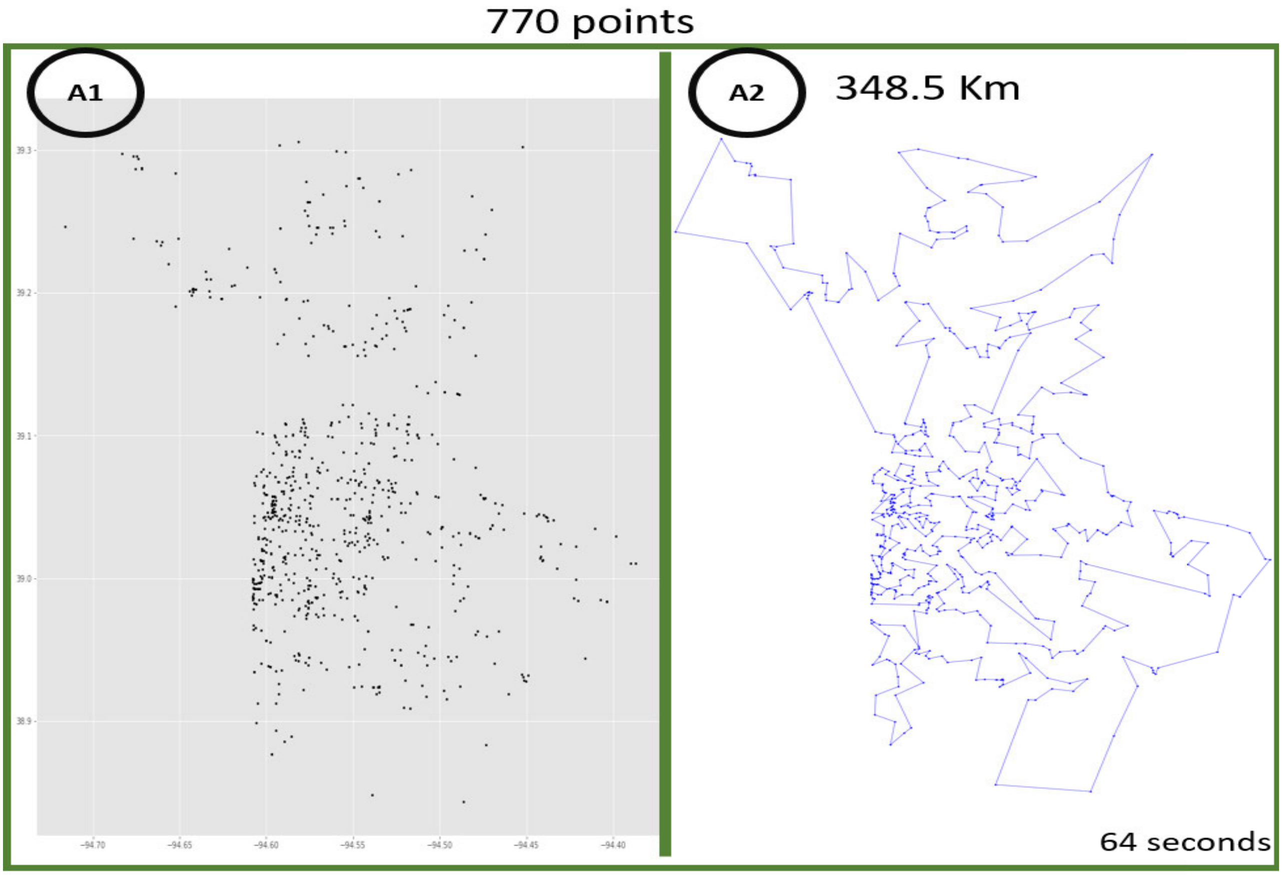


FIGURE 5. A1:770 potholes distributed on the map which have been reported in Spring2016. A2: Route generated by HiCARE.

discuss that the optimum travel path for trucks to reach the scattered potholes includes loading, repairing, and coming back to the initial position times. Whereas, ''transport activity'' refers to the traveling time between potholes. When graphically visualized, the potholes are considered nodes on the graph, while the vectors represent the path. If there is no vector between two nodes, it indicates that there is no direct path available for the two potholes. The paper concluded that when addressing the case of six potholes if compared with their current system, there is a reduction of at least 20 minutes of traveling time. Furthermore, the paper has defined the case studies of more than six potholes also. Similarly, another research for Gävle Municipality focused on 6712 potholes [16]. Their emphasis is on evaluating the minimal total road distance needed to visit each pothole at least once and come back to the initial position. The authors made use of the Nearest Neighbour Algorithm(NNA) along with Simulated Annealing (SA). They turned the potholes dataset into TSP, where each pothole act as a city. Somehow, the SA failed to improve over NNA, and there is a tradeoff of added distance with most solutions. The conclusion is that simulated annealing did not significantly add to the nearest neighbor algorithm. However, there might be room to improve the simulated annealing algorithm to yield better results using some more refined local search heuristics. There has been research work, that [35] focuses on network recovery, defining and prioritizing vital connections in the infrastructure. They have used the greedy algorithm, and have not considered the repair time and traveling time in their work. After each maintenance, they reevaluate links again, also the model is dynamic and the optimal solution changes with changing scenarios. The main point of the research work is to find the link, also referred to as a ''critical link'', which needs to be repaired first so that it has the most impact on the road network quality and avoids congestion. Furthermore, with different links repairing, the different and changed outcome is expected. However, our research focus is different in that we focus on route optimization for everyday maintenance, rather than finding or prioritizing the roads to be repaired first or on urgent bases. In an earlier research carried out in 2007, which is not directly related to the route optimization rather TSP as discussed in [34] improves calculation time, but it does not tackle any maintenance efficiency (i.e., distance and cost). The research concluded that Decomposition algorithms allow a substantial decrease in the running time of the clustered TSP, especially for the large- scale problems. On the contrary, the proposed model, HiCARE improves distance as well as time computation

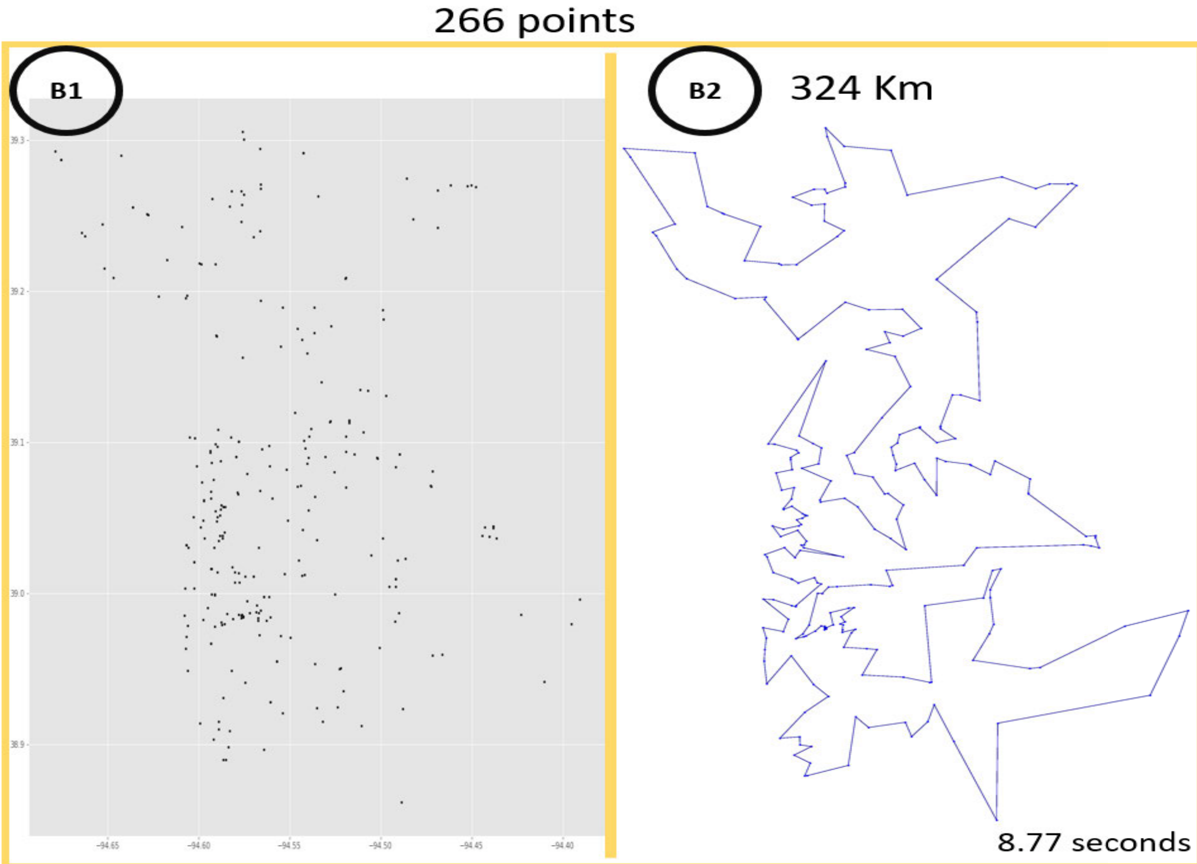


FIGURE 6. B1:266 potholes distributed on the map which have been reported in Fall2016. B2: Route generated by HiCARE.

using real-life data. In the aspect of Travelling Salesman Problem applications [8], [19] and [32] which are hybrid genetic algorithms using a sequential constructive crossover, 2-opt search, and a local search for obtaining a heuristic solution to the clustering problem. These studies measured the efficiency of certain asymmetric and symmetric TSPLIB instances of different sizes against two existing techniques. HiCARE is a Hierarchical Clustering Algorithm for Road Service Routing Enhancement that generates an optimized route for the Kansas City road crew by creating a hierarchical structure and applying two well-known algorithms, Simulated Annealing and Lin-Kernighan Heuristic algorithms, on different levels. Simulated Annealing helps HiCARE in optimizing routes in all clusters while Lin-Kernighan Heuristic algorithm support avoiding crossed edges which helps optimize the final route [4], [6], [8].

IV. ROAD SERVICE REQUEST DATA ANALYSIS

In an attempt to reduce the cost and time used to carry out a traveling pothole crew repair route, there was a need for a dataset that is specifically tailored according to our needs and research requirements. Therefore, the dates and locations of all pothole service requests from the past 14 years in Kansas City, MO were collected. collected the provided by Open Data KC. The program 311 Open Data is used where citizens reported different issues and problems. Also, the official web platform for residents to report problems, request services, and access information was included. It was observed that roughly 2 million service request cases were collected from 2007 to 2022, varying in their nature. Since the proposed research work focuses on road services requests, therefore after further cleaning of the data and filtering out the unwanted data i.e., not belonging to our required criteria. It was scaled down to approximately 67,000 requests.

Table 1 represents the past six years(2015-2020) of data posted on the open data KC website, in each year the number of requests received by different channels is mentioned. After carefully analyzing the data, there were many trends observed. For instance, between 2015 to 2018, requests submitted by phone outnumbered the online submitted requests by almost two folds. While, in 2019 and 2020, online submissions suddenly became the most frequently used reporting method surpassing the other methods. In the same years, it was observed that the volume of pothole reports quintupled from previous years. As illustrated in Figure 1, from 2007 through 2018, pothole service requests remained steady between 2,000 and 4,000, but in 2019 the number of requests jumped to 20,000. One plausible and agreed-upon reason for the spike can be attributed to the convenience of submitting online requests. Another reason which is quite

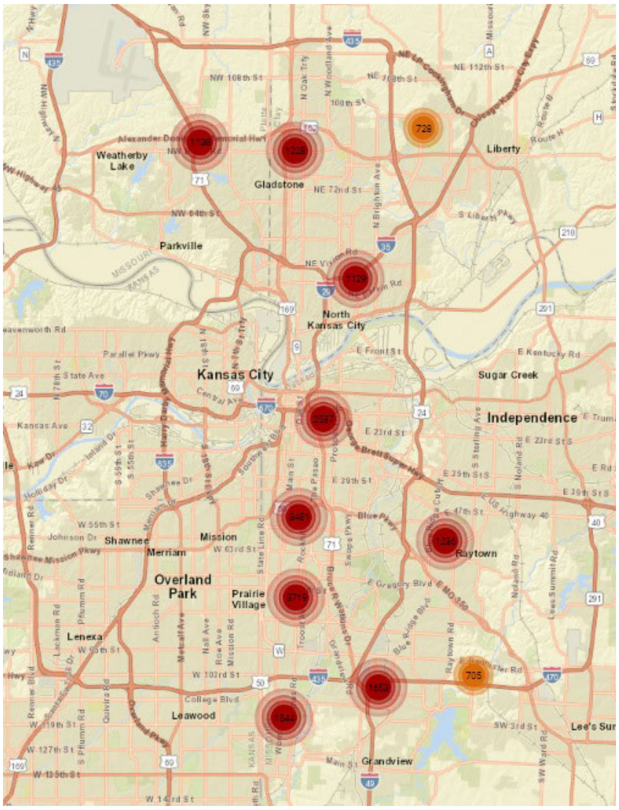


FIGURE 7. Potholes reported in 2019. source: OpenDataKC.

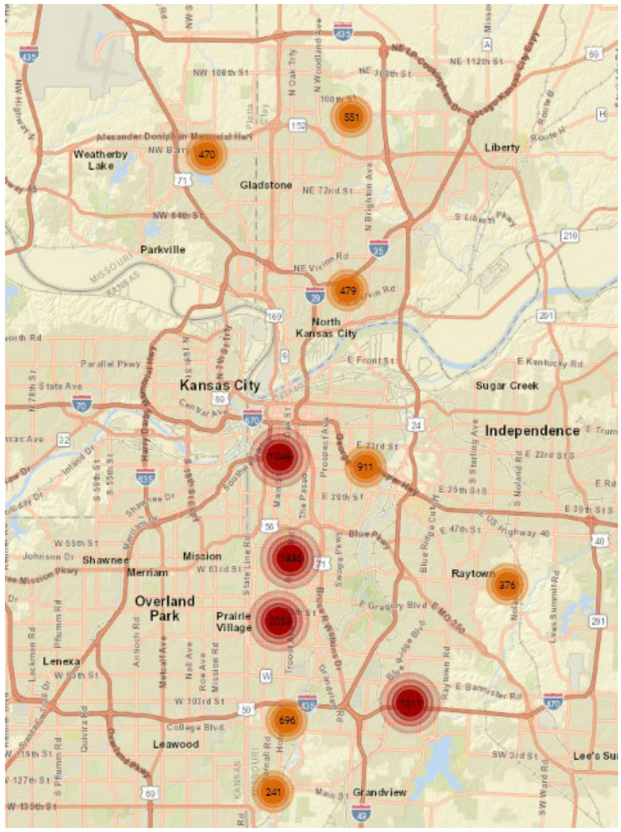


FIGURE 8. Potholes reported in 2020. source: OpenDataKC.

acceptable is due to increased public familiarity with Open Data KC.

It has been observed from Figure 2 that, in January 2020, around 2,500 pothole repair requests were submitted, which is more than twice the amount of requests submitted for the same month in 2019. From February steady growth has been observed, however. In 2020, contrary to the ongoing trend, the number of reported cases has dropped greatly from March. For the same month, the requests in 2019 were reaching 3500, while in 2020 it dropped to less than 1000. This dramatic drop in pothole reports correlates with the massive COVID-19 global lockdown. Which caused reduced traffic on the roads and hence reduced reports. These reduced reports continued until the gradual return started in early 2021 as seen in Figure 1, and normal report requests resumed as usual in Figure 3 for the year 2022. Another interesting pattern found that most pothole reports occur in springtime, as can be observed in Figure 4, where the requests are categorized in groups. Figure 4 breaks down the pothole requests for 14 years by season (winter, spring, summer, and fall). There is a strong reason for this spike in springtime which can be attributed to the freeze-thaw cycle, distressing the pavement during winter precipitation. On the other hand, the fall season has fewer pothole request services. This trend is repeated in all studied years. Figures 5 & 6 A1 & B1 show the distribution of reported service requests in Spring 2016 and Fall 2016.

Each pothole service request from the data has been mapped over Kansas City in order to view pothole distribution patterns by calendar year. Additionally, pothole frequencies are broken into downtown and residential areas and are reflected. Figures 7 & 8 show the marker clustering maps for 2019 and 2020. Clusters of potholes are noted in most of the areas. The Fall and Spring seasons of 2016, as shown in Figures 5 & 6 A1 & B1, reveal similar patterns of pothole clustering, regardless of the sparse total numbers of potholes for that year.

V. HICARE APPROACH

After carefully analyzing the aforementioned data and analyzing it carefully, we designed and developed a multi-layered heuristic algorithm using the pothole clustering inspired by the Hybrid SA (Simulated Annealing) and LK (Lin-Kernighan heuristic) heuristic algorithms. It is worth mentioning here that presenting all the studied patterns and observations was out of the scope of this paper, therefore few of the observations in the paper have been defined.

A. CLUSTERING

In order to build clusters for a given data set, we adapted the k-means clustering algorithm. The algorithm first randomly selects addresses as data points, afterwards, it uses this to form initial estimates for the centroids. It then iterates between these two steps, the data assignment and updating the

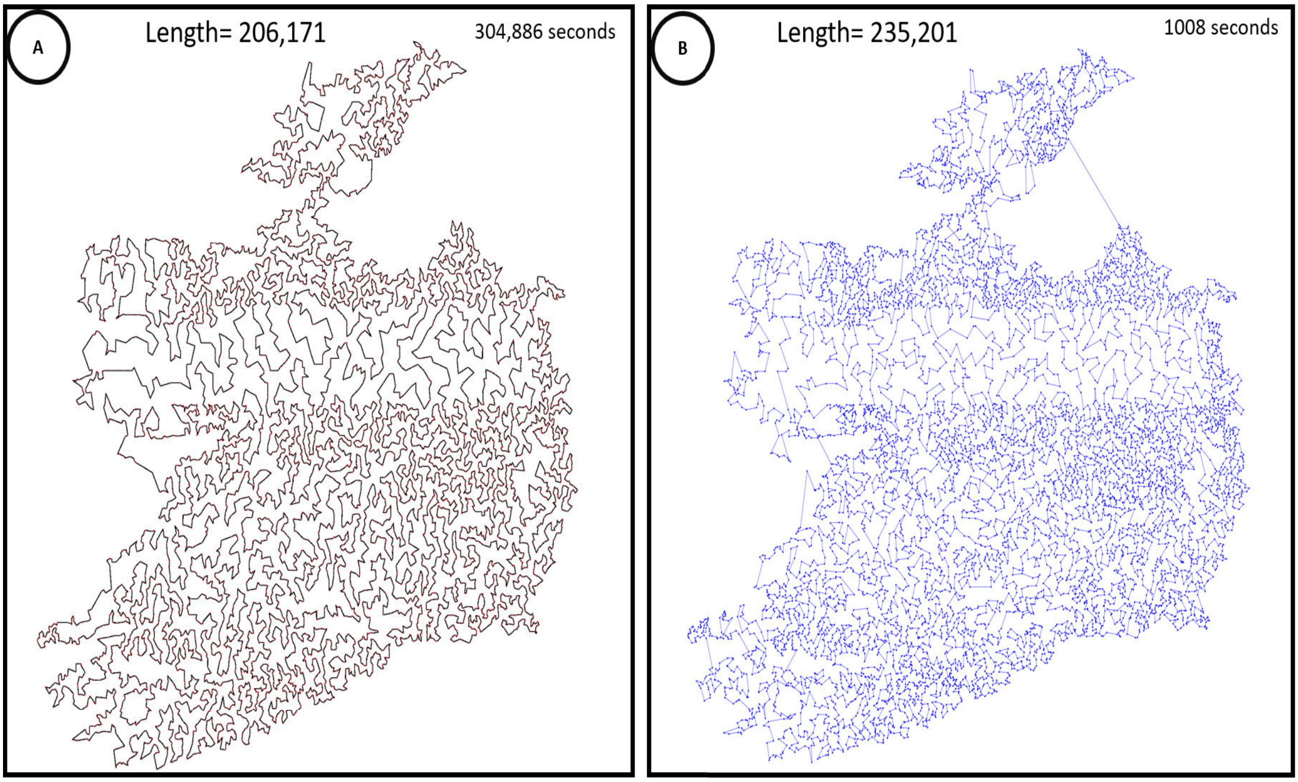


FIGURE 9. A: best route so far. B: HiCARE.

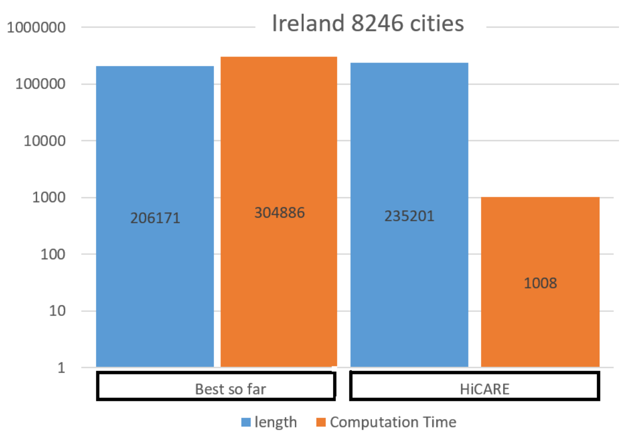


FIGURE 10. Ireland 8246 cities.

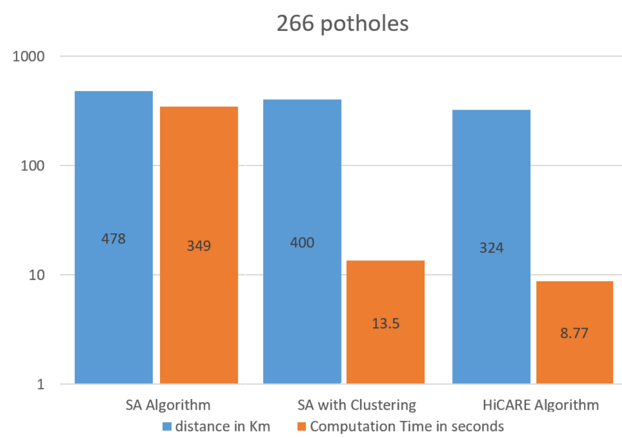


FIGURE 11. SA: Simulated annealing heuristic hicare: hierarchical cluster-based heuristic.

centroids steps, according to the new data [26]. During data assignment, each address is assigned to the nearest centroid according to the squared Euclidean distance between two points in the plane. Which are calculated from coordinates latitude, longitude, height, and width. Centroids are then recalculated using the mean of all data points in that centroid's cluster during the centroid updating steps.

These two steps are repeated exhaustively until no changes occur to the clusters in the centroid updating step. This is achieved by either minimizing the sum of the distances or reaching the maximum number of times the steps can be repeated.

B. OPTIMIZATION AND COSTS

The Algorithm 1 aims to compute an optimized travel route that involves visiting each node or cluster exactly once. To accomplish this, a hierarchical clustering approach is utilized to identify the optimal route between clusters on the top level. The HiCARE algorithm is used to visit each node once and return to the starting cluster, which allows for the

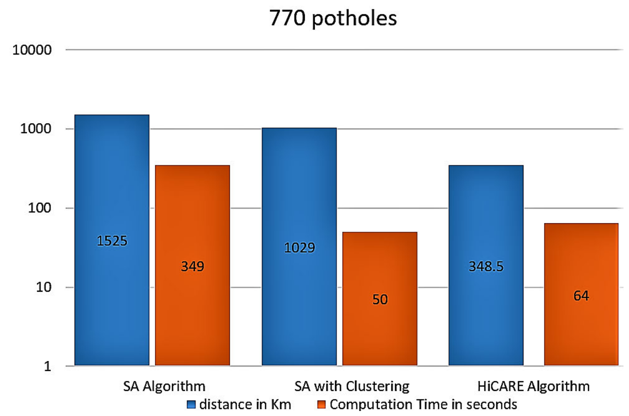


FIGURE 12. SA: simulated annealing heuristic hiCARE: Hierarchical cluster-based heuristic.

Algorithm 1 HiCARE Algorithm

Input: road service requests long lat
Output: recommended route
apply k-means on points ▷ centroids= centers of clusters
apply SA Heuristic Algorithm on centroids
C[i]= points of each cluster i
for i in clusters **do**
  if C[i] > Max **then**:
    apply k-means on C[i]
    calculate optimal route for new centroids
    update centroids accordingly
    update C[] accordingly
  end if
  if C[i] > 1 **then**:
    start= nearest of C[i] to C[i−1][−1]
    C[i][0]=start
    end= nearest of C[i] to centroids[i+1]
    C[i][−1]=end
    C[i][1:−1]= SA Heuristic Algorithm on (C[1:−1])
  end if
  route.append(C[i])
  apply SA Heuristic Algorithm on route
end for
return (route)

calculation of the optimal route between clusters on the top level.

Subsequently, an enhanced heuristic SA (Simulated Annealing) algorithm is applied within each cluster to determine the most efficient route between addresses in that particular cluster. This involves identifying the starting and ending points in each cluster based on the nodes of surrounding clusters. The SA algorithm is then used to generate the optimal path from the starting node to the ending node within each cluster, thereby reducing the computational complexity from [n!] to [(n−2)!] within each cluster on any level. Although the number of iterations may not be maximized at all times due to a threshold level, this reduction in the number of iterations enhances the efficiency of the algorithm in identifying the best route between the start and end points in each cluster.

Finally, the optimal routes from each cluster are merged based on the hierarchical structure of the clustering generated in the upper level. The LK algorithm is applied (with a threshold) to prevent local minima and avoid any crossing links in the final route. The algorithm is governed by the following:

$$Initialsizeofclusters = \frac{potholes}{Avg} \tag{1}$$

Avg: is the average cluster size set by the system admin

Max: is the maximum cluster size set by the system admin. Any cluster that exceeds the maximum will be sub-clustered.

C. ALGORITHMIC FLOW

In this section, we break down the algorithm steps for a better understanding of the flow of the algorithm. The MAX and AVG as discussed earlier are provided by the system admin.

1) First the system administrator has to set the average and maximum size of points for clusters(C).
2) Next, the points are clustered according to the sizes specified in Step 1 and form the top-level cluster.
3) The top level is processed by the hybrid algorithm i.e., SA and LKH algorithms.
4) Once we get the clusters at the top level, they are processed according to the rules provided in the sub-items.
   a) If the cluster(C_i) size is less than the maximum (MAX) size
   b) Then the starting point for the cluster is chosen.
   c) The criteria for choosing the starting point for the current cluster is; based on the distance from the endpoint of the previous cluster ($C_i - 1$).
   d) Similarly, the criteria to select the endpoint for the cluster(C_i) is; based on the distance from the center of the next cluster i.e., ($C_i + 1$))
   e) Next, the hybrid algorithm is applied to the points inside the cluster.
5) On the other hand, If the cluster(C_i) size is greater than the maximum (MAX) size.
6) The system needs to restructure from the beginning and repeat step 4.

D. HIERARCHICAL CLUSTERING STRUCTURE

Determining the optimal initial size of clusters at each level is crucial in improving the computational performance and distance accuracy of the algorithm, also it affects the number of iterations to find the optimal route. The maximum computation time required for finding the optimal path in a hierarchical clustering structure can be represented by Equation 1. After experimentation, we observed that it is more efficient to limit the size of lower-level clusters to no more than the number of clusters at the top-level by 2 e.g.,($No_{o}f_{t}op_{l}evel_{c}lusters$-2). This is because our algorithm

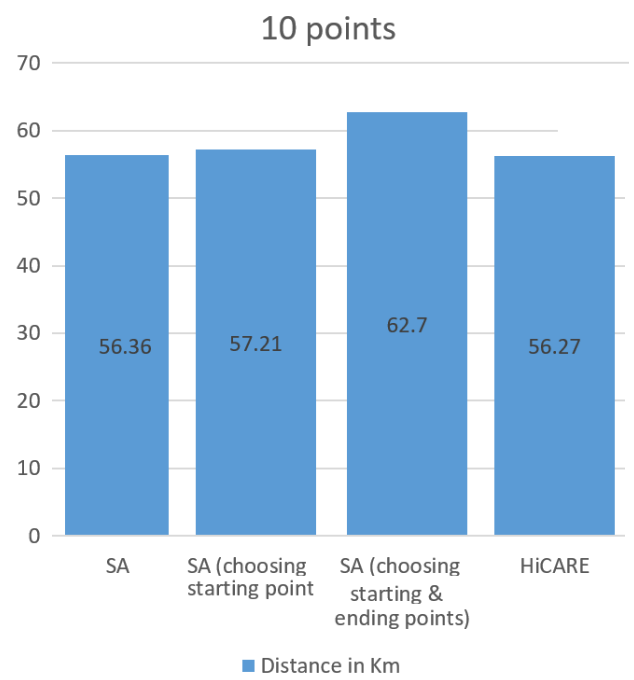


FIGURE 13. **Distance computation for 10 points.**

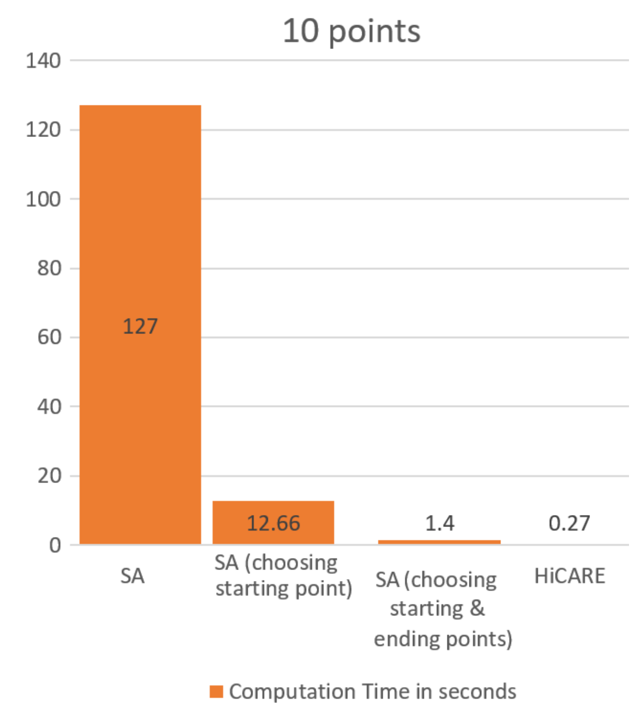


FIGURE 14. **Time computation for 10 points.**

reduces the cost of the lower levels by 2 points when selecting the starting and ending points. For instance, if the size of the lower level cluster is 6, it would require 720 iterations to find the optimal path, which is the result of (6)! while our algorithm makes it only 24 which is the result of (6-2)! In this study, we examined three different scenarios for clustering three groups, each consisting of 45 points. To achieve this, we employed a two-layer clustering approach, with 6 clusters on the top-level and up to 8 points in each cluster for the first group, 7 clusters on the top-level and up to 7 points in each cluster for the second group, and 5 clusters on the top-level and up to 9 points in each cluster for the third group. By doing so, we covered all scenarios where the size of the higher-level cluster is greater than, equal to, or less than the sizes of the clusters in the lower level.

Next, we applied the algorithm equation to each of these three scenarios to determine the number of iterations required to find the optimal route. Our results showed that it would take up to 5136, 5978, and 25410 iterations to find the optimal route for the three groups, respectively. This translates to computation times of up to 0.28, 0.33, and 1.41 seconds, respectively. The third group had the highest number of iterations because the cluster size on the lower-level exceeded the cluster's size on the top-level by more than 2 points.

We further investigated the optimal size of the hierarchical clustering structure for our algorithm. Our findings revealed that the most efficient hierarchical clustering size is to have the size of the lower-level cluster greater than that of the higher-level by precisely 2 points (in the two layers of clusters). If this was not possible, then the next best approach is to ensure that the size of the lower-level cluster does not exceed the size of the higher-level cluster by more than 2 points in any way.

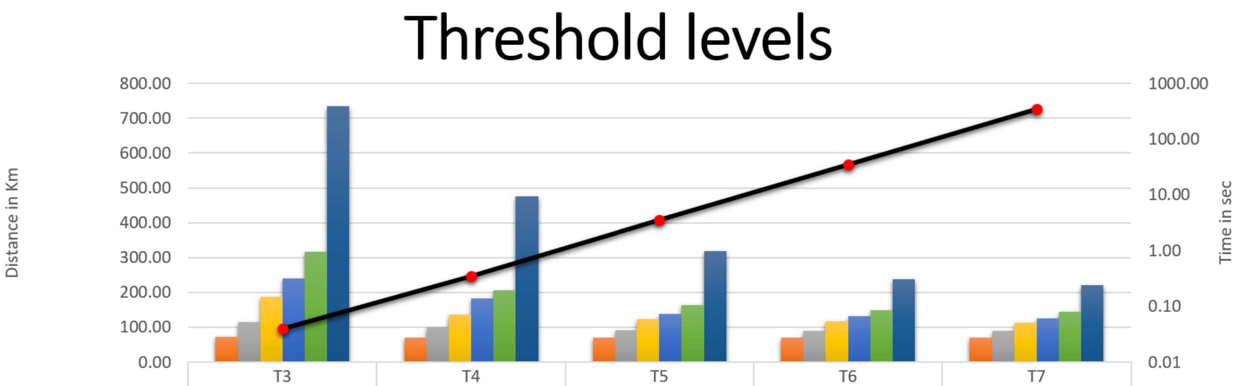
Controlling the difference in cluster sizes between levels can be challenging since the HiCARE clustering process is based on the distance between points, resulting in varying cluster sizes. Some clusters may have significantly more points than others. To address this issue, HiCARE clustering incorporates a sub-clustering approach to reduce the number of clusters that exceed a predefined maximum size, rather than increasing the number of clusters at the top level. In cases where the maximum cluster size is large, HiCARE implements a threshold for iterations to handle oversized clusters efficiently. By leveraging sub-clustering when possible and setting a threshold when necessary, HiCARE can achieve optimal results in clustering.

E. THRESHOLD

HiCARE is a heuristic algorithm, meaning it is designed to quickly find an optimized route rather than running indefinitely. To prevent HiCARE from running indefinitely, a threshold is set. The algorithm will stop searching for a better sub-path in a cluster if it hits the threshold. Determining the best threshold value is an important consideration. To identify an appropriate threshold value, we conducted several experiments on groups containing 10, 20, 30, 40, 50, and 100 points. The results of these experiments are presented in Figure 15 and will be discussed in detail in Section VI.

VI. EVALUATION

In this section, we provide the outcomes of the experiments conducted to evaluate the HiCARE algorithm's performance



	T3	T4	T5	T6	T7
10 points	71.65	70.71	70.68	70.16	70.12
20 points	113.99	99.46	90.84	90.58	90.32
30 points	186.94	136.82	122.51	116.79	113.06
40 points	239.52	183.79	138.14	131.59	126.67
50 points	317.53	207.24	164.09	149.30	144.13
100 points	733.77	475.75	319.30	239.14	220.31
Computation Time	0.04	0.35	3.50	35.00	349.00

FIGURE 15. Results of threshold levels.

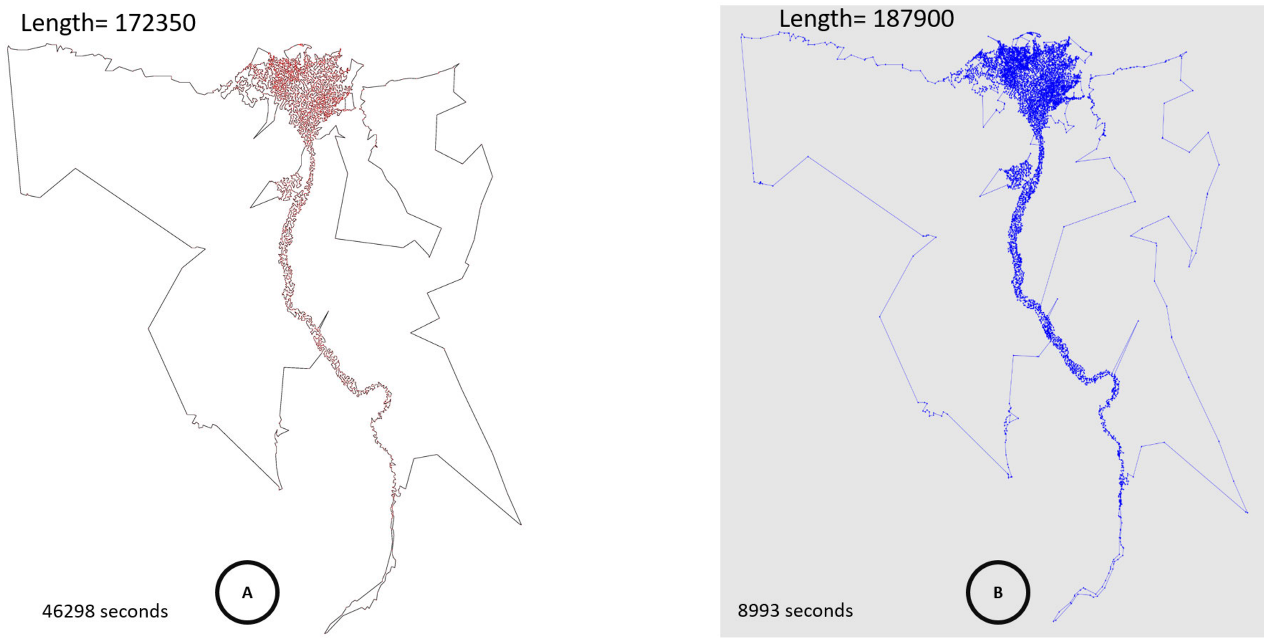


FIGURE 16. Egypt: A: Best route reached so far. B: HiCARE.

regarding travel distance and scheduling time. We measure these parameters against the SA algorithm, which is another TSP heuristic algorithm that uses a threshold of 10^7. We also compared the clustering results of SA to HiCARE.

A. MEDIUM-SIZED AND LARGE-SIZED CLUSTERS

The experimental analysis for different algorithms and potholes are mentioned below ranging from Spring to Fall of the year 2016:

- Figure 12 displays the total traveling distance for 770 potholes reported during Spring 2016.
- The HiCARE algorithm has a traveling range of 348.5 km, which is over four times better than applying the SA algorithm with 10^7 iterations.
- The SA algorithm with 10 million iterations for 770 nodes takes 349 seconds, which is over five times higher in computational time than the HiCARE algorithm (64 seconds) for the same node size.

- HiCARE algorithm takes 14 seconds longer than SA with Clustering in computation time, but it is three times better in terms of distance calculation.
- Figure 11 shows the total traveling distance with 266 potholes reported during Fall 2016.
- HiCARE algorithm has a traveling range of 324 km, which is 47.5% better in terms of distance than applying SA with a threshold of 10^7.
- HiCARE is also 19% better than SA with Clustering in terms of distance.
- SA algorithm with 10 million iterations for 266 nodes takes 349 seconds, and SA with Clustering takes 13.5 seconds.
- The HiCARE algorithm takes 8.77 seconds for the same node size.

Based on the experimental results, it can be concluded that clustering offers a significant improvement in computational complexity and traveling distance compared to SA. Nevertheless, the HiCARE algorithm outperforms SA with clustering. It is worth noting that evaluating the impact of HiCARE on large nodes with hundreds of clusters may not be practically feasible for daily pothole maintenance route calculation. Therefore, we conducted an evaluation of HiCARE and the other algorithms on small-sized clusters, which represents a typical scenario for each cluster.

B. SMALL-SIZED CLUSTERS

In this study, we applied the HiCARE algorithm to multiple smaller-scale scenarios involving ten random nodes with no cluster. Our aim was to investigate the effect of reducing the number of points on the computation time and route distance calculation. To this end, we compared the performance of HiCARE with the simulated annealing (SA) algorithm, which was applied to three groups of ten points each, with one additional group i.e, the fourth group with reduced points. The first group consisted of ten random points, while the second group included ten points with a specified starting point. The third group comprised ten points with both starting and ending points specified.

We present the results of our experiment in Figure 13, which shows the average traveling distances for the four groups. The HiCARE algorithm's traveling range of 56.27 km was found to be better than the SA algorithm's variations. Additionally, HiCARE outperformed SA in terms of computation time, taking only 0.27 seconds for a 10-point cluster with the best route calculation. In contrast, the SA computation time ranged from 1.4 to 127 seconds.

Figure 14 shows the computation time for the four groups. We found that reducing the number of nodes, even by one or two points, resulted in a significant improvement in computation time. Specifically, the computation time for the group with ten random points without specifying starting or ending points was 127 seconds. However, the second group with a specified starting point took only 12.66 seconds, and the third group with both starting and ending points specified took 1.4 seconds.

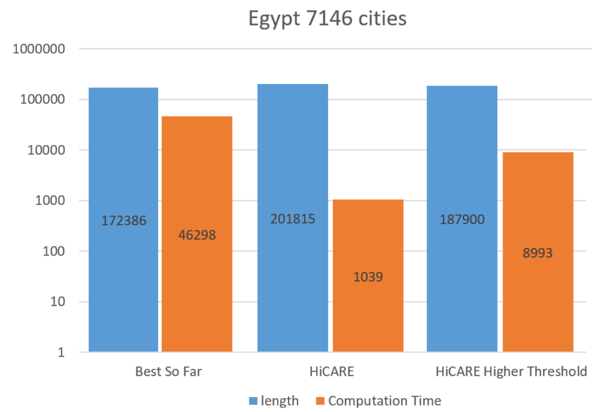


FIGURE 17. Egypt: 7146 cities.

In conclusion, our experiment demonstrates that the HiCARE algorithm outperforms the SA algorithm in terms of computation time and traveling distance, even in smaller-scale scenarios involving ten random nodes with no cluster. Furthermore, reducing the number of nodes by one or two points results in a significant improvement in computation time. These findings provide valuable insights for researchers and practitioners interested in using clustering algorithms for route optimization in transportation and logistics.

C. NATIONAL TRAVELING SALESMAN PROBLEMS

The HiCARE algorithm has demonstrated remarkable results in solving National Traveling Salesman Problems (TSP) with a large number of nodes. In this study, we present the results of HiCARE compared to the best-known solutions thus far.

The results of the HiCARE algorithm on the Ireland TSP with 8246 cities are shown in Figures 9 and 10. We observe that HiCARE generated a route within 1008 seconds, which is 14% of the best route achieved in 304,886 seconds.

Similarly, Figures 16 and 17 show the results of the HiCARE algorithm on the Egypt TSP with 7146 cities. The algorithm generated an optimized route within 19% of the best solution with only a fraction of that time.

In addition, we investigated HiCARE's performance on a relatively smaller-scale problem to Ireland and Egypt scenario, the Oman TSP with 1979 cities. The results are shown in Figures 18 and 19. HiCARE produced an optimized route within 9% of the best-known solution in only 152 seconds, which is a mere 1.5% of the time taken by the best-known solution.

Overall, the results demonstrate that HiCARE is a promising algorithm for solving large-scale TSP problems efficiently and effectively. Its ability to achieve near-optimal solutions in significantly less time than the best-known solutions so far makes it a valuable tool for optimizing TSPs in various real-world applications.

D. THRESHOLD LEVELS

In this study, we investigate the effect of different levels of thresholds on the Simulated Annealing (SA) phase

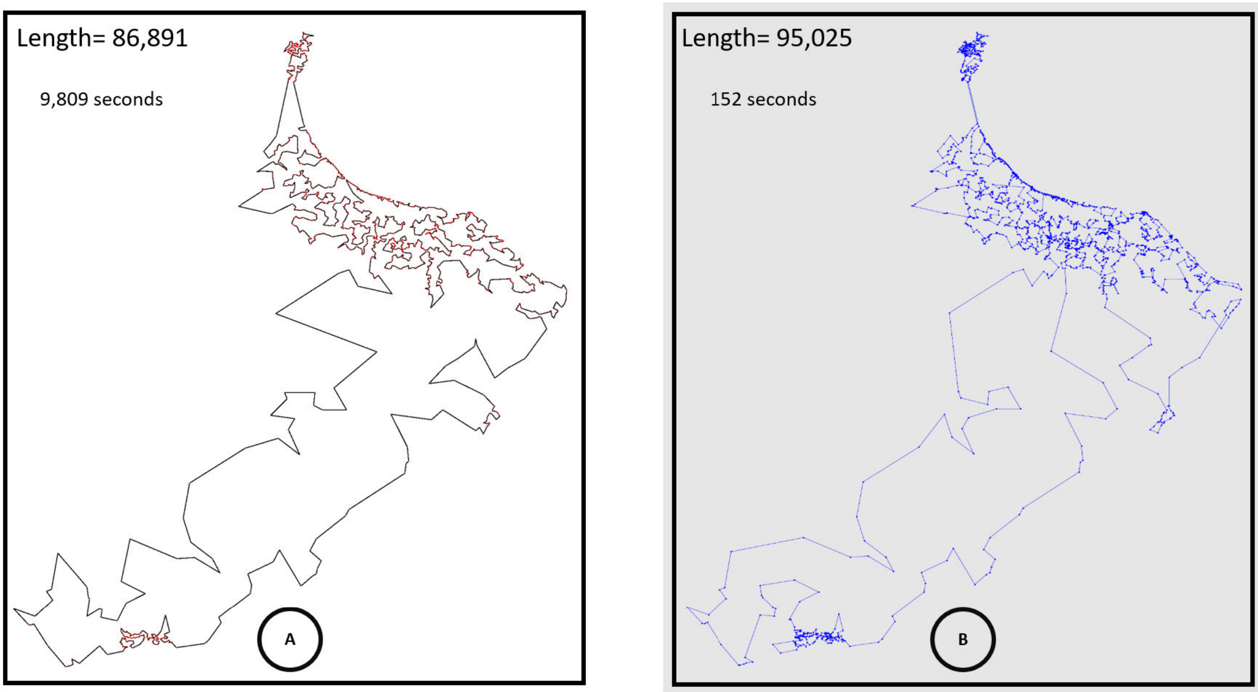


FIGURE 18. Oman: A: Best route reach so far. B: HiCARE.

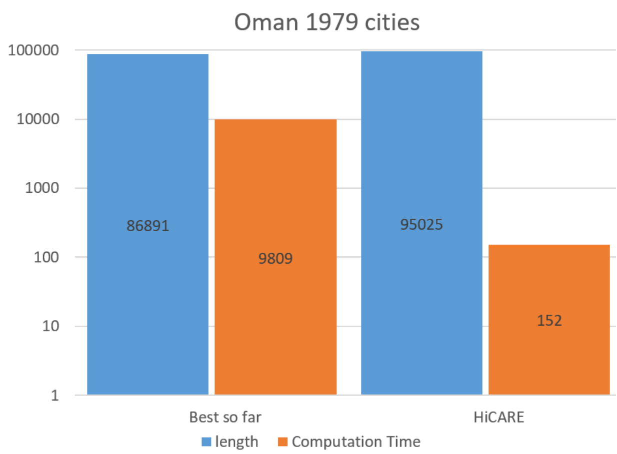


FIGURE 19. Oman: 1979 cities.

of the HiCARE algorithm, which is used to solve the Traveling Salesman Problem (TSP) on random point clusters. We generated 10 random sets for each group and calculated the mean of each group to ensure statistical significance. The results, presented in Figure 15, demonstrate that the threshold value has a significant impact on the performance of the algorithm.

For a 10-point cluster, we found that a threshold of (10^3) is optimal, as there was no noticeable improvement in the route beyond that threshold. As for larger clusters, a threshold of (10^5) generated a good route for 20 to 50-point clusters within 5 seconds. For 30 to 50-point clusters, a threshold of (10^6) could compute 5% better routes with a cost of 52 seconds. For a 100-point cluster, a threshold of (10^6) was reasonable, while a threshold of (10^7) could compute a 10% better route with the cost of 523 seconds. Choosing the appropriate threshold level depends on several factors, including the size and structure of the clusters and the computational resources available. Overall, our findings indicate that selecting an appropriate threshold value can significantly affect the performance of the algorithm in solving TSP problems.

VII. CONCLUSION

In this research work, we highlighted the problems of potholes for vehicles and maintenance teams alike. It was observed that potholes cause monitor damage to vehicle owners, and also it is quite costly to repair. Furthermore, assigning teams for the pothole repair work requires an elaborate plan, and to the best of our knowledge, there was no such system in existence that had tackled the issues as discussed in this paper. The proposed framework optimizes route scheduling necessary for the maintenance team to save time and resources. The past 14 years of pothole data were analyzed and collected from different channels for Kansas City. After observations and studying different patterns from the data, a multi-layered heuristic algorithm which is a Hybrid Algorithm of SA (Simulated Annealing) and LK (Lin-Kernighan heuristic) heuristic algorithm, was designed and developed, called Hierarchical Cluster-Based Heuristic (HiCARE) Algorithm. HiCARE was tested in comparison with SA Algorithm to real-life cases of potholes from OpenDataKC. Furthermore, it was also applied to National Travelling Salesman Problem with large-scale nodes. For instance, for 8246 points problem, HiCARE generated a route within 1008 seconds, compared to 304886 seconds by previous algorithms. Experimental results of HiCARE showed significant results in terms of both distance calculation and time computation.

REFERENCES

[1] *Street Preservation Program*. Accessed: May 2022. [Online]. Available: https://www.kcmo.gov/city-hall/departments/public-works/street-preservation-program

[2] D. Applegate, R. Bixby, V. Chvatal, and W. Cook, *The Traveling Salesman Problem: A Computational Study* (Princeton Series in Applied Mathematics) Princeton, NJ, USA: Princeton Univ. Press, 2007.

[3] B. Gibson, C. Wallace, N. Albright, C. Van Dyke, and D. Kreis, *KYTC Maintenance Field Operations Guide Supplement*. Lexington, KY, USA: Univ. Kentucky Transp. Center, 2020.

[4] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Oper. Res.*, vol. 21, no. 2, pp. 498–516, Apr. 1973.

[5] K. Helsgaun, "An effective implementation of the Lin–Kernighan traveling salesman heuristic," *Eur. J. Oper. Res.*, vol. 126, no. 1, pp. 106–130, Oct. 2000.

[6] S. Kirkpatrick, C. Gelatt Jr., and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, Jan. 1983.

[7] R. A. Rutenbar, "Simulated annealing algorithms: An overview," *IEEE Circuits Devices Mag.*, vol. 5, no. 1, pp. 19–26, Jan. 1989.

[8] J. J. Schneider, T. Bukur, and A. Krause, "Traveling salesman problem with clustering," *J. Stat. Phys.*, vol. 141, no. 5, pp. 767–784, Dec. 2010, doi: 10.1007/s10955-010-0080-z.

[9] C. Wu, Z. Wang, S. Hu, J. Lepine, X. Na, D. Ainalis, and M. Stettler, "An automated machine-learning approach for road pothole detection using smartphone sensor data," *Sensors*, vol. 20, no. 19, p. 5564, Sep. 2020.

[10] A. Basavaraju, J. Du, F. Zhou, and J. Ji, "A machine learning approach to road surface anomaly assessment using smartphone sensors," *IEEE Sensors J.*, vol. 20, no. 5, pp. 2635–2647, Mar. 2020.

[11] B. Varona, A. Monteserin, and A. Teyseyre, "A deep learning approach to automatic road surface monitoring and pothole detection," *Pers. Ubiquitous Comput.*, vol. 24, no. 4, pp. 519–534, Aug. 2020.

[12] G. Baldini, R. Giuliani, and F. Geib, "On the application of time frequency convolutional neural networks to road anomalies' identification with accelerometers and gyroscopes," *Sensors*, vol. 20, no. 22, p. 6425, Nov. 2020.

[13] D. Luo, J. Lu, and G. Guo, "Road anomaly detection through deep learning approaches," *IEEE Access*, vol. 8, pp. 117390–117404, 2020.

[14] E. A. Martinez-Ríos, M. R. Bustamante-Bello, and L. A. Arce-Sáenz, "A review of road surface anomaly detection and classification systems based on vibration-based techniques," *Appl. Sci.*, vol. 12, no. 19, p. 9413, Sep. 2022.

[15] S.-S. Liu, A. Budiwirawan, M. F. A. Arifin, W. T. Chen, and Y.-H. Huang, "Optimization model for the pavement pothole repair problem considering consumable resources," *Symmetry*, vol. 13, no. 3, p. 364, Feb. 2021.

[16] T. Lindblom and O. Tollin, "Optimization and Bayesian modeling of road distance for inventory of potholes in Gvle municipality," Dept. Statist., Faculty Social Sci., Stockholm Univ., Stockholm, Sweden, Tech. Rep. diva2: 1634588, 2022.

[17] N. Hassan, I. Siddiqui, S. Mazhar, and H. Hameed, "Road anomaly classification for low-cost road maintenance and route quality maps," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops*, Mar. 2019, pp. 645–650.

[18] G. Gutin and A. Punnen, *The Traveling Salesman Problem and Its Variations*. New York, NY, USA: Springer, 2002. [Online]. Available: https://books.google.com/books?id5RuNx2TIbIcC

[19] Z. H. Ahmed, "The ordered clustered travelling salesman problem: A hybrid genetic algorithm," *Sci. World J.*, vol. 2014, pp. 1–13, Jan. 2014, doi: 10.1155/2014/258207.

[20] W. M. Springer, "Review of the traveling salesman problem: A computational study by Applegate, Bixby, Chvátal, and Cook (Princeton University Press)," *ACM SIGACT News*, vol. 40, no. 2, pp. 30–32, Jun. 2009, doi: 10.1145/1556154.1556162.

[21] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *Biosystems*, vol. 43, no. 2, pp. 73–81, Jul. 1997.

[22] *How to Cluster Map Markers: Demo and Tutorial*. Accessed: Sep. 2022. [Online]. Available: http://blog.batchgeo.com/map-clustering-demo-tutorial/

[23] *311 Call Center Service Requests*. Accessed: Feb. 2023. [Online]. Available: https://data.kcmo.org/browse?category=311

[24] *Traveling Salesman Problem*. Accessed: Sep. 2022. [Online]. Available: http://mathworld.wolfram.com/TravelingSalesmanProblem.html

[25] *Potholes Cause Problems for KC Area Drivers*. Accessed: Aug. 2022. [Online]. Available: https://www.kshb.com/news/local-news/potholes-cause-problems-for-kc-area-drivers

[26] *Introduction to K-Means Clustering*. Accessed: Aug. 2022. [Online]. Available: https://www.datascience.com/blog/k-means-clustering

[27] *Kansas City Winter Weather Statistics Page*. Accessed: Aug. 2022. [Online]. Available: https://www.weather.gov/eax/kcwinterstats

[28] *Records Broken or Tied in 2007*. Accessed: 2022. [Online]. Available: http://cdn.trb.tv/wdaf-ftp/almanac/wx2007.html

[29] *Batch Reverse Geocoding*. Accessed: 2022. [Online]. Available: https://www.doogal.co.uk/BatchReverseGeocoding.php

[30] *What is MTTR?* Accessed: 2022. [Online]. Available: https://www.fiixsoftware.com/mean-time-to-repair-maintenance/

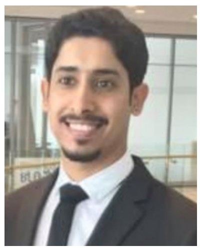
[31] S. E. Schaeffer, "Graph clustering," *Comput. Sci. Rev.*, vol. 1, no. 1, pp. 27–64, Aug. 2007, doi: 10.1016/j.cosrev.2007.05.001.

[32] M. H. Erol and F. Bulut, "Real-time application of travelling salesman problem using Google maps API," in *Proc. Electr. Electron., Comput. Sci., Biomed. Eng.' Meeting (EBBT)*, Apr. 2017, pp. 1–5.

[33] R. Saxena, M. Jain, S. Bhadri, and S. Khemka, "Parallelizing GA based heuristic approach for TSP over CUDA and OPENMP," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2017, pp. 1934–1940.

[34] R. Bazylevych, R. Kutelmakh, R. Dupas, and L. Bazylevych, "Decomposition algorithms for large-scale clustered TSP," in *Proc. IICAI*, 2007, pp. 255–267.

[35] G. Lu, Y. Xiong, C. Ding, and Y. Wang, "An optimal schedule for urban road network repair based on the greedy algorithm," *PLoS ONE*, vol. 11, no. 10, Oct. 2016, Art. no. e0164780.



SAMI ALSHAMMARI received the B.S. degree in information technology from Northern Border University, Rafha, Saudi Arabia, in 2013, and the M.S. degree in computer science from the University of Missouri–Kansas City, Kansas City, MO, USA, in 2017, where he is currently pursuing the Ph.D. degree in telecommunication and computer networking. He is a Lecturer with the Faculty of Computing and Information Technology, Northern Border University. His research interests include smart cities, machine learning, and the IoT technologies.

SEJUN SONG (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science and engineering from the University of Minnesota, Twin Cities, USA, in 1999 and 2001, respectively. He is currently an Associate Professor with the Department of Computer Science and Electrical Engineering, University of Missouri–Kansas City. He and his research team have conducted research in the areas of trustworthy information and computing systems, and software, such as resilient network and system management, software-defined networks, cloud computing auditability, mobile cloud computing, security, high availability, data storage, and embedded systems.

BAEK-YOUNG CHOI (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from the University of Minnesota, Twin Cities. She has been a Faculty Fellow of the National Aeronautics and Space Administration (NASA), the U.S. Air Force Research Laboratory's Visiting Faculty Research Program (AFRL-VFRP), and the Korea Telecom's-Advance Institute of Technology (KT-AIT). She is currently a Professor with the Department of Computer Science and Electrical Engineering, University of Missouri–Kansas City. Her research interests include networking and communications, with a specific emphasis on the Internet of Things, software-defined networking, cybersecurity, and smart city technologies.

• • •