An Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

ELHAM GHAFFARI[1], **AMIR MASOUD RAHMANI**[2], **MORTEZA SABERIKAMARPOSHTI**[3], **AND AMIR SAHAFI**[3]

[1]Department of Computer Engineering, Qeshm Branch, Islamic Azad University, Qeshm 1468763785, Iran
[2]Future Technology Research Center, National Yunlin University of Science and Technology, Douliou, Yunlin 64002, Taiwan
[3]Department of Computer Engineering, Islamic Azad University South Tehran Branch, Tehran 1468763785, Iran

Corresponding author: Amir Masoud Rahmani (rahmani@srbiau.ac.ir)

ABSTRACT Finding the shortest and cleanest path in the cities is vital, especially in metropolises. Although several algorithms and some software have been introduced to manage the traffic or suggest a path with minimum traffic congestion, none considers air quality a deciding factor. This paper introduces a novel algorithm to find the shortest path based on traffic congestion and air quality. In the proposed algorithm, the city map is fetched from the Google Map app and is converted into a weighted graph. Traffic data is collected from GPS devices, which will be available through the local cloud services. The C-means clustering method is used to cluster traffic congestion. Also, the air quality information is collected from air pollution monitoring stations. The graph weights are calculated based on both air quality and traffic congestion factors, simultaneously. Finding the shortest path problem is then defined as an optimization problem, and the linear programming method is used to solve it. Finally, the proposed algorithm's performance is evaluated by finding the shortest path in Tehran, Iran in different scenarios.

INDEX TERMS Intelligent transport system, air pollution, traffic congestion, C-means clustering.

I. INTRODUCTION

Traffic congestion is one of the most critical challenges in many cities, especially in metropolises. Nowadays, people lose a lot of their time in traffic congestion. Traffic congestion also has negative impacts on public health and causes economic losses. Therefore, governments utilize different strategies for controlling traffic [1]. Although infrastructure development can reduce traffic congestion, smart traffic control methods have received much attention in recent years due to economic and environmental constraints. Existing infrastructures are used optimally in these methods by scheduling traffic and encouraging people to use alternative paths [2].

The Internet of things (IoT) gives dramatically opportunity to manage traffic congestion. The development of IoT has led to the development of powerful Intelligent Transportation System (ITS) algorithms. ITS aims to develop services relating to traffic management and enable users to be better informed and make better decisions about their transport [3].

The associate editor coordinating the review of this manuscript and approving it for publication was Maurice J. Khabbaz.

In the ITS, vehicles' data such as speed, direction, location, etc., are shared from fogs and clouds, and intelligent systems use this data for modeling traffic flow and provide the best choices for drivers [4]. ITSs should make the best decision in real-time and be flexible to make accurate decisions in all situations [5]. Another critical challenge in metropolises is air pollution. Air pollution kills thousands of people and costs the economy billions of dollars. Air pollution is hazardous for many people, such as older adults and patients [6]. Air pollutant agents were studied, and air pollution monitoring stations monitor air quality based on these agents. We know traffic congestion plays an essential role in producing air pollution [7]; therefore, it is necessary to manage traffic congestion and air pollution, simultaneously. Investigation of air pollution and traffic congestion has received less attention in finding the shortest path problems. These problems aim to calculate the optimal path that has the lowest traffic congestion and does not pass through more polluted areas as much as possible. Therefore, we investigate this deficiency and introduce a novel method for finding the shortest path by considering traffic congestion and air quality. In the proposed

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

IEEE*Access*

method, the map is converted into a weighted graph by using image-processing methods. To obtain the graph's weights, traffic congestion is clustered by the C-means algorithm, and the traffic density is calculated using the convex hull algorithm. Greenshield's model estimates the weights at each cluster, and then weights are updated based on the air quality. In the proposed algorithm, weights are inversely related to the vehicle's velocity, and the region with lower vehicles' density has low weights and vice versa. The region with good air quality also has lower weights, and the region with hazardous air quality has higher weights. Finally, the problem is formulated as an optimization problem, and a linear programming method is used to solve it. The rest of the paper is organized as follows; related works are presented in section 2. The proposed model is introduced in section 3. Traffic and air pollution clustering are described in sections 4 and 5, respectively. Linear programming algorithm is introduced in section 6. Simulation results and conclusions are presented in sections 7 and 8, respectively.

II. RELATED WORKS

Due to the importance of air pollution and traffic congestion challenges, several researchers have studied them in ITS. For instance, in [8], researchers investigated the shortest path-finding algorithm in the presence of traffic congestion. In that research, researchers used a K-means clustering algorithm for clustering traffic congestion. Similarly, authors in [9] used K-means clustering algorithms for traffic congestion. Due to fuzzy clustering's advantages, in [10], researchers used the C-means algorithm for clustering traffic congestion. Also, finding the shortest path can be defined as an optimal problem. Several methods can be used to solve this problem. Linear programming is an appropriate algorithm for these types of problems due to its simplicity and accuracy. In [11], researchers used the linear programming method for finding the shortest path in the weighted graph. Researchers in [12] used centralized simulated annealing algorithm to find the optimal path by considering traffic congestion. This algorithm selects the optimal path based on five different factors including the average travel speed of the traffic, vehicles density, roads width, road traffic signals and the roads' length. Data is fetched from sensors and it does not use GPS data. Although that research investigated the effect of intelligent transportation system on CO2 emissions, it did not consider air pollution as a decision criterion. In [13], an intelligent system for routing emergency vehicles was introduced. The aim of that paper is to minimize the delay of emergency vehicles. To achieve this purpose, a traffic management system was proposed by researches. In [14] researchers improved simulated annealing to enhance mobility in smart cites. That paper fetched speed of vehicles and travel times using sensors mounted in the city. Therefore, the implementation of that research method requires the installation of intelligent equipment and sensors that are installed in a limited number of cities. The main advantage of that paper was to develop approach for dynamic calculation of

optimal traffic routes. Air pollution controlling also has been investigated in smart cities. In the paper [15], the authors introduced a method for controlling and reducing air pollution. Researchers investigated air quality in Europe and made suggestions for controlling urban pollutants. Therefore, combining air pollution control and ITS have been noticed in recent years. For example, the integration of traffic congestion and air pollution was investigated in [16]. That paper introduced the computer program for this purpose and did not introduce a mathematical algorithm to find the optimal path. In that paper, traffic congestion and air pollution are categorized into several levels, and the computer program is decided based on the ''if-else'' statements. Several algorithms are used for finding the shortest path. Although nonlinear optimization methods have received a lot of attention in recent years, linear methods are still considered due to their lower computational complexity. Linear programming is a special case of mathematical programming and has lower complexity than the majority of linear programming is a special case of mathematical programming. Also, this method is so accurate, and every local minimum is a global minimum. For instance, researchers in [17] used the mixed-integer linear programming method for finding the optimal path in segmented routing networks. That paper converted the network into a directed graph, and then the problem was defined as an optimization problem. Finally, mixed-integer linear programming was used to find the optimal path. In addition, researchers to solve the network shortest path problem in [18] used the linear programming method. That paper addressed the formulation for implementing a single source, single-destination shortest path algorithm on a quantum annealing computer. Similarly, that paper converted the problem into a directed graph.

III. THE PROPOSED MODEL

The proposed algorithm for predicting the shortest path in the urban area is introduced in this section. This algorithm aims to find the shortest path between source and destination by avoiding traffic congestion and polluted air areas. The following assumptions are considered in this research to develop this algorithm:

1) The city map is fetched from Google Map. Google Map offers satellite imagery, aerial photography, street maps, and real-time traffic conditions [19].
2) The current location of the vehicle is considered as the source, and the destination is determined by the driver. In other words, the driver should determine its destination.
3) The traffic data is collected from GPS receivers are mounted in vehicles or mobile phones. Vehicles can share their data through local cloud services.
4) Air pollution information is collected from sensors are mounted in different places of the city. Similarly, this data can be available through local cloud services.

Our algorithm has several steps that are described as follows:
Step 1: the map is converted to a weighted graph using

IEEE Access

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

image-processing procedures. For this purpose, a black and white image is produced from the map, in which all roads and intersections are shown in white, and other objects are shown in black. Step 2: this black and white image is converted into a neighborhood matrix. Elements in the neighborhood matrix indicate the relationship between neighboring pixels and determine a directly connected road is available or not. The neighborhood matrix is shown by G=(V, E), where V is a set whose elements are called vertices and correspond to intersections in the black and white image, and E are named edges that are equivalent to the roads in the black and white image. The weights are assumed to be one, initially, but they update at the next steps. Step 3: The algorithm only runs when the vehicle is near an intersection. In other cases, there is no need to run the algorithm because it is not possible to select an alternative path. Also, in order to optimize energy consumption in the intelligent system and reduce the load on the local cloud service, the algorithm is executed only at certain time intervals. In fact, the execution time of the algorithm may not be more than a few seconds, and during this time there is no change in traffic and pollution. Therefore, this algorithm can perform routing dynamically and update the optimal route in case of changes in traffic congestion and air pollution. Step 4: the C-means clustering algorithm is used to classify traffic congestion. The clustering algorithm aims to determine traffic congestion and calculate the graph weights. These weights are related to the clusters' congestion. Step 5: the weights are updated by considering air pollution data. Step 6: short path finding algorithm is defined as optimization algorithm and linear programming method is used for finding the shortest path. In this method, the shortest path problem is formulated as an optimization problem. The cost roads are equivalent to the weights that are obtained in the previous sections. The proposed algorithm flowchart is shown in Fig.1. It is observed that the algorithm is repeated until the vehicle reaches its destination. The following flowchart indicates that in each iteration, data about traffic congestion, air pollution, and the source location are updated. As mentioned earlier, the origin is the current position of the car. In this flowchart, the waiting time can be fixed or variable. The waiting time can be calculated according to the average speed of vehicles and the distance between intersections. However, for simplicity, the constant waiting time can be considered constant. For example, the algorithm can be updated every 5 minutes.

IV. TRAFFIC CLUSTERING BY USING THE C-MEANS CLUSTERING ALGORITHM

Nowadays, most smart devices have GPS chips; therefore, real-time GPS can be used in intelligent systems by sharing GPS data between devices through local clouds. Vehicle location, speed, and direction can be estimated using GPS data. Therefore, we assume the traffic information will be available. the clustering purpose is to identify traffic density and prevent traffic congestion roads during routing. In this research, fuzzy clustering is used for this purpose.

C-means is one of the most common fuzzy clustering algorithms that is used in many types of research. This algorithm is very similar to the K-means algorithm [15]. In this algorithm, the input data are clustered in k clusters. This algorithm aims to cluster data to minimize an objective function as follows [10]:

$$arg_c min \sum_{i=}^{n} \sum_{j=1}^{k} \frac{1}{\sum_{\ell=1}^{k} \left(\frac{\|x_i - c_i\|}{\|x_i - c_\ell\|} \right)^{\frac{2}{m-1}}} \|x_i - c_i\|^2 \quad (1)$$

where n is the number of observations, C is the cluster's center's vector, and $C = \{c1, c2, \ldots, ck\}$, k is the number of clusters, and m is the factor to determine the level of the cluster membership and is commonly set to 2.

Similar to the K-means algorithm, the C-means algorithm's performance is related to the number of clusters. There are several methods for calculating the best number of clusters. One of the best and most accurate methods is the Elbow method that is used in this paper. This method is a heuristic used to determine the number of clusters in a data set consisting of plotting the explained variation as a function of the number of clusters and picking the elbow of the curve as the number of clusters to use [21]. After clustering the traffic congestion data, the density of clusters will be calculated using the convex hull method [22]. Greenshield's model is used to determine the vehicle velocity at each cluster. Greenshield is an accurate and simple algorithm that can develop a model of uninterrupted traffic flow that predicts and explains the trends observed in real traffic flows. There are more vehicles in dense areas, and as a result, the speed of vehicles will decrease, and the traffic will be heavier. Based on this algorithm, the average velocity in i-th cluster can be calculated as follows [16]:

$$v_i = v_m \left(1 - \frac{D_i}{D_j} \right) \quad (2)$$

where D_j is a traffic density in jam and v_m is the free-flow speed of vehicles. Di shows the cluster density for i-th cluster. Fig.2 indicates the relationship between the velocity of vehicles and the density of vehicles. In this figure, the density ratio means $\frac{D_i}{D_j}$. It can be seen that by increasing the density of vehicles in an area, traffic congestion increases, and vehicles' speed decreases. The Free-flow speed is a maximum allowable speed for vehicles and maybe variable for each city region.

Now, based on the traffic density in different clusters, the weights for i-th cluster can be calculated as follows:

$$w_i = \frac{\frac{1}{v_i + \sigma}}{\sum_{k=1}^{N} \left(\frac{1}{v_k + \sigma} \right)} \quad (3)$$

where σ is the parameter for avoiding divide-by-zero error and N is the number of clusters. The idea behind this method is that vehicle's velocity in the dense cluster is low. In other words, there is an inverse relationship between a vehicle's
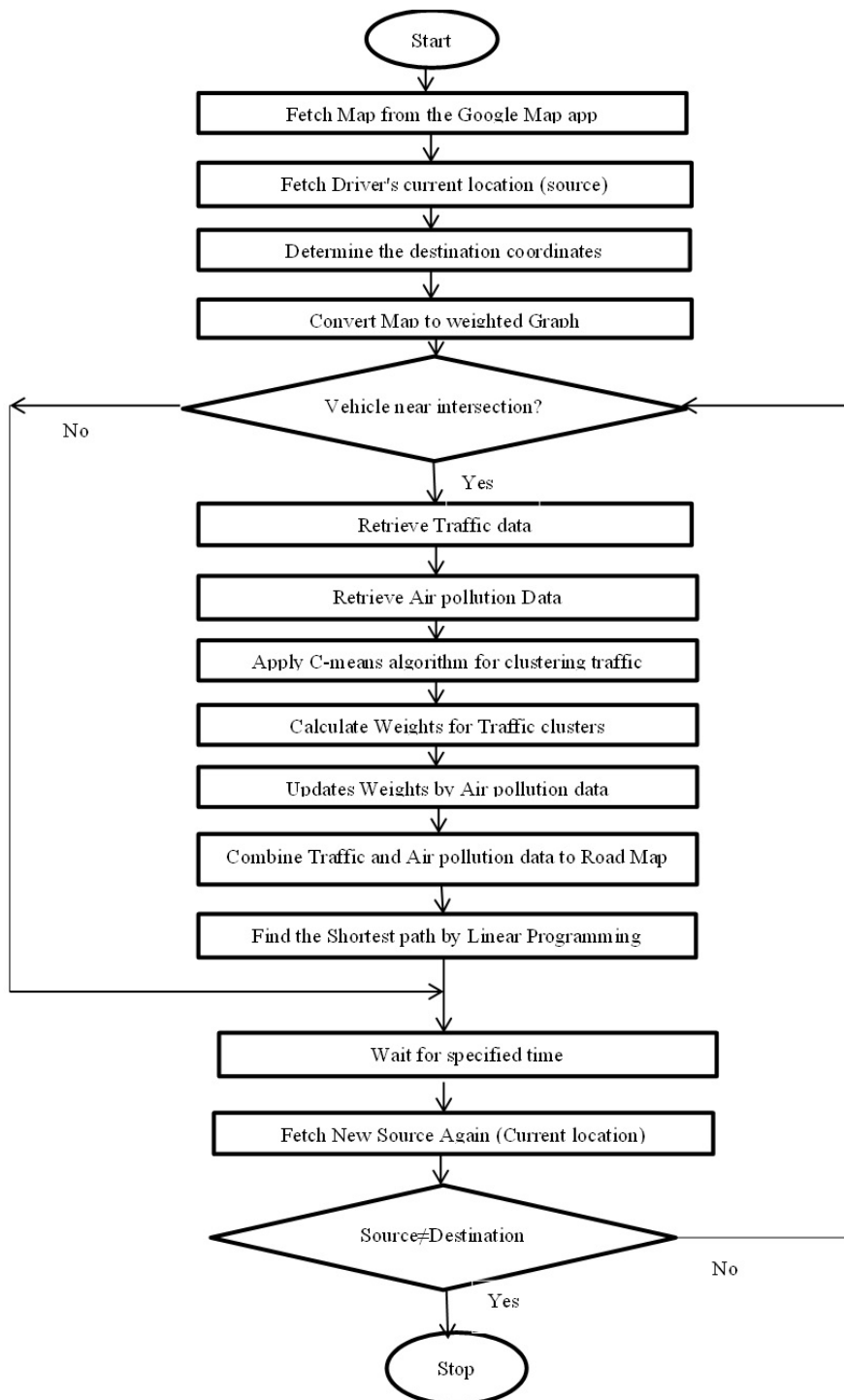
FIGURE 1. The flowchart of the proposed algorithm.

velocity and vehicle density. An important challenge to consider in the routing problem is fairness. As stated in [23], the min-max methods are the suitable for guarantee fairness. In our research, traffic density is used for controlling fairness. Traffic density is calculated in real-time and is the same for all vehicles in the same area. Although vehicles make individual

decisions, the input data of the algorithm is regularly updated, and this data is made available to all vehicles, and in addition, they also perform routing based on the same algorithm. Therefore, this variable can prevent excessive accumulation in one path. In other words, Di is a min-max indicator. If the amount of traffic density on a street is higher, the weight of
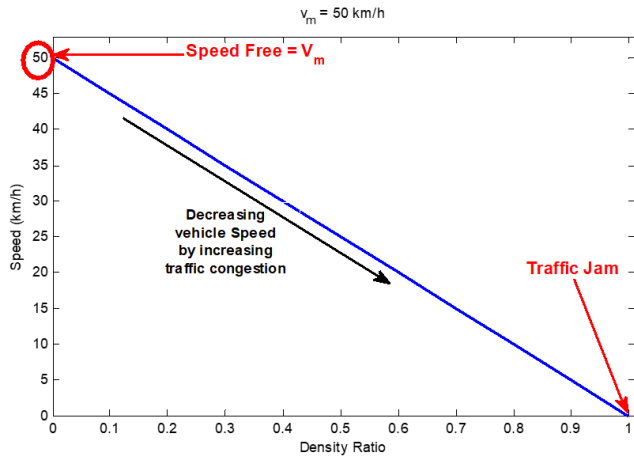
FIGURE 2. speed of vehicles vs. density ratio.

FIGURE 3. weights vs. speed and air pollution factor.

that route will be more, and as a result, the probability of choosing that route will be less, and vice versa. It should be noted, the speed is calculated according to the density of vehicles in each area. Therefore, the streets with more capacity have more vehicles.

V. AIR POLLUTION CLUSTERING

This paper considers air quality more than traffic congestion for routing. Six important pollutants agents are monitored in the air pollution monitoring stations. These agents are NO2, SO2, CO, O3, PM2.5, and PM10. Vehicle motors that are burning fossil fuels produce NO2. NO2 is very dangerous to the human lungs, and asthmatic patients should be kept away from it. SO2 is a toxic gas and has significant impacts on human health, plants, and animal life. CO is very toxic, which is caused by the incomplete combustion of carbon. CO affects neurotransmitters or blood vessels related to many diseases such as hypertension, neurodegeneration, heart diseases, and pathological inflammation. Ozone or O3 affects heart and the human nervous system and is very dangerous for pregnant women. PM2.5 is an atmospheric particulate matter of 2.5 micrometers, and it causes more than 4 million deaths and several million diseases such as heart diseases, lung cancer, etc., across the world. PM10 is an atmospheric particulate matter of 10 micrometers. These particles are less lethal than PM2.5. Air pollution monitoring stations indicate the daily average (24 hours) values of PM2.5 and PM10 [16], [24], [25]. Based on the above discussions, the air quality levels are shown in table 1. It is observed, that air quality is divided into six categories [16]. Many air pollution monitoring stations work based on this table and show air quality by a particular color. The amount of NO2 and CO2 that are shown in this table are average values in one hour. The amount of CO and O3 that are shown in this table are average values in 8 hours. The amount of PM2.5 and PM10 are average values in one day. In this research, each category (L) level is used as a factor for updating traffic cluster weight.

Now, the weights that are obtained from Eq. 3 are updated by considering air quality. Weights play an essential role in
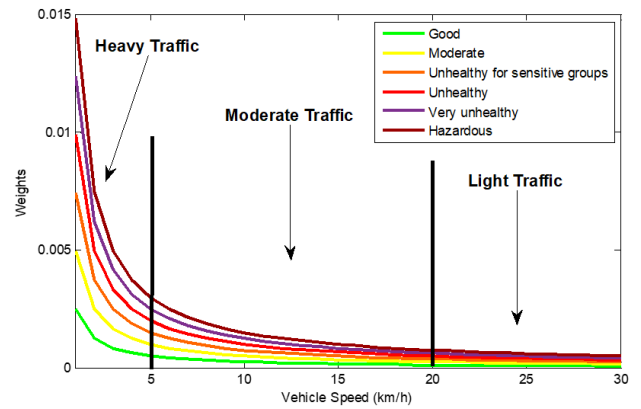
finding the shortest path. In other words, roads with higher weights (cost) have less chance to select by the algorithm as the best path. Polluted and crowded paths will not be selected unless there is no alternative path. In the proposed method, the weights that are obtained by Eq. 3 are updated as follows:

$$w_i = L \times \frac{\frac{1}{v_i + \sigma}}{\sum_{k=1}^{N}\left(\frac{1}{v_k + \sigma}\right)} \quad (4)$$

where L is the air pollution factor and its value is shown in table 1. It is observed, that for clean air, L is one, and for hazardous is six. Fig. 3 indicates the weights based on vehicles' speed and air pollution factors. The weights of the graph decrease when the vehicle's speed increases, and the graphs' weights increase with increasing the pollution factor. In our proposed method, the more appropriate paths have lower weights. In this figure, we assume that if the vehicles' speed is less than 5 km/h, the traffic congestion will be heavy, and if the speed of the vehicles is more than 20 km/h, the traffic congestion will be light.

VI. LINEAR PROGRAMMING ALGORITHM

Linear programming algorithm aims to achieve the best outcome in a mathematical model. This algorithm is used to optimize linear objective functions and is used in various optimization problems. For finding the shortest path by using this algorithm, the weighted graph is formulated as follows [26]:

$$min \sum_{(i,j)\in A} (w_{ij} \times x_{ij}) \quad (5)$$

Subject to:

$$\sum_{(i,j)\in\delta^+} x_{ij} - \sum_{(j,i)\in\delta^-} x_j i = \begin{cases} 1, & i = source \\ -1, & i = destination \ \forall i \in V \\ 0, & otherwise \end{cases} \quad (6)$$

$$\sum_{(i,j)\in\delta^+} x_{ij} \leq 1 \ \forall i \in V \quad (7)$$

$$x_{ij} \in \{0, 1\} \ \forall(i, j) \in A \quad (8)$$

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

IEEE*Access*

TABLE 1. Air quality category.

| Quality	L	Color	NO2 (ppm)	SO2 (ppm)	CO (ppm)	O3 (ppm)	PM2.5 (μg/m3)	PM10 (μg/m3)
Good	1	Green	0-53	0-35	0.0-4.4	0-0.054	0-12.0	0-54
Moderate	2	Yellow	54-100	36-75	4.5-9.4	0.055-0.070	12.1-35.4	55-154
Unhealthy for sensitive groups	3	Orange	101-360	76-185	9.5-12.4	0.071-0.085	35.5-55.4	155-254
Unhealthy	4	Red	361-649	186-304	12.5-15.4	0.086-0.105	55.5-150.4	255-354
Very unhealthy	5	Purple	650-1249	305-604	15.5-30.4	0.106-0.20	150.5-250.4	355-424
Hazardous	6	Maroon	1250-2049	605-1004	30.5-50.4	>0.2	250.5-500.4	425-604
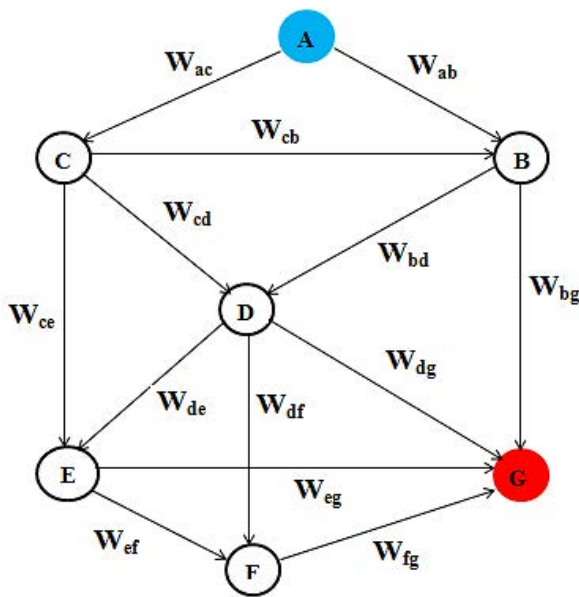
FIGURE 4. A simple routing by linear programming.

where w_{ij} is a weight (cost) road between node i-th and node j-th, x_{ij} is a binary variable that can be one or zero which One indicates the $arc(i, j)$ belongs to the path A. V is the graph vertexes and δ^+ and δ^- are the set of outgoing and incoming arcs of a node, respectively. Eq. 6 shows flow conservation constraints, and Eq. 7 guarantees each node's outgoing degree is at most one. Eq. 4 calculates the weights. Eq. 4 indicates that regions with less traffic density and high air quality have a small value, and for regions with high traffic density and low air quality, the weights have a large value. In other words, the congested roads or air pollutant zones will not be selected as a path unless there is an alternative path. To understand the linear programming method, consider a graph that is shown in Fig. 4. Node A is a source, and node G is a destination. W is obtained by using Eq. 4 shows weights between nodes. The aim is to find the minimum path between the source and destination. Now, we need to expand Eq. 6. Table 2 is calculated for the graph based on this equation. By setting

coefficients of x_{ij} as $+1$ and -1 for j and i, respectively, the linear program's corresponding values are listed in table 2.

Therefore, the optimization problem can be written as follows:

$$Min\ Z = W_{ab}x_{ab} + W_{ac}x_{ac} + W_{bd}x_{bd} + W_{bg}x_{bg} + W_{cb}x_{cb}$$
$$+ W_{cd}x_{cd} + W_{ce}x_{ce} + W_{de}x_{de} + W_{df}x_{df}$$
$$+ W_{dg}x_{dg} + W_{ef}x_{ef} + W_{eg}x_{eg} + W_{fg}x_{fg} \quad (9)$$

Subject to:

$$-x_{ab} - x_{ac} = -1 \quad (10)$$
$$x_{ab} - x_{bd} - x_{bg} + x_{cb} = 0 \quad (11)$$
$$x_{ac} - x_{cb} - x_{cd} - x_{ce} = 0 \quad (12)$$
$$x_{bd} + x_{cd} - x_{de} - x_{df} - x_{dg} = 0 \quad (13)$$
$$x_{ce} + x_{de} - x_{ef} - x_{eg} = 0 \quad (14)$$
$$x_{df} + x_{ef} - x_{fg} = 0 \quad (15)$$
$$x_{bg} + x_{dg} + x_{eg} + x_{fg} = +1 \quad (16)$$

Therefore, the problem is a binary integer programming problem. After solving this problem, the optimal solution is obtained. For instance, assuming $x_{ab} = 1$ and $x_{bg} = 1$ and other coefficients are zero, the optimal path is $A \rightarrow B \rightarrow G$.

VII. SIMULATION RESULTS

This section evaluates the proposed method by finding the best path between two locations in Tehran, Iran. Fig.5 indicates the small part of Tehran city. In this figure, the source is shown in red, and the destination is shown in green. It is observed that this figure is too crowded. Indeed, we cannot calculate the weighted graph of the image without this preprocessing procedure. This figure is converted to a black and white image, as shown in Fig. 6. The image details are removed in this figure, and only roads and intersections are shown in white. Then, this figure is converted into a weighted graph that is explained in the previous sections.

The traffic congestion in the streets is shown in Fig. 7. In this figure, each vehicle is shown by a small blue circle. It is observed traffic congestion is more in some streets. Therefore, it is necessary to cluster traffic. As mentioned

IEEE *Access*

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

TABLE 2. Linear programming values.

Z	x_{ab} W_{ab}	x_{ac} W_{ac}	x_{bd} W_{bd}	x_{bg} W_{bg}	x_{cb} W_{cb}	x_{cd} W_{cd}	x_{ce} W_{ce}	x_{de} W_{de}	x_{df} W_{df}	x_{dg} W_{dg}	x_{ef} W_{ef}	x_{eg} W_{eg}	x_{fg} W_{fg}	
A	-1	-1												=-1
B	+1		-1	-1	+1									=0
C		+1			-1	-1	-1							=0
D			+1			+1		-1	-1	-1				=0
E							+1	+1			-1	-1		=0
F									+1		+1		-1	=0
G				+1						+1		+1	+1	=+1

FIGURE 5. Road Map with source and destination.

FIGURE 6. Black and white map.
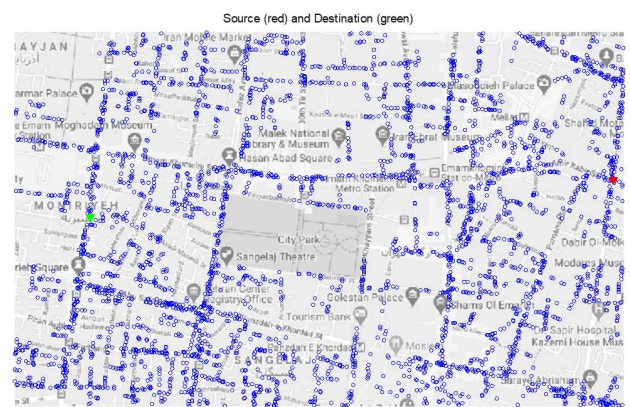
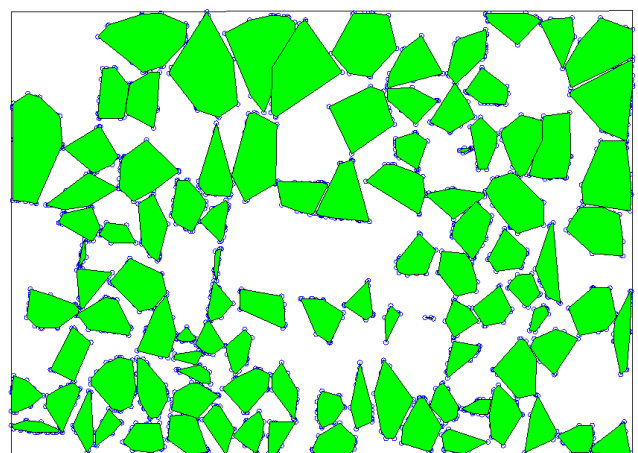**FIGURE 7.** Traffic congestion view in the map.

FIGURE 8. Clustering traffic cognations by C-means.

before, the C-means algorithm is used to classify traffic in each region.

The result of the C-means clustering and the density in each region is shown in Fig. 8. It is observed, its region border is specified by using the convex hull algorithm. The cluster's density is evident in this figure. In some regions, the traffic density is more than in other regions. The density of each region is calculated by using the algorithm that is explained in the previous sections.
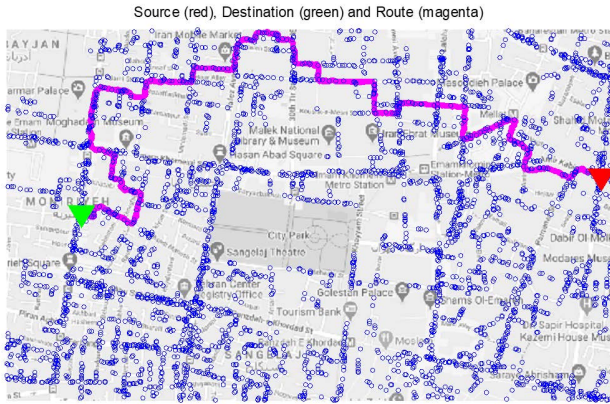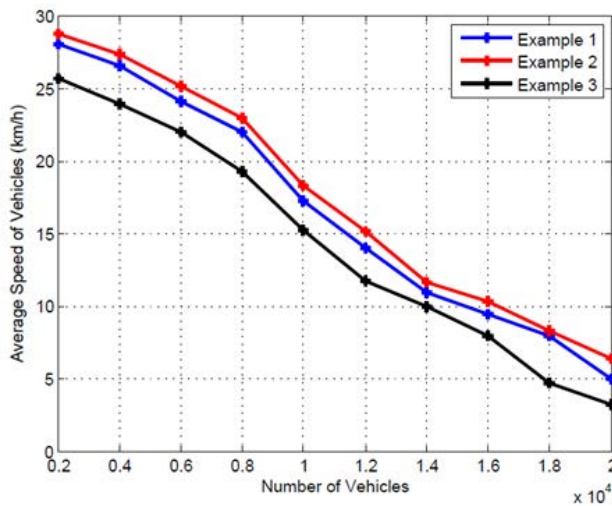
E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

IEEE *Access*

FIGURE 9. Result from the algorithm.

FIGURE 10. Average speed of vehicles vs. number of vehicles.

FIGURE 11. Distance between the source and the destination vs. number of vehicles.

FIGURE 12. Distance between the source and the destination vs. number of vehicles for different air pollution indices.

The result of the proposed algorithm is shown in Fig. 9. Result of the proposed algorithm is shown by purple line. Fig. 9 indicates that the shortest path between the source and the destination can be calculated when we do not consider traffic congestion and air quality.

Now, the performance of the algorithm is investigated by different examples. Free-flow speed of vehicles is assumed 30 km/h, and the number of vehicles is considered a variable. In these examples, the source and the destination are different. In each example, the aim is to find the optimal route in different areas of Tehran city. It is assumed that pollution is higher in areas close to the downtown, and the level of pollution index Downtown is 5, and in other areas is between 2 and 4. Figure 10 shows the average speed of vehicles for all examples. It is observed that the speed is decreased by increasing the number of vehicles. Therefore, increasing traffic congestion causes people to spend more time on the route. It is observed in Figure 11 that the proposed algorithm selects different routes for each example based on traffic congestion. It is observed that the algorithm selects alternative routes between the source and the destination.
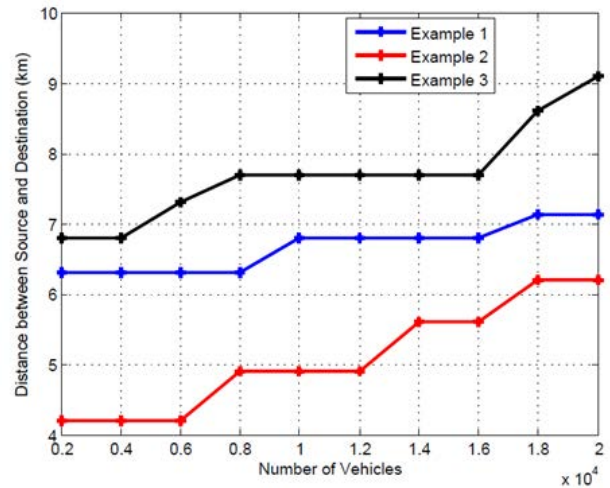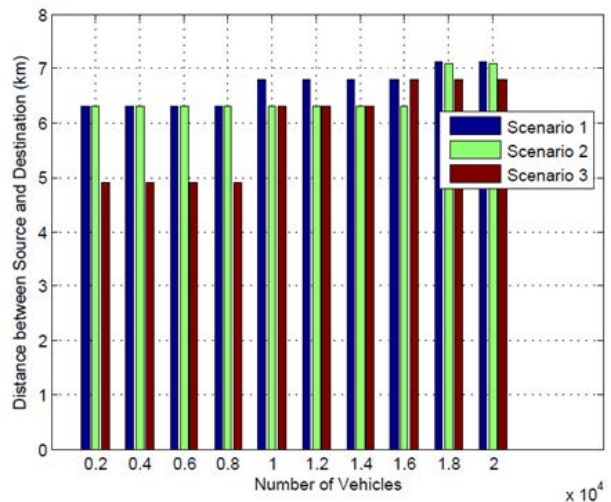
Figure12 shows the performance of the algorithm for different air pollution indices in three different scenarios. In all these scenarios, the source and the destination are the same. The details of these scenarios are as follows: Scenario 1: the level of pollution index Downtown is 5, and in other areas is between 2 and 4.

Scenario 2: the level of pollution index Downtown is 4, and in other areas is between 2 and 4.

Scenario 3: the level of pollution index Downtown is 3, and in other areas is between 2 and 3.

It is observed that the optimal path is changed based on air pollution indices. When the level of air pollution index is low, the algorithm selects the shortest path between the source and the destination. This figure indicates both air pollution and traffic congestion factors affect the optimal path finding algorithm.

The performance of the proposed algorithm is compared with other papers and the results are shown in figure 13.
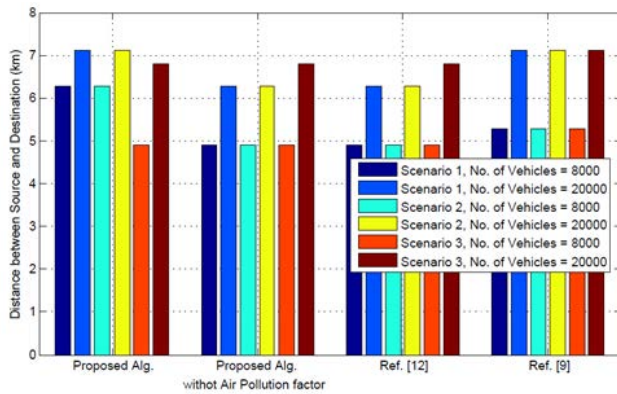
IEEE Access

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

FIGURE 13. Compare the performance of the algorithm with other methods.

The proposed algorithm is investigated with two different assumptions. In the first assumption, it is assumed both traffic congestion and air pollution factors are considered for finding the shortest path, while in the second one, only the traffic congestion factor is considered. Scenarios are similar to scenarios that are defined in figure 12. This figure shows when only traffic congestion is considered, both the proposed algorithm and the method in [12] have the same performance. The algorithm of [12] selects the optimal path based on five different factors including the average travel speed of the traffic, vehicles density, roads width, road traffic signals, and the roads' length, while the proposed algorithm considered traffic density and estimates average speed based on the traffic density. However, both algorithms have better performance than the method introduced in [9]. This figure shows for the proposed algorithm, the path may be longer when the air pollution factor is considered. When the air pollution factor is considered, the algorithm attempts to find the path that has the cleanest air.

The proposed algorithm uses different methods to find the optimal path. Its complicity is determined based on the complexity of two algorithms. The computational complexity of the linear programming algorithm for calculating the shortest path is $O(n2)$, where the number of vertices. It is observed that with increasing the number of intersections between the source and destination, the dimensions of the optimization problem become larger and as a result the computational complexity also increases. For the c-means clustering algorithm, computational complicity is $O(NCT)$, where N is the number of points, C is the number of clusters, and T is the number of iterations to run by the procedure. Increasing the number of vehicles in the city increases the amount of N, thus increasing the computational complexity of the clustering algorithm.

VIII. CONCLUSION

This paper investigates the optimal shortest path routing in smart cities. In this method, the map is fetched from Google Map, and the map is converted to a weighted graph using image-processing methods. Traffic data is collected from

GPS devices, and air pollution data is collected from air pollution monitoring stations. The weights of the graph are calculated based on traffic congestion and air quality. Higher weights are allocated to dense regions or contaminated areas, and fewer weights are allocated to regions with good air quality and low traffic congestion. Then, the routing problem is formulated as an optimization algorithm, and a linear programming method is used to solve this problem. Polluted and crowded paths will not be selected unless there is no alternative path. The proposed algorithm's performance is evaluated using the Tehran city map to find the shortest path.

REFERENCES

[1] G. P. R. Filho, R. I. Meneguette, J. R. T. Neto, A. Valejo, L. Weigang, J. Ueyama, G. Pessin, and L. A. Villas, "Enhancing intelligence in traffic management systems to aid in vehicle traffic congestion problems in smart cities," *Ad Hoc Netw.*, vol. 107, Oct. 2020, Art. no. 102265, doi: 10.1016/j.adhoc.2020.102265.

[2] D. Li, L. Deng, and Z. Cai, "Intelligent vehicle network system and smart city management based on genetic algorithms and image perception," *Mech. Syst. Signal Process.*, vol. 141, Jul. 2020, Art. no. 106623, doi: 10.1016/j.ymssp.2020.106623.

[3] T. Lusikka, T. K. Kinnunen, and J. Kostiainen, "Public transport innovation platform boosting intelligent transport system value chains," *Utilities Policy*, vol. 62, 2020, pp. 1–8, doi: 10.1016/j.jup.2019.100998.

[4] O. Krivolapova, "Algorithm for risk assessment in the introduction of intelligent transport systems facilities," *Transp. Res. Proc.*, vol. 20, pp. 373–377, Jan. 2017, doi: 10.1016/j.trpro.2017.01.056.

[5] V. Vlasov, I. Konin, P. Pribyl, and V. Bogumil, "Development of standards and their harmonization with international standards as a necessary condition of normative and technical support in construction and development of intelligent transport systems in Russia," *Transp. Res. Proc.*, vol. 20, pp. 690–694, Jan. 2017, doi: 10.1016/j.trpro.2017.01.112.

[6] V. Lozhkin, B. Gavkalyk, O. Lozhkina, S. Evtukov, and G. Ginzburg, "Monitoring of extreme air pollution on ring roads with $PM_{2.5}$ soot particles considering their chemical composition (case study of Saint Petersburg)," *Transp. Res. Proc.*, vol. 50, pp. 381–388, Jan. 2020, doi: 10.1016/j.trpro.2020.10.045.

[7] E. Pecorari, M. Menegaldo, E. Innocente, A. Ferrari, G. Giuponi, G. Cuzzolin, and G. Rampazzo, "On which grounds a decision is taken in waterborne transport technology to reduce air pollution?" *Atmos. Pollut. Res.*, vol. 11, no. 12, pp. 2088–2099, Dec. 2020, doi: 10.1016/j.apr.2020.07.018.

[8] V. Pattanaik, M. Singh, P. K. Gupta, and S. K. Singh, "Smart real-time traffic congestion estimation and clustering technique for urban vehicular roads," in *Proc. IEEE Region Conf. (TENCON)*, Nov. 2016, pp. 3420–3423, doi: 10.1109/TENCON.2016.7848689.

[9] Q. Yu, Y. Luo, C. Chen, and X. Zheng, "Road congestion detection based on trajectory stay-place clustering," *ISPRS Int. J. Geo-Inf.*, vol. 8, no. 6, p. 264, Jun. 2019, doi: 10.3390/ijgi8060264.

[10] G. Zhu, J. Chen, and P. Zhang, "Fuzzy C-means clustering identification method of urban road traffic state," in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discovery (FSKD)*, Aug. 2015, pp. 302–307, doi: 10.1109/FSKD.2015.7381958.

[11] R. C. de Andrade and R. D. Saraiva, "An integer linear programming model for the constrained shortest path tour problem," *Electron. Notes Discrete Math.*, vol. 69, pp. 141–148, Aug. 2018, doi: 10.1016/j.endm.2018.07.019.

[12] H. M. Amer, H. Al-Kashoash, M. Hawes, M. Chaqfeh, A. Kemp, and L. Mihaylova, "Centralized simulated annealing for alleviating vehicular congestion in smart cities," *Technolog. Forecasting Social Change*, vol. 142, pp. 235–248, May 2019.

[13] H. M. Amer, H. A. A. Al-Kashoash, A. Kemp, L. Mihaylova, and M. Mayfield, "Coalition game for emergency vehicles re-routing in smart cities," in *Proc. IEEE 10th Sensor Array Multichannel Signal Process. Workshop (SAM)*, Jul. 2018, pp. 306–310.

E. Ghaffari *et al.*: Optimal Path-Finding Algorithm in Smart Cities by Considering Traffic Congestion and Air Pollution

IEEE *Access*

[14] H. Amer, N. Salman, M. Hawes, M. Chaqfeh, L. Mihaylova, and M. Mayfield, "An improved simulated annealing technique for enhanced mobility in smart cities," *Sensors*, vol. 16, no. 7, p. 1013, Jun. 2016.

[15] *Regional Office for Europe. Health Risk Assessment of Air Pollution: General Principles*, World Health Organization. Regional Office for Europe, World Health Organization, Geneva, Switzerland, 2016. [Online]. Available: https://apps.who.int/iris/handle/10665/329677

[16] G. Díaz, H. Macià, V. Valero, J. Boubeta-Puig, and F. Cuartero, "An intelligent transportation system to control air pollution and road traffic in cities integrating CEP and colored Petri nets," *Neural Comput. Appl.*, vol. 32, no. 2, pp. 405–426, Jan. 2020, doi: 10.1007/s00521-018-3850-1.

[17] X. Li and K. L. Yeung, "Traffic engineering in segment routing networks using MILP," *IEEE Trans. Netw. Service Manage.*, vol. 17, no. 3, pp. 1941–1953, Sep. 2020.

[18] T. Krauss and J. McCollum, "Solving the network shortest path problem on a quantum annealer," *IEEE Trans. Quantum Eng.*, vol. 1, pp. 1–12, 2020.

[19] [Online]. Available: https://www.google.com/maps/@35.7355566,51.3364166,12z

[20] S. Zeng, Z. Wang, R. Huang, L. Chen, and D. Feng, "A study on multi-kernel intuitionistic fuzzy C-means clustering with multiple attributes," *Neurocomputing*, vol. 335, pp. 59–71, Mar. 2019, doi: 10.1016/j.neucom.2019.01.042.

[21] M. L. Baptista, E. M. P. Henriques, and K. Goebel, "More effective prognostics with elbow point detection and deep learning," *Mech. Syst. Signal Process.*, vol. 146, 2020, pp. 1–14, doi: 10.1016/j.ymssp.2020.106987.

[22] H. Cevikalp, "High-dimensional data clustering by using local affine/convex hulls," *Pattern Recognit. Lett.*, vol. 128, pp. 427–432, Dec. 2019, doi: 10.1016/j.patrec.2019.10.007.

[23] J. Kleinberg, Y. Rabani, and E. Tardos, "Fairness in routing and load balancing," in *Proc. 40th Annu. Symp. Found. Comput. Sci.*, Oct. 1999, pp. 568–578.

[24] G. Dı' az, H. Macia, V. Valero, and F. Cuartero, "An intelligent transportation system to control air pollution in cities using complex event processing and colored Petri nets," in *Advances in Computational Intelligence*. Cham, Switzerland: Springer, 2017, pp. 415–426, doi: 10.17632/cbjxbhzn43.1.

[25] S. Gurram, A. L. Stuart, and A. R. Pinjari, "Agent-based modeling to estimate exposures to urban air pollution from transportation: Exposure disparities and impacts of high-resolution data," *Comput., Environ. Urban Syst.*, vol. 75, pp. 22–34, May 2019, doi: 10.1016/j.compenvurbsys.2019.01.002.

[26] L. Taccari, "Integer programming formulations for the elementary shortest path problem," *Eur. J. Oper. Res.*, vol. 252, no. 1, pp. 122–130, Jul. 2016, doi: 10.1016/j.ejor.2016.01.003.

AMIR MASOUD RAHMANI received the B.S. degree from Amir Kabir University, Tehran, in 1996, the M.S. degree from the Sharif University of Technology, Tehran, in 1998, and the Ph.D. degree from IAU University, Tehran, in 2005, all in computer engineering. He is currently a Professor of computer engineering. His research interests include distributed systems, the Internet of Things, and evolutionary computing.

MORTEZA SABERIKAMARPOSHTI received the Ph.D. degree in computer science from the University of Technology Malaysia. He is an Assistant Professor of computer science with IAU, South Tehran Branch. He works in IAU, South Tehran Branch, as a Faculty Member and has several years of work experience in the areas of teaching, research, administrative, (head of department and research institute), programming, and student's affairs; and arranging/organize research conferences, seminars, workshops, and events. He also has several research publications in well-known international journals and conferences.

ELHAM GHAFFARI received the B.S. degree in computer engineering from the Islamic Azad University of Shiraz and the M.S. degree in computer engineering from Islamic Azad University, Science and Research Branch of Fars. She is currently pursuing the Ph.D. degree with the Department of Computer Engineering, Qeshm Branch, Islamic Azad University, Qeshm, Iran. Her research interests include cloud computing, the Internet of Things, and data mining.

AMIR SAHAFI received the B.Sc. degree from the Shahed University of Tehran, Iran, in 2005, the M.Sc. and Ph.D. degrees from the Science and Research Branch of Islamic Azad University, Tehran, in 2007 and 2012, respectively, all in computer engineering. He is an Assistant Professor with the Department of Computer Engineering, South Tehran Branch, Islamic Azad University. His current research interests include distributed and cloud computing.

• • •