Journal of Science and Technology ISSN: 2362-1443
2023, Vol. 3, No. 1, 74-84

Travel Route Planning Using Fog-Cloud Architecture and Optimum Path Calculation

Hritik Jung Basnet
Nepal College of Information Technology
hritik.1929111@ncit.edu.np

Niranjan Khakurel
Nepal College of Information Technology
niranjan@ncit.edu.np

Article History:
Received: 15 July 2023
Revised: 8 October 2023
Accepted: 17 December 2023

Keywords—Fog-Cloud Architecture, Roadside Units, VANET Communication, Road Weight Measurement

***Abstract*—Traffic flow is random and discontinuous bearing in mind the incalculable human behaviors and also because of various factors such as traveling time, fuel consumption, etc. There are also other various influential factors such as accidents, condition of roads, weather, among others which makes route planning a difficult job. Traffic congestion detection and route recommendation are major aspects while performing congestion control. The motivation of this paper is to present a route recommendation system based on a multi-layer fog-cloud architecture. We deploy Roadside Units (RSUs) which are important for calculating the information about traffic influencers such as travel time, congestion area intersections prediction, etc. A cloud service provider and a road side unit are responsible for detection of traffic and sharing of information with the help of VANET communication between vehicles that are relayed to other vehicles with the help of beacons and nearby RSUs using software defined network architecture. Road Weight Measurement (RWM) is used to detect the congestion affected roads and provide appropriate route recommendations using various fitness functions. The model can be useful for easing the concern of traffic congestion issues using optimum path calculation for finding the shortest and the fastest routes. comparison to other guidance systems.**

I. INTRODUCTION

It is found that with better traffic prediction and route planning, the amount of travel time can be reduced considerably, while the speed can also be reduced. Also, the fuel consumption can be reduced for both passenger vehicles and trucks.

Although there have been various traffic flow prediction systems and models, because of improper travel route planning, the traffic congestion situation does not tend to improve and to solve this problem is the real motivation of this paper.

II. BRIEF HISTORY

A. Map Matching

The process of finding out the exact location of automobiles using various navigation systems with the help of various data from smart cards, geographic information system (GIS), global positioning system (GPS) and so on. It is important to relate information on these data because it helps improve reliability, continuity and accuracy of moving objects. Some of the major map matching algorithms are geometric information, road topology information and integrated information. [1]

Li and Huang [2] determined the degree of geometric resemblance between desired routes and GPS trajectories by proposing map matching models. A genetic model is used for comparison and A* search calculates the cost of each path. The use of dynamic time warp approach can be seen for calculating the shape similarities. These structures that are used for map matching on the basis of geometric information are considered light and cannot produce accurate findings in cases where they are dense crowding situations. [3]

The map matching models can have high accuracy if they work on the basis of probability because of the tendency for giving complete matching trajectories. The downside of such methods are that they require huge amounts of data and can cause map mismatching if there is not a required amount of training data.

There are also some approaches that are based on the topological relationship of road network for estimation of intersections of road and map matching based on historical trajectory information. The major problem of such a method includes only the GPS sites and candidate road distance without concern about direction of drive [4]. The model also cannot perform well using trajectories from GPS having many peculiar locations.

Mohamed et al. [5] used a model based on the hidden markov model (HMM) which is very much promising for GPS and mobile which has highly accurate map matching. The HMM based model has also been modified with one that uses the Viterbi method for construction of high confidence partial route [6].

B. Route Planning

There have been various techniques, methods and algorithms that has been devised for route prediction and guidance over the years. One of the first methods for finding the best route was using a genetic algorithm which was proposed by Steven Chien [7]. An exhaustive search algorithm was applied for validating whether the algorithm could converge to the optimal solution. A. Karbassi [8] described a route prediction and time of arrival estimation technique based on initial information from origin, destination station and trip start time. The most probable routes are calculated using a hierarchical tree data structure after new data are received. However, the results could have been improved if there was an increase in the frequency of position/time data.

R. Claes [9] combined the ant colony algorithm technique with link travel time prediction for finding routes that could reduce the amount of time spent by travelers on the basis of link travel time prediction. S. Chowdhury [10] developed a model for determining the effects of personalized information provision from smartphones for planning trips on user's ability. The main idea was to identify the perception of users riding public transport through multi destination trip planning smartphone applications.

H. Huang [11] proposed a method for merging public transport and car pooling networks for multi-modal route planning. The main idea behind the model is on the basis of drive time areas and point of action. This enables users to plan trips from origin to destination using different combinations of modes. T. Liebig [12] presented a system that could incorporate future traffic hazards while routing for individual trip planning on the basis of sensor reading using spatio-temporal random fields. The model could compute traffic flow using Gaussian process regression for low sensor coverage areas. This model allowed incorporation of historical data, streamed data and rich dependency structure in parallel to each other.

P. Perez-Murueta [13] proposed a model for vehicle redirection for congestion avoidance on the basis of deep learning for predicting future state of traffic networks. The information from the previous step are vital for predicting the future state and alternative routes using entropy balanced k shortest path algorithm (EbkSP). S. Koh [14] used a deep reinforcement learning method for real time vehicle routing and navigation. Smart agents are integrated to facilitate navigation with more significance for maps with more edges.

1) Rerouting for Lower Travel Time

Google maps is the most popular navigation application used by people for quick and effective directions. Dijkstra's algorithm, articulation of Edsger Dijkstra which is a map search algorithm, is used for finding the shortest path in which we use k-search strategies where each node is visited from beginning to the end for finding the path. Before, google maps used the data collected from sensors to generate traffic information. But, with the increase in global population's usage of Google, the traffic information is now generated from their own users who have set location to state on in Google Maps app. The prediction is then based on various historical patterns that are analyzed from different roads during peak hours and normal hours. Despite the use of infrastructure based traffic information for calculating the shortest routes, prevention of traffic congestion explicitly is not performed because these are reactive solutions and provide similar guidance to all vehicles. Hence, there are problems of route oscillations that are similar to that of computer networks [15].

2) Dijkstra's Algorithm

As Google maps uses Dijkstra's which is the most popular navigation system used today, it is an important aspect used for finding the shortest path between two points in a weighted graph. Comparatively, there are only few disadvantages of Dijkstra's algorithm to that of other algorithms such as A shortest path with repulsion (AR*), the random k shortest path (RkSP), the entropy-balanced kSP (EBkSP), and the flow-balanced kSP (FbkSP). Even though all these methods can perform better than the state of art solutions such as in-car navigation systems (Garmin and TomTom), web services based route computation (Google and Microsoft) and dynamic traffic assignment (DTA), Despite the lengthy complexity of Dijkstra's algorithm, the capability of modern computers have created the possibility of using Dijkstra's algorithm [16].

3) Cloud Service Provider

The centralized cloud service provider may be placed at the traffic management center (TMC) that comprises an expansive set of equipment, program and administration apparatuses that are required to store, manage and prepare the information. The information in the center is static as well as dynamic where the static information consists of capacity, length, speed of each road constantly being updated. The distributed layer is responsible for collection of information from various vehicles and roadside units that reside in a particular location [17].

During peak hours, there can be huge amounts of traffic requests that the TMC has to respond to and solving challenges of delay in the outcome, overhead and bottleneck. For this purpose, the segmentation of road networks into sub road networks can help the process.

4) Roadside Units

These devices that are fixed in different geographic locations (mainly on roadside or pedestrian passageways) can act as transceivers. After the sub road networks have been defined, each area can be allocated a single RSU so that we can process, sense, exchange and communicate information. These are fog-devices that are capable of performing congestion detection and rerouting for vehicles along with a traffic management center.

However, RSU can only perform operations within its own area in addition to estimating inflow / outflow. In outline, we have a congestion detection system working collaboratively and a re-routing process taking place linking the RSU and the TMC. The main role of TMC is evaluating the traffic weight for each street utilizing the inflow / outflow information from vehicles and the RSU [18].

5) Intelligent Traffic lights (iTL)

These fixed elements are similar to deployment of RSU at each intersection in the distributed layer. The iTL's primary task is control signals for allowing or blocking vehicle's flow during different time instances and estimating inflow /

outflow. Finally, iTLs use the closest RSU for sending information and the RSU then forwards the information to the TMC [16].

III. METHODOLOGY

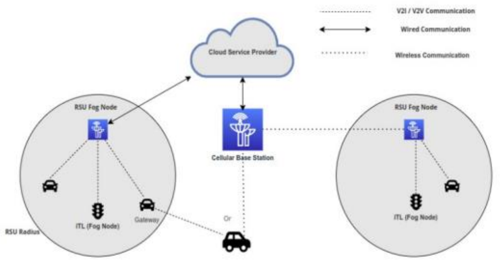

Fig. 1. Proposed Infrastructure

A. *Rerouting Process*

Inductive loop detector sensors which belong to stationary sensing roadside units are the most commonly used technique for detecting traffic overcrowding [20]. Other techniques involve the use of magnetic sensors and radio-frequency identification readers. Probe based techniques also play an important role for detecting traffic congestion. Probe based techniques can use global positioning systems and cellular signal based methods that require accelerometers and cellular signals for traffic congestion detection.

For effective route planning and recommendation, we use both the roadside unit (RSU) and a cloud service provider for congestion detection and processing the information to finally provide a re-route plan to every vehicle. RSU are able to perform rerouting within its own area while cloud service providers can reroute vehicles on areas where RSU has not covered. The detailed process for routing and route recommendation is shown in the figure below:

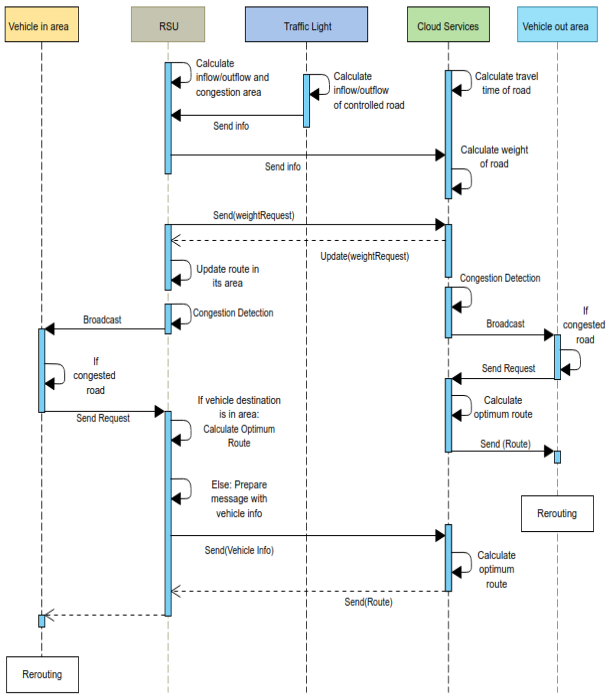

Fig. 2. Rerouting process

B. *Mathematical Modeling*

We define a road network consisting of weighted coordinated charts for calculating the weight of each road network.

1) *Traffic Conditions Determination*

The metrics that are considered while calculating the road weight measurement (RWM) are:

- Traveling time (TT)
- Congestion detection measure (CDM)
- Predicted traffic flow model of road.

2) *Traveling time (TT)*

$$\widehat{TT}_{e_{v_i,v_{i+1}}}(t) = \begin{cases} \dfrac{L_{e_{v_i,v_{i+1}}}}{Smax_{e_{v_i,v_{i+1}}}} & t = 0 \\ \left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t-1) \times \beta\right) + \left(\dfrac{L_{e_{v_i,v_{i+1}}}}{\bar{S}_{e_{v_i,v_{i+1}}}(t)} \times (1-\beta)\right) & t \neq 0 \end{cases} \tag{1}$$

If there are no vehicles on the road, we will achieve the maximum speed. Then, the average traveling speed of a vehicle is estimated as follows:

$$\bar{S}e_{v_i,v_{i+1}}(t) = \frac{\sum n \in N^{s_n}}{|N|} \tag{2}$$

The normalization of traveling time for value to be between 0 and 1 is given by:

$$N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right) = \begin{cases} \left(\dfrac{L_{e_{v_i,v_{i+1}}}}{Smax_{e_{v_i,v_{i+1}}} \times Lmax^{net}_{e_{v_i,v_{i+1}}}}\right) & t = 0 \\ \left(N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t-1)\right) \times \beta\right) + \left(\left(\dfrac{L_{e_{v_i,v_{i+1}}}}{\overline{S_{e_{v_i,v_{i+1}}}}(t) \times Lmax^{net}_{e_{v_i,v_{i+1}}}}\right) \times (1-\beta)\right) & t \neq 0 \end{cases} \tag{3}$$

Finally, rescaling value again to 0 and 1

$$N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} = \left(\left(\frac{\left(1 - min_{N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)}\right)}{max_{N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)} - min_{N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)}}\right) \times \left(N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right) - min_{N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)}\right)\right) + min_{N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)} \tag{4}$$

3) *Congestion detection measure (CDM)*

In order to measure the road traffic condition, we define a mean speed of vehicles on the road which is given by a speed index whose value is between 0 and 1, i.e.

a) Density Index

$$\widehat{DI}_{e_{v_i,v_{i+1}}}(t) = \frac{N_{e_{v_i,v_{i+1}}}(t)}{C_{e_{v_i,v_{i+1}}}} \tag{5}$$

$$\widehat{SI}_{e_{v_i,v_{i+1}}}(t) = 1 - \frac{\overline{S}_{e_{v_i,v_{i+1}}}(t)}{Smax_{e_{v_i,v_{i+1}}}} \tag{6}$$

b) The maximum capacity of road segment is given by:

$$C_{e_{v_i,v_{i+1}}} = \frac{L_{e_{v_i,v_{i+1}}}}{L_{Gap}+L_{vehicle}} \times N_{lane} \tag{7}$$

c) The road congestion measure (RCM) is then given as:

$$\widehat{RCM}_{e_{v_i,v_{i+1}}}(t) = \left(\widehat{SI}_{e_{v_i,v_{i+1}}}(t)\right) + \left(\widehat{DI}_{e_{v_i,v_{i+1}}}(t)\right) \tag{8}$$

The subgraph from the original graph composing various roads on the basis of their communication radius is present on each RSU and finally we calculate the weight of traffic of an area known as the zone congestion measurement (ZCM) which is between 0 and 1.

$$\widehat{ZCM}_{RSU_r}(t) = \frac{\sum_{e_{v_i,v_{i+1}} \in [RSU_r]}^{R} \left(L_{e_{v_i,v_{i+1}}} \times \frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2}\right)}{\sum_{e_{v_i,v_{i+1}} \in [RSU_r]}^{R} L_{e_i}} \tag{9}$$

Finally, the congestion level inside an area is given as:

$$\widehat{CDM}_{e_{v_i,v_{i+1}}}(t) = \frac{\widehat{ZCM}_{RSU_r}(t) + \frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2}}{2} \tag{10}$$

C. Traffic Flow Model

More often than not, vehicles comprise a predefined way that has street groupings from the area of vehicles as of now to its goal route (destination).

a) Stream of traffic inflow and outflow for a road

We consider, the inflow rate of a road denoted by the following terms respectively: -

$$\Theta_{e_{v_i,v_{i+1}}}(t)$$

$$\Delta_{e_{v_i,v_{i+1}}}(t)$$

Then, the sum of inflow rate and outflow rate in a given time interval is given by Eqn. 13 and Eqn. 14 respectively:

$$\text{Sum of inflow rate} = \sum_{u=1}^{U_{e_{v_i,v_{i+1}}}} \Theta_{e_{v_i,v_{i+1}}}(t) \tag{11}$$

$$\text{Sum of outflow rate} = \sum_{d=1}^{D_{e_{v_i,v_{i+1}}}} \Delta_{e_{v_i,v_{i+1}}}(t) \tag{12}$$

b) Traffic low diffusion (TFD)

We consider the traffic lights in each intersections, the green time ratio which is in accordance with the traffic flow during certain time interval and finally the traffic lifecycle to calculate the speed of traffic flow diffusion which is given by:

$$\text{TFD} = \frac{T^g(t)}{T_{cycle}} \times \frac{1}{\widehat{TT}_{e_{v_i,v_{i+1}}}(t)} \tag{13}$$

If there are intersections without traffic light we consider only

$$\frac{1}{\widehat{TT}_{e_{v_i,v_{i+1}}}(t)}$$

where, the road travel time is calculated from the equation above.

Therefore, the inflow and outflow rate under the guidance of traffic is defined by the Eqn. 14 and Eqn. 15 given below.

$$\Theta_{(e_{v_i,v_{i+1}})_u}(t) = \sum_{u=1}^{U_{e_{v_i,v_{i+1}}}} Q_{(e_{v_{i-1},v_i})_u}(t) \times \theta_{(e_{v_{i-1},v_i})_u \to e_{v_i,v_{i+1}}}(t) \times \frac{T^g_{v_{i-1},v_i \to (e_{v_i,v_{i+1}})_d}(t)}{T_{cycle}} \times \frac{1}{\widehat{TT}_{e_{v_i,v_{i+1}}}(t)} \tag{14}$$

$$\Delta_{e_{v_i,v_{i+1}}}(t) = \sum_{d=1}^{D_{e_{v_i,v_{i+1}}}} Q_{e_{v_i,v_{i+1}}}(t) \times \delta_{e_{v_i,v_{i+1}} \to (e_{v_{i+1},v_{i+2}})_d}(t) \times \frac{T^g_{e_{v_i,v_{i+1}} \to (e_{v_{i+1},v_{i+2}})_d}(t)}{T_{cycle}} \times \frac{1}{\widehat{TT}_{e_{v_i,v_{i+1}}}(t)} \tag{15}$$

During time instance t + 1, the instant traffic prediction on

$$e_{v_i,v_{i+1}}$$

is given by:

$$\widehat{Q}_{e_{v_i,v_{i+1}}}(t+1) = max\left\{Q_{e_{v_i,v_{i+1}}}(t) - \sum_{d=1}^{D_{e_{v_i,v_{i+1}}}} \Delta_{e_{v_i,v_{i+1}}}(t) - Q_{arr,e_i}(t), 0\right\} + \sum_{u=1}^{U_{e_{v_i,v_{i+1}}}} \Theta_{e_{v_i,v_{i+1}}}(t) + Q_{dep,e_{v_i,v_{i+1}}}(t) \tag{16}$$

c) Traffic inflow and outflow for a region

In order to calculate the traffic inflow and outflow for a certain region, we current anticipated traffic weight through different regions which is given by:

$$\widehat{Q}_{RSU_r}(t+1) = max\left\{\sum_{e_{v_i,v_{i+1}}=1}^{E_{RSU_r}} Q_{RSU_r}(t) - \Delta_{RSU_r}(t) - \sum_{e_{v_i,v_{i+1}}=1}^{E_{RSU_r}} Q_{arr,RSU_r}(t), 0\right\} + \Theta_{RSU_r}(t) + \sum_{e_{v_i,v_{i+1}}=1}^{E_{RSU_r}} Q_{dep,RSU_r}(t) \tag{17}$$

Also,

$$\Theta_{RSU_r}(t) = \sum_{e_i^h=1}^{H_{e_i}} \sum_{u^H=1}^{U_{e_i}^H} Q_{(e_i^h)_{u^H}}(t) \times \theta_{(e_i^h)_{u^H} \to \acute{e}_i}(t) \times \frac{T^g_{(e_i^h)_{u^H} \to e_i}(t)}{T_{cycle}} \times \frac{1}{\widehat{TT}_{(e_i^h)_{u^H}}(t)} \tag{18}$$

$$\Delta_{RSU_r}(t) = \sum_{e_i^{out}=1}^{E_{RSU_r}^{out}} \sum_{d=1}^{D_{e_i}} Q_{e_i^{out}}(t) \times \delta_{e_i^{out} \to (e_{i+1})_d}(t) \times \frac{T^g_{e_i^{out} \to (e_{i+1})_d}(t)}{T_{Cycle}} \times \frac{1}{\widehat{TT}_{e_i^{out}}(t)} \tag{19}$$

Finally, the combined weight of road can be calculated as follows:

$$\widehat{RWM}_{e_{v_i,v_{i+1}}} = e^{(\Gamma \times a)} + (\Lambda \times (a+b+c)) \tag{20}$$

There are four necessary test cases that need to be examined during situations where the road belongs to a RSU and when it does not belong to an RSU.

Road within a RSU

$$a = \begin{cases} \widehat{CDM}_{e_{v_i,v_{i+1}}}(t) & \widehat{CDM}_{e_{v_i,v_{i+1}}}(t) \geq \varepsilon \\ N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} & \text{Otherwise} \end{cases} \tag{21}$$

$$\acute{a} = \begin{cases} N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} & \widehat{CDM}_{e_{v_i,v_{i+1}}}(t) \geq \varepsilon \\ \widehat{CDM}_{e_{v_i,v_{i+1}}}(t) & \text{Otherwise} \end{cases} \tag{22}$$

Road not within a RSU

$$a = \begin{cases} \widehat{RCM}_{e_{v_i,v_{i+1}}}(t) & \widehat{RCM}_{e_{v_i,v_{i+1}}}(t) \geq \varepsilon \\ N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} & \text{Otherwise} \end{cases} \tag{23}$$

$$\acute{a} = \begin{cases} N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} & \widehat{RCM}_{e_{v_i,v_{i+1}}}(t) \geq \varepsilon \\ \widehat{RCM}_{e_{v_i,v_{i+1}}}(t) & \text{Otherwise} \end{cases} \tag{24}$$

$$b = \frac{\widehat{Q}_{e_i}(t+1)}{C_{e_{v_i,v_{i+1}}}} \tag{25}$$

$$c = \begin{cases} \dfrac{\widehat{Q}_{RSU_r}(t+1)}{\sum_{\acute{e}_{v_i,v_{i+1}}=1}^{\acute{E}_{RSU_r}} C_{\acute{e}_{v_i,v_{i+1}}}} & \text{If the road is in the RSU area} \\ 0 & \text{Otherwise} \end{cases} \tag{26}$$

d) Total fitness of road

The weight of roads within RSU is given by:

If $\widehat{CDM}_{e_{v_i,v_{i+1}}}(t) \geq \varepsilon$:

$$\widehat{RWM}_{e_{v_i,v_{i+1}}} = e^{\left(\Gamma \times \widehat{CDM}_{e_{v_i,v_{i+1}}}(t)\right)} + \Lambda \times \left(N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} + \frac{\widehat{Q}_{e_{v_i,v_{i+1}}}(t+1)}{C_{e_{v_i,v_{i+1}}}} + \frac{\widehat{Q}_{RSU_r}(t+1)}{\sum_{e=1}^{E_{RSU_r}} C_{e_{v_i,v_{i+1}}}}\right) \tag{27}$$

Else if $\widehat{CDM}_{e_{v_i,v_{i+1}}}(t) < \varepsilon$:

$$\widehat{RWM}_{e_{v_i,v_{i+1}}} = e^{\left(\Gamma \times N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale}\right)} + \Lambda \times \left(\widehat{CDM}_{e_{v_i,v_{i+1}}}(t) + \frac{\widehat{Q}_{e_{v_i,v_{i+1}}}(t+1)}{C_{e_{v_i,v_{i+1}}}} + \frac{\widehat{Q}_{RSU_r}(t+1)}{\sum_{e=1}^{E_{RSU_r}} C_{e_{v_i,v_{i+1}}}}\right) \tag{28}$$

The weight of roads not within RSU is given by:

If $\frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2} \geq \varepsilon$:

$$\widehat{RWM}_{e_{v_i,v_{i+1}}} = e^{\left(\Gamma \times \frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2}\right)} + \left(\Lambda \times \left(N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale} + \frac{\widehat{Q}_{e_{v_i,v_{i+1}}}(t+1)}{C_{e_{v_i,v_{i+1}}}}\right)\right) \tag{29}$$

Else if $\frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2} < \varepsilon$:

$$\widehat{RWM}_{e_{v_i,v_{i+1}}} = e^{\left(\Gamma \times N\left(\widehat{TT}_{e_{v_i,v_{i+1}}}(t)\right)_{rescale}\right)} + \left(\Lambda \times \left(\frac{\widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{2} + \frac{\widehat{Q}_{e_{v_i,v_{i+1}}}(t+1)}{C_{e_{v_i,v_{i+1}}}}\right)\right) \tag{30}$$

Based on these calculations, we can define traffic congestion on the basis of following classification:

TABLE I. TRAFFIC CONGESTION CLASSIFICATION

Speed Performance Index	Traffic State level
0-0.3	Free flow
0.3-0.5	Slight congestion

0.5-0.75	Mild congestion
0.75-1	Heavy Congestion

2) *Traffic Congestion Detection*

The traffic information is collected in a regular interval in a periodic manner and based on the information that has been generated, the congestion is checked on every road. The efficiency of congestion area can be raised by:

$$\widehat{TCD}_{e_{v_i,v_{i+1}}}(t) = \frac{\widehat{CDM}_{e_{v_i,v_{i+1}}}(t) + \widehat{ZCM}_{RSU_r}(t)}{2} = \frac{\left(\frac{\widehat{ZCM}_{RSU_r}(t) + \widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{3}\right) + \widehat{ZCM}_{RSU_r}(t)}{2} = \frac{\left(4\times\widehat{ZCM}_{RSU_r}(t)\right) + \widehat{RCM}_{e_{v_i,v_{i+1}}}(t)}{6} \quad (31)$$

D. *Protocol Stack*

The protocol stack is based on the principle of forwarding the network statistical information through radio transmission and internet routing. The protocol stack works between software designed network controllers and also with other RSU and vehicles.

The radio link control layers are used in order to access the radio part of the vehicle and RSU. The layers encapsulate the IP packet by incorporating MAC-in-MAC encapsulation mechanism. After that, RSU forwards the data packet from ingress RSU to igress RSU. This is the reason for the MAC header to be used for data delivery in mobile core network layers [18].

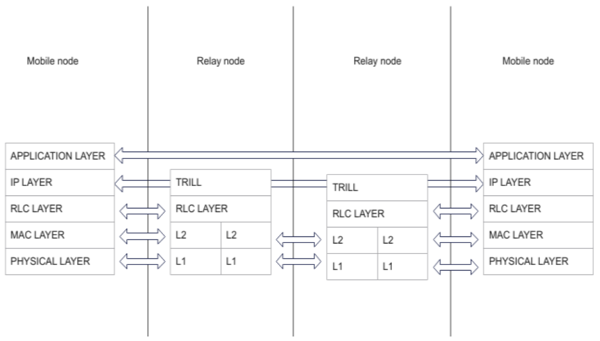

Fig. 3. Protocol stack based on RSU

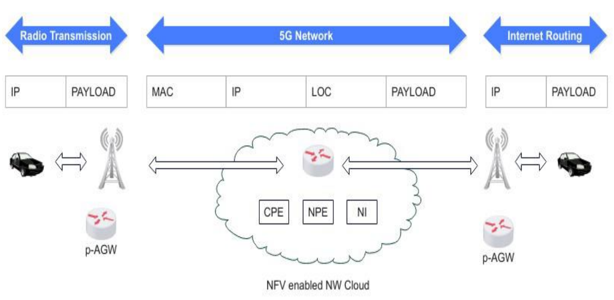

Fig. 4. Data delivery in mobile core network layers

1) *Shortest and Fastest Path Calculation*

The Dijkstra algorithm and simplified ant colony optimization algorithm are some of the ways of calculating the shortest or the fastest path by providing solutions related to various problems in the network such as identifying specific sites and the shortest paths to access them. The method is a graph search algorithm that can create a road tree. The algorithm takes use of Greedy search strategy where we calculate the path two points that depend on the feature of the GIS programs that provide maps.

TABLE II. COMPARISON OF SACO AND DIJKSTRA

Properties	Dijkstra	SACO
Ability to repeat roads	Yes	No
Number of points	Any number	Sometimes any number
Total distances	From start to all nodes	From start to end
Total time	Little	Big
Complexity	O(n2)	O(n3)
Be optimal	Yes	No

E. *Evaluation*

For controlling the level of congestion, we execute an algorithm on the TMC as well as RSU where we first create the network graphs of roads with the help of city maps. This is followed by determination of all RSU's locations by the TMC as well as the roads where the RSU's range is not reachable. The TMC then collects information regarding the traffic situation from every RSUs and vehicles. The weight of roads is calculated with the help of real time information statistics every now and then and finally the TMC inquires about the traffic congestion. Upon detection of road congestion, the TMC selects the vehicles in affected zones that require re-routing. Finally, the calculation of optimum route is performed by applying Shanon entropy method for vehicles [19].

The RSUs are set a unique ID and locations are set manually and distributed on the basis of communication range and road network dimensions for detecting nearby edges.

The RSU on the other hand collects information related to traffic in its own area then draws a sub-graph of road-network. The RSU then gathers congestion information and inflow / outflow of road traffic. The information is then sent to the TMC using different message formats. The congestion control by RSU is performed in its own area by assessing the area congestion and finally selecting vehicles based on the BFSMaxDepth algorithm. The re-routing process of RSU involves:

a) RSU inquiring about the destination of vehicles locating in its own location.

- If yes, k-shortest paths algorithms are used for calculating routes for each vehicle. Optimum route determination is using the Shanon entropy method.

b) If destination is in different location,

- RSU sends information to TMC.

- TMC determines the optimum route considering global traffic conditions .
- TMC sends information to RSU.

Algorithm 1 Congestion Control by TMC

procedure TMC FUNCTION (GRAPH)
  Graph ←Create of graph from road network
  RSU_{list} ← All RSUs and locations
  $Roads_{list}$ ← Retrieval all roads that not to be in the RSU range
  Collect traffic information from RSUs and vehicles
  $k \leftarrow 4,\ l \leftarrow 3$
  while SIMULATION **do**
    if SimulationTime % 600 s == 0 **then**
      $PauseSimulation$
      for all roads in Graph **do**
        if the road is in the RSU area **then**
          Calculate RWM_{road} by using Eq. Eqs. ( ) and ( )
        else
          Calculate RWM_{road} by using Eqs. ( ) and ( )
        end if
      end for
      for all roads in $Roads_{list}$ **do**
        Calculate congestion level by using Eq.(9)
        if $RCM_{e_{v_j,v_{j+1}}} \geq \alpha$ **then**
          $Vehicles_{list}$ ← Selection of vehicles for re-routing
        end if
      end for
      for all vehicles $\in Vehicle_{List}$ **do**
        $Road_{Current}$ ← vehicle.CurrentPosition()
        $Road_{Last}$ ← vehicle.getLastRoad()
        kSP← kShortestPaths(Graph,Origin,Dest,k)
        $Path_{new}$ ← SelectedPath(kSP)
        vehicle.SetPath($Path_{new}$)
      end for
    end if
  end while
end procedure

F. Simulation

The important aim of simulation study is to check performance of the proposed model for congestion detection and effective route planning. The simulation can also help compare the proposed model performance with regards to time taken and cost with respect to different route recommendation models. The research study is based on carrying out simulation in the following environments.

OMNeT++ simulation, which is an Integrated Development Environment (IDE) based on the

- Eclipse platform that consists of Veins simulation for showing VANET communication.
- SUMO Simulator that allows for intermodal simulation including pedestrians and has a large number of tools to create complex scenarios.
- QGIS Desktop to provide a GIS data viewer and calculate the shortest and the fastest route.

Algorithm 2 Congestion Control by RSU

procedure RSU FUNCTION(Graph)
  Collect information related to traffic from traffic lights and vehicles
  Create subgraph from Graph
  $k \leftarrow 4,\ l \leftarrow 3$
  while SIMULATION **do**
    if SimulationTime % 600 s == 0 **then**
      $PauseSimulation$
      $ZCM_{RSU_r}(t)$ ← ZoneCongestion ($subgraph$)
      $Q_{RSU_r}(t+1)$ ← TrafficFlowArea ($subgraph$)
      $Msg_{vehicle}$ ← Create $Message(ZCM_{RSU_r}(t), Q_{RSU_r}(t+1))$
      Send $Message(ZCM_{RSU_r}(t), Q_{RSU_r}(t+1))$ to TMC for calculate RWM
      for all roads in subgraph **do**
        if $ZCM_{RSU_r}(t) \geq 0.5$ **then**
          calculate TCD of road
          if $TCD \geq \alpha$ **then**
            $Vehicle_{List}$ ← $BFSMaxDepth(subgraph, road, reverse = True)$
          end if
        else
          calculate RCM of road
          if $RCM \geq \alpha$ **then**
            $Vehicle_{List}$ ← $BFSMaxDepth(subgraph, road, reverse = True)$
          end if
        end if
      end for
      for all vehicle $\in Vehicle_{List}$ **do**
        $Road_{Current}$ ← vehicle.CurrentPosition()
        $Road_{Last}$ ← vehicle.getLastRoad()
        $Msg_{vehicle}$ ← Message($Road_{Current}$, $Road_{Last}$)
        if $Road_{Last} \in Subgraph$ **then**
          if $Road_{Current} \neq Road_{Last}$ **then**
            kSP← kShortestPaths(subgraph,Origin,Dest,k)
            $Path_{new}$ ← SelectedPath(kSP)
            vehicle.SetPath($Path_{new}$)
          end if
        else
          $Path_{new}$ ← TMC ($Msg_{vehicle}$)
          vehicle.SetPath($Path_{new}$)
        end if
      end for
    end if
  end while
end procedure

IV. RESULTS AND DISCUSSION

A. Simulation Scenario

TABLE III. COORDINATES

Coordinates		
Direction	Latitude	Longitude
South-west	27.6461	85.3046
North east	27.6739	85.3582

TABLE IV. TRAFFIC DEMAND

Traffic Demand	
No of trips to generate	10
No of vehicles	20
Obstacles created	Yes

TABLE V. PLAYGROUND SIZE

Playground Size	
X-direction	8091m
Y-direction	10053m
Z-direction	50m

TABLE VI. MOBILITY

Mobility	
Congestion detection duration	73sec
Vehicle in state of halt	50sec
Connection	Proxy TCP

TABLE VII. CONGESTION AND RSU DETAILS

Congestion and RSU Details	
Location	Gwarko
Destination	Khumaltar
RSU Placement	Satdobato

B. *Environment Configuration*

The example here shows the communication between vehicles when we have a traffic congestion. The first vehicle comes to a halt for a certain time period for generating a simulation for traffic congestion. This results in vehicles sending information to one of the road side units and to other vehicles as well. The repetition of messages occurs with some particular amount of delay resulting in vehicles receiving the first wave of messages and knowing about the traffic. This results in vehicles making decisions regarding changing the routes and replaying the received messages by sending to other vehicles and calculating the distance between the next automated route.

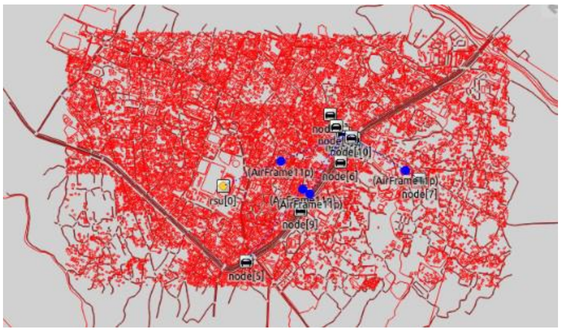

Fig. 5. Qtenv environment where a RSU is handling nodes of vehicles

The simulation can show that the traffic congestion will occur at 73 seconds as we defined earlier. The vehicles are piling as a result of an accident behind the first vehicle. The blue line indicates that the communication is occurring between vehicle to vehicle in order to notify about the traffic congestion which are called Airframes. This would allow the vehicles to either change the route or turn around. A rsu[0] component can be seen on the corner while the nodes which are moving are vehicles and the red blocks are nothing but buildings.

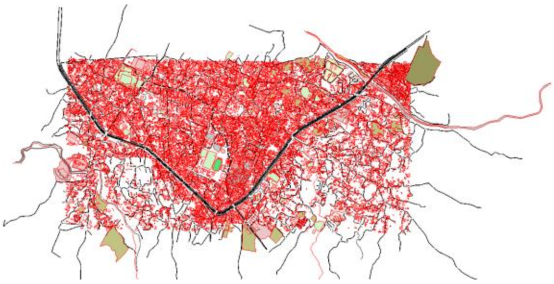

Fig. 6. SUMO Simulation

C. *Analysis / Results*

After the simulation is completed, we can check the results of the analysis from the results folder and create a new analysis file for analyzing the simulation. We can check different parameters for all the vehicles.

a) Speed

The change in speed between vehicles during the simulation is shown in the following chart:

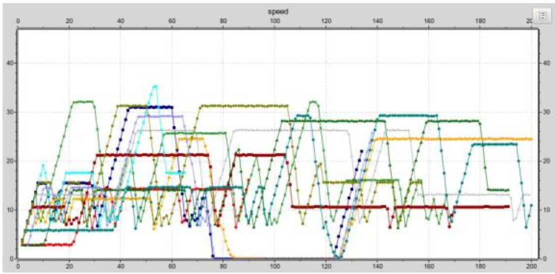

Fig. 7. Speed Results

b) Acceleration

Now looking at some vector data, we can also plot these parameters for analysis purposes. The most common vector data that can be used for analysis is acceleration among different vehicles.

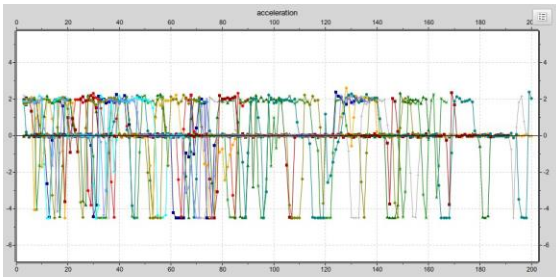

Fig. 8. Acceleration

c) Position Coordinates

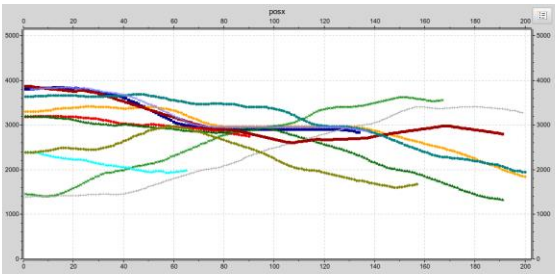

Fig. 9. Posx Comparison

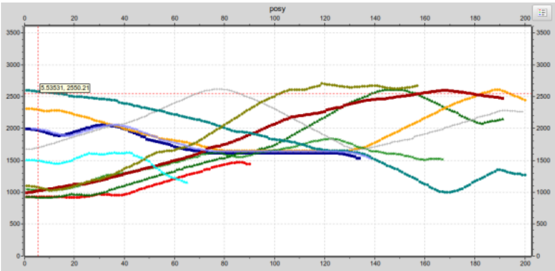

Fig. 10. Figure: Posy Comparison

d) Shortest and Fastest Path

QGIS Desktop provides advanced directions options to calculate the shortest and the fastest routes between multiple locations using ORS tools consisting of routing, isochrones and matrix calculations. These are interactive in map canvas or from point files within the processing framework producing extensive attributes as outputs that includes duration, length and start/end locations. A batch job is created for calculating the shortest and the fastest route by specifying the directions. The shortest and the fastest distance differs in some cases where the shortest distance can take longer because we may go through a city area where there can be comparatively more traffic than roads that might be longer but can take less time. The scenario below shows a condition where we have to send route information for going from Gwarko – Lagankhel – Khumaltar and finding out the respective distance and sending information based on shorter or faster distance.

Fig. 11. Shortest Path

Fig. 12. Fastest Distance

e) Isochrones

An isochrone map is important for geography and urban planning because it helps depict the areas that are possible to access from a point in time based on certain threshold provided. In this simulation, we will look at the possible area a vehicle might travel from within a time interval of 2minutes, 5 minutes and 10 minutes from every point on the layer. These are the boundary areas that will be possible to travel within the given time frame.

Fig. 13. Gwarko Isochrone

The coverage of area can also be represented in distance as a dimension rather than time to find out the isochrome map of an area within some kilometers. For this let's look at an example where we will be seeing the boundary areas upto which we can travel within 2000m.

Fig. 14. Isochrone for distance <= 2000m

D. Evaluation

The evaluation of the model is based on different scenarios where we first test the performance of the proposed model with respect to variation in traffic congestion detection (TCD) threshold. Finally, to check the appropriateness of the system, we compare it with other models with respect to travel time and fuel consumption. The road network analyzed for evaluation consist of following characteristics

a) Impact of congestion threshold

The scenario is based on the investigation of influence of dissimilarities in congestion threshold during travel period and fuel efficiency. We use the value of 0.2 to 0.8 for TCD and the results .

b) Accuracy of road weight measurement

- SSRGS also known as Single-Static Route Guidance System, where we have static metrics and are calculated as the ratio of length of a road and the maximal permitted speed.

- SDRGS also known as Single-Dynamic Route Guidance System which is the ratio of length of road and velocity of vehicles.
- MDRGS also known as Multi-Dynamic Route Guidance System including the time period spent on congestion, ratio of road density and average road density,
- MHRGS also known as Multi-Hybrid Route Guidance System including the road length, vehicles number and extra number of vehicles with corresponding weight.

The estimated weight function from the proposed model showed more accurate and effective results with the decrease in traveling time and fuel consumption compared to other standard techniques by outperforming them. With regards to mean travel period, the performance was better by 66.80%, 40.28%, 42.09%, 63.18% and 61.45 % and with regards to fuel consumption, the reduction was by 50.75%, 28.90%, 30.49%, 47.90% and 45.06% compared to above models. Hence, we can say that the proposed method is effective enough for reducing the road congestion and traffic flow within different locations.

E. Results Validation

Route 1:

Situation	Time	Length	Optimized Results	
Normal Route	14 minutes	4.2 km	Time	Length
Fastest Route	11 minutes	3.285 km	21.42%	21.78%
Shortest	11.5 minutes	3.112 km	17.85%	25.90%

Route 3:

Situation	Time	Length	Optimized Results	
Normal Route	14 minutes	4.2 km	Time	Length
Fastest Route	11.6 minutes	4.442 km	17.14%	5.71% (Increase)
Shortest	13.8 minutes	4.247 km	1.47 %	1.2 % (Increase)

Route 4:

Situation	Time	Length	Optimized Results	
Normal Route	14 minutes	4.2 km	Time	Length
Fastest Route	14.5 minutes	4.009 km	3.5% (Increase)	4.54%
Shortest	16.5 minutes	3.562 km	17% (Increase)	15.19%

Route 2:

Situation	Time	Length	Optimized Results	
Normal Route	14 minutes	4.2 km	Time	Length
Fastest Route	13.2 minutes	3.609 km	5.71%	14%
Shortest	14.1 minutes	3.428 km	0.71% (Increase)	25.90%

Hence, we can see that with better route recommendations, we can see that the travel time can be reduced by up to 21.42% and length can be reduced by up to 25.90%. Therefore, the system can help reduce the travel time and fuel consumption with better travel route planning and route recommendation.

V. CONCLUSION

This research paper provides a route guidance system working in multi-layers using fog-based cloud architecture for real time traffic congestion detection and route recommendation. The architecture is based on incorporating traffic hazards, congested road detection and its effects and possible future traffic status based on real time prediction of dynamic traffic status using a multi-fitness function known as road weight measurement within a range of RSUs components. The method also helps detect road congestion within an area and provide route recommendation based on vehicular communication and routing information for easing the concern of traffic congestion issues using optimum path calculation for finding the shortest and the fastest routes. The open source Python based program is used for connection and control of traffic simulation using SUMO simulator for verifying the model on reducing the fuel consumption and travel time in-comparison to other guidance system.

REFERENCES

[1] S. Sun, J. Chen, and J. Sun, “Traffic congestion prediction based on GPS trajectory data,” Int. J. Distrib. Sens. Netw., vol. 15, no. 5, p. 155014771984744, 2019.

[2] Li Q and Huang L. “A map matching algorithm for GPS tracking data,” Acta Geodaet Cartograph Sinica, vol. 39, no. 2, pp 207-212, 2010.

[3] M. Nikolić and J. Jović, “Implementation of generic algorithm in map-matching model,” Expert Syst. Appl., vol. 72, pp. 283–292, 2017.

[4] C. Y. Goh, J. Dauwels, N. Mitrovic, M. T. Asif, A. Oran, and P. Jaillet, “Online map-matching based on Hidden Markov model for real-time traffic sensing applications,” in 2012 15th International IEEE Conference on Intelligent Transportation Systems, 2012, pp. 776–781.

[5] A. Luo, S. Chen, and B. Xv, “Enhanced map-matching algorithm with a hidden Markov model for mobile phone positioning,” ISPRS Int. J. Geoinf., vol. 6, no. 11, p. 327, 2017.

[6] G. R. Jagadeesh and T. Srikanthan, “Online map-matching of noisy and sparse location data with hidden Markov and route choice models,” IEEE Trans. Intell. Transp. Syst., vol. 18, no. 9, pp. 2423–2434, 2017.

[7] Chien, S., Yang, Z. and Hou, E., 2022. Genetic Algorithm Approach for Transit Route Planning and Design.

[8] A. Karbassi and M. Barth, “Vehicle route prediction and time of arrival estimation techniques for improved transportation system management,” in IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No.03TH8683), 2004.

[9] R. Claes and T. Holvoet, “Ant colony optimization applied to route planning using link travel time predictions,” in 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and Phd Forum, 2011.

[10] S. Chowdhury, University of Auckland, N. Giacaman, and University of Auckland, “En-Route Planning of Multi-Destination Public-Transport Trips Using Smartphones,” J. Public Trans., vol. 18, no. 4, pp. 31–45, 2015.

[11] H. Huang, D. Bucher, J. Kissling, R. Weibel, and M. Raubal, “Multimodal Route Planning With Public Transport and Carpooling,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 9, pp. 3513–3525, 2019.

[12] T. Liebig, N. Piatkowski, C. Bockermann, and K. Morik, “Dynamic route planning with real-time traffic predictions,” Inf. Syst., vol. 64, pp. 258–265, 2017.

[13] P. Perez-Murueta, A. Gómez-Espinosa, C. Cardenas, and M. Gonzalez-Mendoza Jr, “Deep learning system for vehicular re-routing and congestion avoidance,” Appl. Sci. (Basel), vol. 9, no. 13, p. 2717, 2019.

[14] S. Koh et al., “Real-time deep reinforcement learning based vehicle navigation,” Appl. Soft Comput., vol. 96, no. 106694, p. 106694, 2020.

[15] H. Noori and M. Valkama, “Impact of VANET-based V2X communication using IEEE 802.11p on reducing vehicles traveling time in realistic large scale urban area,” in Proc. Int. Conf. Connected Vehicles Expo (ICCVE), Dec. 2013, pp. 654–661.

[16] Lanning, D.R., Harrell, G.K. and Wang, J., 2014, March. Dijkstra's algorithm and Google maps. In Proceedings of the 2014 ACM Southeast Regional Conference (pp. 1-3).

[17] Rezaei, M., Noori, H., Mohammadkhani Razlighi, M. and Nickray, M., 2021. ReFOCUS+: Multi-Layers Real-Time Intelligent Route Guidance System With Congestion Detection and Avoidance. IEEE Transactions on Intelligent Transportation Systems, 22(1), pp.50-63.

[18] Din, S., Paul, A. and Rehman, A., 2019. 5G-enabled Hierarchical architecture for software-defined intelligent transportation system. Computer Networks, 150, pp.81-89.

[19] Pan, J., Popa, I., Zeitouni, K. and Borcea, C., 2013. Proactive Vehicular Traffic Rerouting for Lower Travel Time. IEEE Transactions on Vehicular Technology, 62(8), pp.3551-3568.

[20] Yue, W., Li, C., Mao, G., Cheng, N. and Zhou, D., 2021. Evolution of road traffic congestion control: A survey from the perspective of sensing, communication, and computation. China Communications, 18(12), pp.151-177.