VANET-based Smart Navigation for Emergency Evacuation and Special Events

Ahmed Elbery
School of Computing
Queen's University
Kingston, Ontario, Canada
aelbery@cs.queensu.ca

Hossam S. Hassanein
School of Computing
Queen's University
Kingston,Ontario, Canada
hossam@cs.queensu.ca

Nizar Zorba
EE Department
Qatar University
Doha, Qatar
nizarz@qu.edu.qa

Hesham A. Rakha
CEE Department
Virginia Tech
Blacksburg, Virginia, USA
hrakha@vt.edu

***Abstract*—In this paper we propose, develop, and analyze the performance of a new system-optimum navigation model that utilizes Vehicular Ad-hoc Networks (VANETs), linear programming optimization, and stochastic routing to efficiently and smartly navigate vehicle crowds in case of an emergency evacuation or after special events. The objective of the proposed system is to clear the network in a shorter time by better utilizing the network resources while taking into consideration the road capacities. In this model, road links are weighted based on travel time. Road link capacities and current traffic conditions are used as constraints in the optimization problem. Vehicles are employed as sensors to compute travel times of the links and send this information to the Traffic Management Center (TMC) in real-time. The TMC periodically optimizes the traffic assignment. Subsequently, routes for vehicles are created/updated based on the latest optimized assignments. To test the model, a real network with calibrated traffic is used. The proposed model is compared to the Sub-population Feedback Dynamic Time-dependant Assignment (SFDTA) navigation. Moreover, we analyze its sensitivity to the re-optimization interval at different traffic demand levels. The results show that the proposed system decreases the network-wide travel time and is successful in clearing the network earlier especially in the case high vehicle traffic demands.**

***Index Terms*—VANET-based Navigation, Smart Cities, Crowd Management, Stochastic Routing, Constrained Routing.**

I. Introduction

Intelligent Transportation Systems (ITSs) employ advanced navigation techniques to improve mobility by reducing travel time [1], [2], fuel consumption, and the environmental impact from the transportation sector [3]. However, navigation systems that perform well in day-to-day traffic conditions are not suited for some non-recurrent events such as large sporting events or emergency evacuation, in case of natural disasters. In such events, there is a large number of vehicles (vehicular crowds) needs to be routed and exit the crowd area.

Most of the current navigation techniques will not work efficiently in such situations because most of them utilize best path routing models. The main problem with best path routing is the single cost function. These techniques do not account for other parameters such as road capacity, traffic volume, or underutilized alternative routes. Moreover, at any given time, shortest path navigation techniques provide the same guidance for all vehicles based on their destinations, causing the shortest path to collapse, while other longer paths are underutilized. It also can result in route oscillations and unstable global traffic behavior [4].

The challenge in non-recurrent events is two-fold. First, the sheer volume of traffic that needs to leave the event area can cause serious congestion and may result in network grid-lock. Consequently, adversely affecting the mobility (i.e., increasing travel time, fuel consumption, and emission levels). Second, the road networks are not designed to support such a high traffic demand, basically, because of the required high road capacities [5] and special traffic control techniques [6].

Thus, vehicle crowds combined with road network resource constraints, bring forward the need for efficient and smart management techniques that better utilize the network facilities while taking into consideration the road capacity constraints.

Inspired by Vehicular Ad-hoc Networks (VANETs) [7] the advancement in information technology, vehicular navigation based on real-time information shows potential benefits for crowd management [8] to reduce travel time [2] and energy/fuel consumption [9], [10]. Utilizing VANETs in real-time navigation brings new opportunities to address the problem of vehicular crowd navigation.

Therefore, this paper focuses on utilizing VANET communication, to efficiently and smartly route vehicular crowds to minimize the network-wide travel time and to clear the network faster. The contributions of this paper are:

- Proposing and developing a system-optimum navigation model for vehicular crowds,
- Using a real network and a calibrated traffic to test the proposed model, and
- Performing sensitivity analysis against two system parameters; traffic load and re-optimization interval.

Firstly, the proposed system uses vehicles as network sensors and utilizes VANETs as an infrastructure to collect road state-conditions (traffic volume and travel time on each road segment) in real-time. The collected information is used along with historical information to optimize the network-wide vehicular traffic-assignment using linear programming (LP) [11]. In the optimization problem, the road capacities and current traffic loads are used to constrain the network congestion. Then, a stochastic route construction algorithm is used to create/update the vehicles' routes.

The main idea behind the proposed model is to efficiently utilize all the network resources, allowing vehicles traveling

978-1-7281-1016-5/19/$31.00 ©2019 IEEE

from the same origin to the same destination, at the same time to be assigned different routes. Consequently, in order to minimize congestion and improve the network-wide mobility, some vehicles may be subjected to longer travel times.

Secondly, to test the proposed model, we use the FIFA World Cup 2022, which will be held in Qatar, as a case study. The Doha road network in Qatar, is implemented and used to compare our proposed system to the SFDTA user-optimum traffic assignment.

Thirdly, we perform system sensitivity analysis against the re-optimization interval at different traffic loads.

In terms of previous related work, there is some research effort [1], [2] in addition to online services such as Google Maps and INRIX [12]. However, all of these utilize the shortest path techniques (which are user equilibrium models) or at most provide alternative routes without considering the system-wide performance or trying to minimize network-wide congestion. Both [4] and [13] developed a heuristic approach to assign vehicles to different routes. Our proposed model is a step up because we use matured optimization model to assign routes. We also account for the current network states in assigning the traffic. The authors in [9] developed a system-optimum eco-routing model to minimize fuel consumption. Our proposed model focuses on clearing the network faster. Moreover, we focus on improving navigation in special non-recurrent events.

The remainder of the paper is organized as follows. Section II is the network model and problem definition. Section III details the proposed system and its components. Section IV describes our case study using Doha network in Qatar and the results. The final conclusions are presented in V.

II. Network Model and Problem Definition

Fig. 1 shows the network model being studied in this paper. We assume that there is a vehicular crowd, represented by the red area in Fig. 1 which is a stadium where sport matches are held. The destinations of these vehicles are distributed across the network. Consequently, these vehicles are grouped into a set of Origin-Destination (OD) pairs based on their destinations. Each OD pair represents a traffic flow from the stadium to a given destination, as shown by the yellow arrows in Fig. 1. Thus, the network has a set $\mathcal{F}$ of f concurrent flows. Each flow $k \in \{1, 2, ..., f\}$ has V_k cars that need to leave within a time interval τ. Therefore, each flow $k \in \mathcal{F}$ has a traffic rate q^k (in vehicles per hour veh/h), where $q^k = V_k/\tau$. We assume these vehicles are connected and follow the route recommendations they receive from the TMC. We call these flows the controlled traffic (CT).

In addition to CT, we assume that there is a background traffic (BT) that is traversing the network. BT vehicles use the shortest path routing.

The road network is represented by a directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L}, \mathcal{C})$, where $\mathcal{N} = \{i : i = 1, 2, ..., n\}$ is a set of n nodes and $\mathcal{L}$ is a set of l directed links (road segment), i.e., $\mathcal{L} = \{L_{i,j} : i, j \in \mathcal{N}\}$, where $L_{i,j}$ is the road segment from node i to node j. Each road segment has a capacity $C_{i,j}$ which is the maximum traffic flow rate that can enter/exit

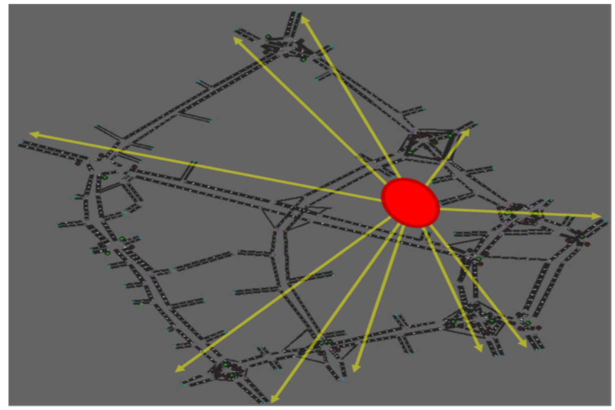

Fig. 1: The Network Model

this link. And each segment has time dependent cost $T_{i,j}$ that represents its estimated travel time which depends on the traffic conditions on segment $L_{i,j}$.

Our objective is to assign routes to the CT vehicles in such a way that minimizes the network-wide travel time, consequently, clearing the network faster. The next section describes the proposed framework to achieve this objective.

III. Vehicular Crowd Management Framework

This section describes the proposed framework and its components, including data collection and processing, the optimization problem formulation and its constraints, and the stochastic route construction algorithm.

In contrast to deterministic shortest path routing, the proposed system applies stochastic system-optimum routing. The main goal is to utilize all available network resources by routing vehicles (going to the same destination) simultaneously through alternative routes while maintaining traffic load on each road segment within its capacity. To achieve these objectives, the system uses vehicles as sensors to collect network-state information in real-time. The collected data is processed and fused with historical data. Then, this data is used to optimize the traffic-assignment. By traffic-assignment we simply mean the portion of each flow k that will be assigned to each road segment (described in detail in subsection III-B).

After computing the optimized traffic-assignment, the TMC builds stochastic routes for vehicles upon receiving routing or rerouting requests. Fig. 2 shows the three components of the system.

A. Data Collection and Processing

As shown in Fig. 2 vehicles can communicate to the TMC using either Vehicle-to-Infrastructure (V2I) or Vehicle-to-Vehicle (V2V) multi-hop connection. In addition to vehicle communication facilities, we assume that vehicles are equipped with a Global Positioning Systems (GPSs), so a vehicle can identify which link it is currently traversing. These facilities are used to collect and communicate the information needed by the TMC that computes two parameters for each road link $L_{i,j}$, namely, travel time $T_{i,j}$ and traffic load $\zeta_{i,j}$ on each link.

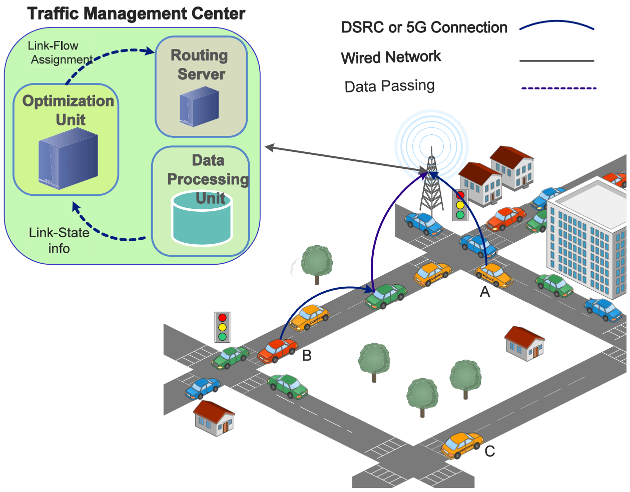

Fig. 2: System Components

Whenever, a vehicle enters a new road link, it initializes the travel time to the current time, and when it exits this road segment, it computes its travel time on this segment. Subsequently, it builds a message with this information along with the current time as well as the next link to be traversed. The message format is shown in Fig. 3. The vehicle then tries to send this message to the TMC using either V2I (such as car A in Fig. 2) or V2V (such as car B in Fig. 2). If the car does not have a connection (car C in Fig. 2), it will store the message until reaching an area that is covered by the wireless network, or finding a vehicle-to-vehicle path to the TMC.

When the TMC receives a message, it finds whether it is expired by comparing the current time to the message time stamp. The message expiry interval is assumed to be 180 seconds. The unexpired messages are used to update the link travel times. The TMC fuses the new travel with the historical data using exponential smoothing as shown in Eq. 1. In our case we used an exponential factor α of 0.2.

$$T^{t}_{i,j} = \alpha \ \widehat{T}_{i,j} + (1-\alpha) \ T^{t-1}_{i,j} \tag{1}$$

where $T^{t}_{i,j}$, $T^{t-1}_{i,j}$, and $\widehat{T}_{i,j}$ are the new travel time of link $L_{i,j}$, the old value, and the received update, respectively.

To compute the currant traffic load on each road link, the system uses the "$Next\ LinkID$" field in the received messages. It maintains a linked-list for each road segment that stores the times at which vehicles enter this link in the last 180 seconds (which is the same as the expiry time). Periodically, the system computes the average inter-arrival interval and the traffic volume on each segment. Then, it fuses this new rate

0	8	16	24	32	40	48	56
Code							
Seq_Num				VehicleID			
LinkID				Link Travel Time			
Currnt Time				Next LinkID			

Fig. 3: The Link-State Message Format

with the link history in the same manner as the travel time. The importance of maintaining a linked-list for each segment is to enable the system to consider the delayed packets within the last 180 seconds interval.

The link-state information is then used by the traffic assignment optimization module.

B. Traffic Assignment Optimization

The optimization problem is formulated to allow vehicles from the same traffic flow k to use different paths at the same time. This allows load-balancing the traffic among different alternative routes.

The Optimization Problem

The objective of the linear program is to minimize the network-wide travel time by computing the link-flow assignment parameters for each link-flow pair. Vehicles in the k^{th} flow whose rate is q^k can be assigned different road links. The link-flow assignment denoted by $q^k_{i,j}$ is the portion of this traffic flow k that will go through $L_{i,j}$.

Based on this definition, the optimization problem can be formulated as:

$$\begin{aligned}
&\underset{q^k_{i,j}}{\text{minimize}} \sum_{i=1}^{n}\sum_{j=1}^{n} T_{i,j} \sum_{k=1}^{f} q^k_{i,j} \\
&\text{subject to :} \\
&\sum_{d=1}^{n} q^k_{i,d} - \sum_{s=1}^{n} q^k_{s,i} = 0, \qquad \text{if } i \text{ is an intermediate node} \\
&\sum_{d=1}^{n} q^k_{i,d} - q^k = 0, \qquad \text{if } i \text{ is the source of the } k^{th} \text{ flow} \\
&q^k - \sum_{d=1}^{n} q^k_{s,i} = 0, \qquad \text{if } i \text{ is the destination of the } k^{th} \text{ flow} \\
&\zeta_{i,j} + \sum_{k=1}^{f} q^k_{i,j} \leqslant C_{i,j} \quad \forall \ L_{i,j} \in \mathcal{L}, \\
&q^k_{i,j} \geq 0.
\end{aligned} \tag{2}$$

where $i, s, d \in \mathcal{N}$ such that $L_{i,d}$ is a link exiting node i, and $L_{s,i}$ is a links entering node i. If the network has f flows and l links, the linear program calculates $l.f$ argument variables.

In the problem formulation, there are three sets of constraints. The first is to satisfy the route continuity. The second is to guarantee that total flow rate on each link does not exceed its capacity, and finally the positive traffic assignment.

The Route Continuity Constraints: This set of constraints is represented by the first three constraints in Eq. 2. The objective of this constraint is to ensure that the computed traffic-flow assignment results in connected routes for each flow k to its destination.

The route continuity condition is achieved by enforcing the individual flow balance at each node, which can be formulated as follows.

- For each intermediate node i, and for each individual flow k, the summation of the sub-flows of the k^{th} flow entering this i^{th} node must be equal to the summation of the sub-flows of the k^{th} flow exiting this node.
- For each source (or destination) node, we assume there is another fictitious source (or destination) sending q^k to (or receiving q^k from) it.

The Link Capacity Constraints: For each directed link $L_{i,j} \in \mathcal{L}$, the total flow rate traversing it should not exceed its capacity $C_{i,j}$. The total traffic rate on the link is calculated as the summation of its current load $\zeta_{i,j}$ (computed based on the received link-state information messages) and all the flows/sub-flows assigned to this link.

The Positive Assignment Constraints: The last constraint is to allow only positive values for $q^k_{i,j}$, to make it consistent with the directed links.

The system initializes link costs based on the maximum speed for each link. Then, it uses the latest updated link information to periodically optimize the link-flow assignment. This interval is called the Re-optimization Interval. The link-flow assignment matrix is used to build or update stochastic routes when requested by a vehicle.

C. Building Vehicle Routes

When a vehicle starts its trip, it sends a route request to the TMC. The TMC calls the route construction algorithm, shown in Algorithm 1, which takes four input parameters: the flow number k, the start node s, the destination node d, and the link-flow assignment vector $q^k_{i,j}$ for the k^{th} flow. The algorithm stochastically creates a route as follows. It starts with the vehicle's origin node s, and uses it as the current node i. Then, it finds all the links going out of this node i, the link set $\check{L} = \{L_{i,j} : L_{i,j} \in \mathcal{L}\}$. For each link $L_{i,j} \in \check{L}$, it computes $p_j = q^k_{i,j} / \sum_{L_{i,d} \in \check{L}} q_{i,d}$, which is the probability to select this link as the next link in the route. Then, based on the computed values p_j, the algorithm stochastically selects one of these links to be the next link in the route. After adding the selected link to the route, the algorithm uses its end node j as the current node i, and repeats the process until reaching the vehicle's destination.

Algorithm 1

1: **procedure** BUILD A VEHICLE ROUTE(k,$q^k_{i,j}$, s, d)
2: $\quad R \leftarrow \phi$
3: $\quad i \leftarrow s$
4: $\quad$ **while** $i \neq d$ **do**
5: $\quad\quad R \leftarrow (R, i)$
6: $\quad\quad \check{L} \leftarrow L_{i,j} : L_{i,j} \in \mathcal{L}, L_{i,j}$ *is non-restricted*
7: $\quad\quad$ **for each** $L_{i,j} \in \check{L}$ **do**
8: $\quad\quad\quad$ *compute* p_j
9: $\quad\quad r \leftarrow randomnumber$
10: $\quad\quad$ **for** `j=1` **to** `sizeof` $\check{L}$ **do**
11: $\quad\quad\quad$ **if** $\sum_{\check{j}=1}^{j} p_{\check{j}} \leqslant r$ **then**
12: $\quad\quad\quad\quad i \leftarrow j$
13: $\quad\quad\quad\quad$ *break*
14: $\quad$ **return** R $\quad\quad$ ▷ Return the computed route.

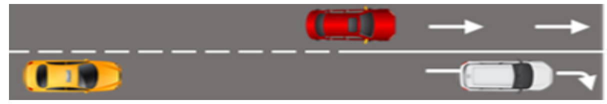

Fig. 4: Lane Striping and Re-routing (The red and white cars cannot change the next link in their routes)

D. Vehicle Re-routing

Vehicle re-routing is the process of updating the vehicle route while it is moving to react to the new traffic conditions. This process is challenging because: 1) road network moving restrictions, and 2) zero value traffic assignment.

Firstly, in road networks, there are some restrictions on moving a vehicle from one link to another. In other words, a vehicle on a road segment might not be able to move to another segment despite that the two segments are connected in the appropriate directions. Such restrictions include, but not limited to, turn prohibition and lane striping.

Turn prohibition at intersections indicate which direction a vehicle cannot move (left, right, and/or U-turns). To deal with this prohibition in our model, when building the route or updating it, $\check{L}$ in step 6 within Algorithm 1 is defined as the set of links that the subject vehicle can move to (from its current link). This is why we add the non-restricted condition in this step in Algorithm 1.

Lane striping poses another re-routing challenge. When a vehicle approaches the end of a road segment, it has to be in the appropriate lane for its direction of travel. After a given point on the segment, marked by the dashed striping line in Fig. 4, a vehicle cannot move to another lane. So, after passing this point and before leaving this link, if a vehicle received a better route it may not be able to update its current one because of this lane restriction. To overcome this issue, when re-routing a car, we have to start from the road segment after the vehicle's next segment, we call this the re-routing start link (RSL). This way, we make sure that the lane striping on the current link will not introduce any restriction, moreover, on the next link, this vehicle will take the appropriate lane for the new route.

The second problem with vehicle re-routing is the traffic assignment for the RLS. More specifically, this problem happens if, in the latest optimized assignment, $q^k_{i,j} = 0$ for the RSL (i.e., RSL = $L_{i,j}$ and $q^k_{i,j} = 0$). In this case, the re-routing algorithm will not be able to find a route for this vehicle from the RSL, and the vehicle will continue following its current route until the next re-routing time.

IV. Simulation and Results

To test the performance of the proposed model, we developed it within a microscopic traffic simulator, namely, the INTEGRATION software [14]. INTEGRATION is an agent-based microscopic traffic simulation and assignment framework. It is characterized by its accuracy that comes from its microscopic nature. It has a time granularity of 0.1 seconds that allows it to accurately capture the vehicle steady-state car-following behavior which results in accurate estimation of travel time. It also accounts for traffic lights and their impact on the travel time. Moreover, its microscopic nature enables it to accurately model vehicle queues and congestion which are important parameters in link travel time computation.

For the optimization purpose, we use the CPLEX Optimizer [15] to optimize the traffic assignment.

To study the efficiency of the proposed framework, we compare it to the Sub-population Feedback Dynamic Time-based traffic Assignment (SFDTA) [14] which uses the shortest path routing based on the dynamic link travel times since this would best reflect the state-of-practice routing. SFDTA also tries to overcome the shortest path routing problem and utilize the network resources by dividing the traffic into 5 sub-populations, each sub-population is routed at different time of the routing/re-routing interval. This way, it can utilize alternative routes [14].

Regarding the network and the traffic, we use the FIFA World Cup 2022 event which will be held in Qatar as a case study. After football matches, we expect a large number of vehicles need to leave the stadium's parking lot, resulting in a vehicle crowd surrounding the stadium area. The following subsection describes the network and traffic demand setting.

A. Network and Traffic Setup

To achieve realistic results, a part of Doha city in Qatar, shown in Fig. 1, with calibrated background vehicular traffic demands is used. To build this network, data were collected from different sources that include road network Geographic Information System (GIS) Shapefile, intersection data from OpenStreetMap (OSM), Google Maps, and ArcGIS. The Doha city shapefile was used to generate the network nodes and links. OSM data were used to extract intersection traffic control information including the traffic control methods (stop sign, yield sign, or traffic lights). Google Maps and ArcGIS were utilized for validating road attributes including the number of lanes, one-way streets, and speed limits for each road segment. The resulting simulation network has 169 nodes, 301 road segments, and 11 traffic lights.

We assume that the red area in Fig. 1 is a stadium from which vehicles belong to the controlled traffic (CT) will depart to different network points (as shown by the yellow arrows). This traffic is distributed over 10 network exit points. In regrades to background traffic (BT), which represents the regular traffic traversing the network, it was calibrated using the maximum likelihood technique based on car counts data collected from the OSM website. The BT rate in the first row ($S1$) in Table I is 10% of the calibrated traffic. In this scenario $S1$, we assume that there are 2800 cars as the CT. The rates in $S1$ is multiplied by scaling factors 2 through 5 to compute the higher traffic rates in $S2$ though $S5$. We assume that these vehicles should leave within one hour with uniform inter-departure intervals. This way, these car counts can be converted into traffic rates.

Table I: Traffic Levels.

Scenario #	Controlled Traffic	Background Traffic	Total
S1	2800	1603	4403
S2	5600	3383	8983
S3	8400	5245	13645
S4	11200	7127	18327
S5	14000	9085	23085

Increasing the traffic more than $S5$ results in optimization failure which means the traffic rate becomes higher than the network capacity.

B. Evaluation Metrics

Our main objective is to minimize the network clearance time by minimizing network-wide travel time. So, we use the following two metrics: (1) ***Network Clearance Time*** is the time at which the last vehicle reached it destination and the network is completely cleared, and (2) ***Average Vehicle Travel Time*** is computed as the summation of all vehicles' travel times divided by the number of vehicles, i.e., $T_{av} = \frac{\sum_{v \in V} T_v}{||V||}$. The travel time for each individual vehicle T_v is the time it takes from its scheduled departure time to the time it reaches its destination.

C. Simulation Results

We run each of the 5 traffic levels using both the proposed model (Stochastic LP) and SFDTA. To have statistically significant results, each scenario is run 15 times with different seeds and the average evaluation metrics are computed.

1) Network Clearance Time: Fig. 5 compares the network clearance time for both the SFDTA and our stochastic LP routing. It demonstrates that: (1) at low traffic demand levels ($S1$, $S2$), there is no significant difference between the two routing techniques at 0.05 confidence level, (2) for the higher traffic levels ($S3$, $S4$, $S5$), the proposed model reduces the network clearance time by 28%, 32%, and 38%, respectively. These two findings demonstrate the importance of the proposed model in the case of large events or evacuation, which are the typical scenarios for which it is proposed.

2) Average Vehicle Travel Time: Fig. 6 shows the average travel time for CT cars. It is clear that the proposed model is successful in decreasing the average network wide average travel time except for the lower traffic demand level ($S1$) where the CT travel time increased by approximately 9%. The reason for this increase in the CT travel time is that for the low traffic rates, the network is not congested. And, in order to avoid causing congestion on the shortest paths, the stochastic LP routing assigns the CT vehicles longer routes that increases the travel time. While, in the higher traffic rates, the congestion level becomes higher and the shortest path becomes congested which makes the stochastic LP more beneficial.

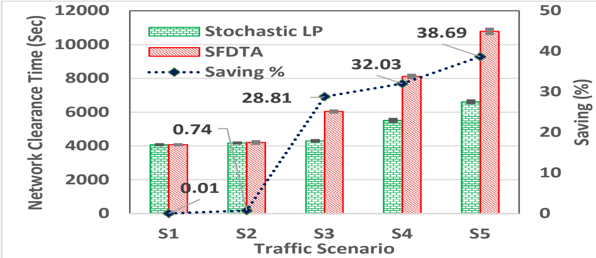

Fig. 5: Network Clearance Time

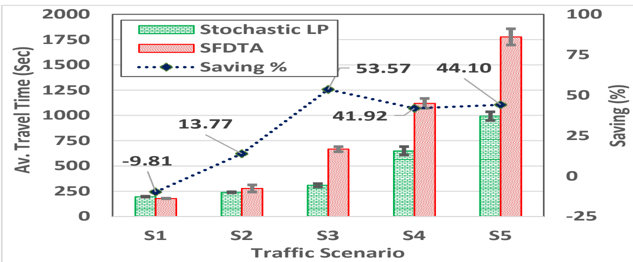

Fig. 6: Average Travel Time

3) Sensitivity analysis: This section investigates the clearance time sensitivity to re-optimization interval. We ran extensive simulations at different traffic demand rates and different re-optimization intervals.

Fig. 7 shows the average network clearance time versus the re-optimization interval (60, 120, 240, 360 and 600 seconds) for the 5 traffic cases. It is clear that, at low traffic demands (S1, S2), changing the re-optimization interval does not have significant effect on the network clearance time. This can be reasoned to the fact that, at these traffic demands, network conditions do not change significantly, and there is no congestion occurring in the network that needs re-optimization.

However, as the traffic demand increases, the need for shorter re-optimization interval becomes higher to adapt to the dynamic network-state conditions.

V. Conclusion

In this paper, we develop and test a VANET-based navigation system that can enhance navigation in the large gathering events and emergency evacuation cases. In this system, vehicles are employed as network sensors, and VANET is used as infrastructure to collect the network-state conditions in real-time. The collected data is utilized to optimize the link-flow assignment at the TMC, then the TMC stochastically assigns routes to vehicles.

The micro-simulation of the proposed model shows its superiority to currently available navigation models, as SFDTA, in the case of high traffic demands.

A future extension of this work is to study the effect of route acceptability probability on the system performance. Additionally, we plan to consider the communication network load and to study the communication requirements. Moreover, studying the sensitivity of the proposed navigation system to the communication reliability stands as a potential extension.

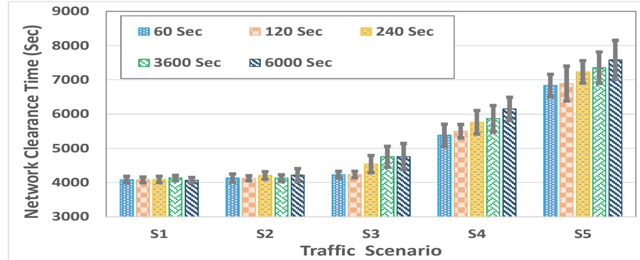

Fig. 7: Network Clearance Time vs. Re-optimization Interval

Acknowledgment

This work was made possible by National Priorities Research Program (NPRP) grant No. NPRP 9-185-2-096 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] Q. Qin, M. Feng, J. Sun, and B. Sun, "Prediction of road resistance based on historical/real-time information and road quality," in *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Aug 2015, pp. 1073–1077.
[2] Y. Regragui and N. Moussa, "Investigating the impact of real-time path planning on reducing vehicles traveling time," in *2018 International Conference on Advanced Communication Technologies and Networking (CommNet)*, April 2018, pp. 1–6.
[3] A. Elbery and H. Rakha, "City-wide eco-routing navigation considering vehicular communication impacts," *Sensors*, vol. 19, no. 2, 2019. [Online]. Available: http://www.mdpi.com/1424-8220/19/2/290
[4] J. Pan, M. A. Khan, I. S. Popa, K. Zeitouni, and C. Borcea, "Proactive vehicle re-routing strategies for congestion avoidance," in *2012 IEEE 8th International Conference on Distributed Computing in Sensor Systems*, May 2012, pp. 265–272.
[5] Q. Z. Xiongfei Zhang and Q. Luo, "Evaluation of transportation network reliability under emergency based on reserve capacity," *Journal of Advanced Transportation*, 2019.
[6] X. Zhao, M. jie Feng, H. ying Li, and T. Zhang, "Optimization of signal timing at critical intersections for evacuation," *Procedia Engineering*, vol. 137, pp. 334 – 342, 2016, green Intelligent Transportation System and Safety.
[7] "IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture," *IEEE Std 1609.0-2013*, pp. 1–78, March 2014.
[8] H. Hassanein, N. Zorba, S. Han, S. S. Kanhere, and M. Shukair, "Crowd management," *IEEE Communications Magazine*, vol. 57, no. 4, pp. 18–19, April 2019.
[9] A. Elbery and H. A. Rakha, "A novel stochastic linear programming feedback eco-routing traffic assignment system," in *Transportation Research Board 96th Annual Meeting, TRB2017*, 2017.
[10] A. Elbery and H. A. Rakha, "VANET communication impact on a dynamic eco-routing system performance: Preliminary results," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2018, pp. 1–6.
[11] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear programming and network flows*. John Wiley & Sons, 2011.
[12] INRIX. http://INRIX.com. Accessed: April, 2019.
[13] G. J. Lim, M. Rungta, and M. R. Baharnemati, "Reliability analysis of evacuation routes under capacity uncertainty of road links," *IIE Transactions*, vol. 47, no. 1, pp. 50–63, 2015.
[14] H. A. Rakha, K. Ahn, and K. Moran, "INTEGRATION framework for modeling eco-routing strategies: Logic and preliminary results," *International Journal of Transportation Science and Technology*, vol. 1, no. 3, 2012.
[15] CPLEX optimizer. https://www.ibm.com/analytics/cplex-optimizer. Accessed: April, 2019.