IoT-Based Crowd Management Framework for Departure Control and Navigation

Ahmed Elbery, *Member, IEEE*, Hossam S. Hassanein, *Fellow, IEEE*, Nizar Zorba, *Senior Member, IEEE*, and Hesham A. Rakha, *Fellow, IEEE*

***Abstract*—This paper exploits crowdsensing to propose a novel IoT-Based Vehicle Crowd Management (IoT-VCM) framework. By efficiently managing vehicle departures and navigation, the IoT-VCM clears the network in a shorter time, while maintaining the network at low congestion levels to reduce the average travel time. To compromise between these conflicting objectives, the proposed system encompasses two subsystems that work in harmony, namely; the Travel-Time System-Optimum Navigation (TTSON) and the Vehicle Departure Control (VDC). The IoT-VCM uses different network sensory devices (connected vehicles and smartphones) to collect network information that is fused to compute the current road state conditions, based on which, the VDC determines the allowable vehicle departure rates, and the TTSON optimizes their navigation. The proposed system is developed in a microscopic traffic simulator and tested on a calibrated simulated real network. The IoT-VCM controller is compared to the state-of-the-art techniques reported in the literature, namely the dynamic time-dependent incremental user-optimum traffic assignment.**

***Index Terms*—Crowd management, IoT, system optimum navigation, vehicle departure control, stochastic routing, constrained routing.**

I. Introduction

In RECENT years, vehicular crowds have become prevalent in urban areas. Sports events at stadiums are typical cases of vehicular crowds where a large number of attendees (that may exceed 100 000) gather in a small area. A large portion of those attendees drive to these events, which means tens of thousands of vehicles need to leave the area after the event. Such vehicular crowd will cause severe congestion if they are not smartly and efficiently controlled in both departure sequencing and traffic assignment. In addition to sports events, other types of social events, such as music festivals or concerts, also draw large vehicular crowds. Another case for vehicular crowds is an emergency evacuation due to natural disasters [1].

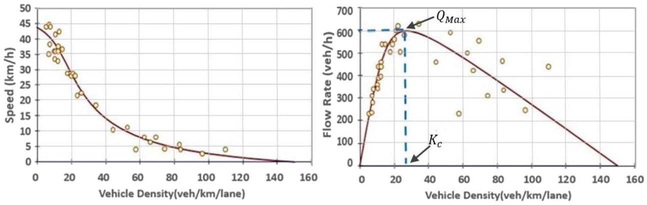


Fig. 1. Fundamental relationships between vehicle density, speed, and flow rates.

The high vehicular traffic volume in such cases exceeds the road network capacity. Cities are not designed to support such high traffic volumes for many reasons. First of all, these events are non-recurrent or occasionally occur infrequently, e.g. every couple of months or years. Secondly, the high cost of increasing the capacity of the current road networks [2]. This cost is not only financial but also administrative because there are other types of networks (i.e., water networks) intersecting with the road network; consequently, many other authorities would have to be involved. Thirdly, in some cases, it is not possible to increase the road capacity because of the area limitation or infrastructure limitations (e.g., bridges). Lastly, such events need special traffic control techniques [3], which may include customized traffic signal timing combined with blocking some roads and detouring others.

By coupling this high traffic volume and the road network capacity constraints, one can soon realize how vehicular crowds can adversely affect mobility in such cases. From a driver's perspective, the high traffic volume increases the vehicle density on the roads and results in lower average travelling speeds and longer travel times, according to the fundamental relationship between speed and density [4] shown in Fig. 1. From the road link level perspective, increasing the vehicle density above a predefined threshold, known as density-at-capacity, denoted as K_c in Fig. 1, moves the network towards the congested regime, resulting in lower flow rates and lower vehicular speeds. Thus, the best link utilization is achieved at a vehicle density of K_c, the jam density, which produces the maximum flow on the link Q_{Max} shown in Fig. 1. Q_{Max} is known as the link capacity, which is the maximum number of vehicles that can exit a road segment per unit time [5]. In Fig. 1 also, we can see the non-linear relationships between the three parameters, which make traffic assignment a non-trivial problem, especially in large networks with different road segments and different parameters for each segment.

Manuscript received March 6, 2020; revised October 6, 2020; accepted December 7, 2020. Date of publication December 31, 2020; date of current version February 12, 2021. This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant RGPIN-2019-05667. The review of this article was coordinated by Prof. Jian Weng. *(Corresponding author: Ahmed Elbery.)*

Ahmed Elbery and Hossam S. Hassanein are with the School of Computing, Queen's University, Ontario, Kingston K7L 3N6, Canada (e-mail: aelbery@vt.edu; hossam@cs.queensu.ca).

Nizar Zorba is with the Electrical Engineering Department, Qatar University, Doha 2713, Qatar (e-mail: nizarz@qu.edu.qa).

Hesham A. Rakha is with the Civil and Environmental Engineering Department, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061-0131, USA (e-mail: hrakha@vt.edu).

Digital Object Identifier 10.1109/TVT.2020.3048336

It is obvious that higher traffic demand increases vehicle queuing at intersections, which increases travel time. Moreover, in some cases, vehicles may stop at traffic signals for more than one cycle, and in complex road networks, this may result in network grid-locks.

Following large events, the high traffic demand and its impact on the mobility in such constrained road networks bring forward the need for efficient and smart traffic control and route planning techniques. Such techniques should efficiently utilize the network facilities while maintaining a certain performance level that satisfies the drivers' expectations.

The widespread proliferation of the Internet of Things (IoT) [6], including transportation applications [7], [8], opens new doors to boost mobility and reduce congestion problems in smart cities. The ubiquitous use of smartphones and the foreseeable prevalence of connected vehicles pave the way to building new IoT paradigms that exploit crowdsensing to tackle city-wide complex problems.

In this paper, we utilize crowdsensing and IoT to introduce a novel vehicular crowd management framework, the IoT-VCM, that can better manage vehicle crowds after special events or in emergency evacuation cases. The proposed framework takes advantage of crowdsensing to collect data from drivers through smartphones or connected vehicles. This information is used to compute the network state conditions, that are used to optimize the route assignment and manage vehicle departures.

The objectives of the proposed system are to clear the network faster and, in the meantime, maintaining low average travel times, by limiting the vehicle density on each road segment to its density-at-capacity K_c. Achieving these objectives is challenging because of the trade-offs between them. In order to clear the network faster, there are two requirements. Firstly, vehicles are required to depart as early as possible. Secondly, vehicles should travel at high speed to minimize travel time. However, these two requirements conflict with each other because allowing high vehicle departure rates results in high traffic volumes, which increases the congestion level, consequently, increases the travel time. On the other hand, by reducing the vehicle departure rate, vehicles will travel at higher speed and experience shorter travel times, but some vehicles will start their trips very late, which will again increase the network clearance time.

To address this problem, the proposed system offers a compromise between these two objectives (the maximum allowable departure rates and lower travel times) by integrating two subsystems:

1) *The Vehicle Departure Control (VDC)* which is responsible for controlling vehicle departures to allow vehicles to leave as early as possible while maintaining a maximum congestion level. It achieves this target by computing the maximum allowable vehicle departure rates based on both the current states of road links and their capacities.
2) *The Travel-Time System Optimum Navigation (TTSON)* which minimizes the network-wide travel time. TTSON stochastically assigns routes to vehicles in such a way that utilizes the available alternative routes, and at the same time, considers their available capacities in order to avoid congesting these roads.

The IoT-VCM uses the collected data to compute the road state conditions (traffic volume and travel time on each road segment) in real-time. The road state condition information is used along with historical data by the TTSON to optimize the network-wide vehicle route assignment using Linear Programming (LP) [9]. In the formulated LP optimization problem, the link capacities and their current traffic loads are used to constrain the network congestion.

The same optimization model is used by the VDC to compute the maximum allowable vehicle departure rates after the event, hence, the vehicle inter-departure intervals. Periodically, the VDC notifies the drivers about their estimated departure times. In the case of high traffic demand, some drivers may be requested to postpone their departure. As a motivation, the system can offer them direct incentives such as discounted tickets for other events, or spending some time to take pictures and receive signatures from public figures. A more advanced incentive mechanism can be utilized such as the social-aware incentive mechanism based on deep reinforcement learning proposed in [10].

The proposed system is developed within a microscopic simulator and tested on a real road network with real calibrated traffic. The 2022 FIFA World Cup, which will be held in Doha, Qatar, is used as the case study. The Doha road network is implemented and used to compare our proposed system to the dynamic time-dependent incremental user-optimum traffic assignment as a typical real-time navigation system that is currently in use (e.g., Waze and Google Maps).

The remainder of the paper is organized as follows. Section II explores the related work. Section III explains the network model. Section IV describes the IoT-VCM system and its components. Section V tackles our case study on the Doha network in Qatar and the results. The final conclusions are presented in Section VI.

II. Related Works

The advances of vehicular communication and IoT have promoted smart mobility and navigation systems that use vehicles and mobile devices as sensors to collect real-time information. Online services, such as Google Maps and Waze, provide online dynamic navigation guidance based on the estimated travel time information collected from vehicles or mobile devices. Other online services, such as INRIX [11], provide real-time traffic information to assist drivers and autonomous vehicles selecting routes. Some research efforts [12], [13] propose a mobile crowdsensing that uses both current and historical information to predict traffic conditions and travelling speeds to enable the dynamic routing of drivers wishing to avoid congestion. All these guidance systems typically do an all-or-nothing traffic routing by assigning all vehicles to the shortest path (which are user equilibrium models) or, at most, provide alternative routes without considering the system-wide performance or trying to minimize network-wide congestion.

Other research efforts consider balancing the traffic across alternative routes. For instance, [14] proposes a heuristic approach to randomly assign vehicles to different routes based on each vehicle's remaining travel time and the route popularity. Compared to this algorithm, our proposed model is a step up because we utilize optimization to assign routes instead of assigning routes randomly. Moreover, our proposed model accounts for the network-wide state conditions in assigning the traffic (i.e., the current traffic load on each road segment).

An evacuation route planning model presented in [15] computes the relationship between clearance time, number of evacuation paths, and congestion probability during an evacuation. The model in [15] does not utilize vehicular networks and does not account for the current network conditions. Instead, it focuses on road capacity uncertainty. In our work, we collect actual road information, from which we can estimate the available road capacities. Consequently, a portion of the traffic can be assigned to underutilized routes. In [16], the authors develop a system-optimum eco-routing model to minimize the total system-wide fuel consumption. By utilizing multiple routes, they are able to reduce traffic congestion and fuel consumption. Compared to [16], our proposed model focuses on the objective of clearing the network faster by enabling vehicles to use alternative routes and by adjusting the vehicle departure times based on the network conditions. The authors in [17] utilize crowdsensing to propose a personalized route planning mechanism based on the quality of the road surface. However, they use shortest path techniques and do not consider the system-wide performance. The authors in [18] propose a grid road network model for path planning in emergency situations. They use a shortest path technique which is applied to the grid road network model they develop.

In literature, crowd management and crowdsensing have been used in different ways to serve different purposes. For instance, in [19], the authors propose a framework to resolve congestion by controlling the movement of crowds of persons. In [20], [21], the authors study the task assignment cost and performance in mobile crowdsensing, and based on that, they combined the path planning and the task assignment using different techniques to improved task assignment performance.

To the best of our knowledge, none of those previous works on crowd management use vehicle departure control as being proposed in this paper.

III. Network Model and Problem Definition

In the proposed system, the road network is represented by a directed graph $\mathcal{G}(\mathcal{N}, \mathcal{L}, \mathcal{Q})$, as shown in Fig. 2. In this graph, $\mathcal{N} = \{i : i = 1, 2, \ldots, n\}$ is a set of n nodes (e.g., road intersections), and $\mathcal{L}$ is a set of l directed links, each link is a 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 $Q_{i,j}$, which is the maximum traffic flow rate that can exit (or enter) this link [22], [23] (corresponding to Q_{Max} in Fig. 1). Moreover, each segment has a variable travel time $T_{i,j}$ that depends on the traffic conditions on the segment. Additionally, each road link has a time-varying traffic load $\zeta_{i,j}$ that represents the average vehicle rate passing it. Thus, the available capacity $\hat{Q}_{i,j}$ on the road segment $L_{i,j}$ is the difference between its capacity and its current load, i.e., $\hat{Q}_{i,j} = Q_{i,j} - \zeta_{i,j}$.

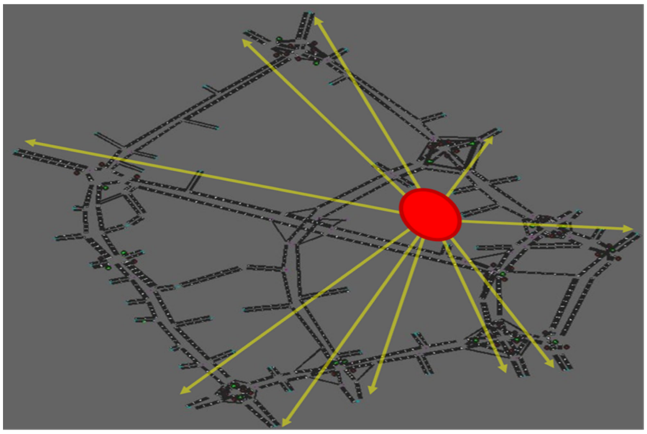


Fig. 2. Network Model.

The vehicular crowd in the network is represented by the red area in Fig. 2. In our case study, this red area is a stadium where some soccer matches will be held. The destinations of vehicles are distributed across the network. Consequently, vehicles in this crowd are grouped into a set of traffic flows, known as Origin-Destination (OD) demand pairs [24]. These traffic flows are built based on the driver destination information collected from the smartphones as described in Section IV-B. Each flow is a set of vehicles that will leave the crowded area after the event to a given destination.

There is a set $\mathcal{F}$ of f concurrent flows, each flow is identified by $k \in \{1, 2, \ldots, f\}$, and each has V_k cars that need to leave within a time interval τ. Thus, each flow $k \in \mathcal{F}$ has a traffic rate q_k (in vehicles per hour veh/h), where $q_k = V_k/\tau$. The initial value for the time interval τ is calculated based on the information collected from the drivers. We assume these vehicles are connected and follow the route recommendations they receive from the Traffic Management Center (TMC). We call these flows the Controlled Traffic (CT).

In addition to the CT, there is Background Traffic (BT), which is the regular day-to-day traffic traversing the network. Vehicles in the BT use the dynamic time-dependent incremental user-optimum (Nash optimum) traffic assignment.

Using crowdsensing, data processing, and optimization techniques, we aim to clear the network in a shorter time using two techniques. The first is the stochastic system optimum routing, which is the core function of the TTSON subsystem. Its goal is to efficiently utilize the network resources by simultaneously routing CT vehicles (going to the same destination) through multiple alternative routes, in such a way that minimizes the network-wide travel time (system-optimum traffic assignment). The second technique is the departure control that computes the CT vehicle maximum allowable departure rates and controls vehicle departure times based on these rates to avoid network congestion. These two techniques are jointly optimized to allow vehicles to depart as early as possible while avoiding network congestion. This coupling results in reasonably high

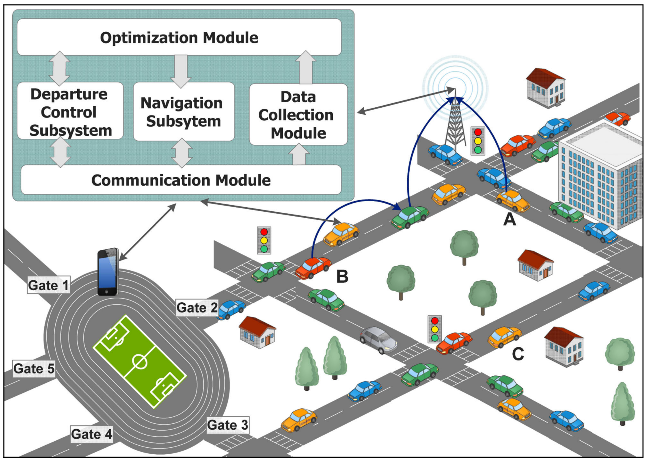


Fig. 3. System Components.

average speeds. Consequently, it clears the network in a shorter time.

IV. IoT-Based Vehicle Crowd Management Framework

The IoT enables the integration of several technologies and communications paradigms such as identification and tracking technologies, wired and wireless sensor and actuator networks, communication protocols, and smart devices [25]. Our proposed system utilizes IoT by integrating different sensors (connected vehicles, smartphones) and actuators (autonomous vehicles and human drivers) to enable better crowd management techniques for vehicular crowds. We will refer to “connected vehicle” to both smart vehicles or non-smart vehicles with a passenger holding a connected smartphone.

This section describes the proposed framework and its components, including the data collection and processing, the optimization process, the stochastic navigation subsystem, and the departure control subsystem.

The front-end of the system is an application that can be installed on vehicle on-board computers or smartphones. Using the Global Positioning System (GPS) capabilities in these devices, this application computes the travel time that the vehicle experiences on each road segment. Then, it communicates this information to the TMC. Additionally, it can request routes from the TMC. The mobile application also has an interface to the VDC. Through this interface, the user can set the destination and receive the departure control information as will be shown in Section IV-E.

The back-end system is the TMC, which has the following modules; data collection and processing module, optimization module, VDC, and TTSON. The system architecture is shown in Fig 3.

A. Communication Module

As shown in Fig. 3, the communication module is responsible for communicating the different system modules to external components such as vehicles or smartphones. When the communication module receives a message from a connected vehicle, it performs the required processing such as authentication, decryption, and unpacking. Then, based on the message contents, it forwards it to the appropriate module. For instance, route request messages are forwarded to the navigation module. Through the communication module, all the components can also send messages to the users through the communication network which is the infrastructure that communicates the system back end to different types of sensors/actuators. The communication depends on the types and capabilities of the network and the different devices. For instance, smartphones can use cellular or WiFi networks. Vehicles can use their VANET [26] or the 5G [27] communication capabilities using either vehicle-to-infrastructure (V2I) or Vehicle-to-Vehicle (V2V) multi-hop connections [28].

B. Data Collection and Processing

To optimize navigation and vehicle departures, the proposed system needs to compute two parameters for each road segment $L_{i,j}$: its travel time $T_{i,j}$ and its current traffic load $\zeta_{i,j}$. Moreover, it needs to estimate the traffic flows and their rates. The system employs crowdsensing to collect data from different sources to infer these network state conditions.

1) Travel Destinations and Flows: The destination information from drivers is necessary for estimating the traffic flows and their distribution across the network. During or before the event, the system polls the destination from divers. The driver destinations are used to build the traffic flows, where the system divides the city into a set of destination areas. All the drivers going to the same destination area are aggregated in the same traffic flow. The number of vehicles in each flow is used to estimate the flow’s initial average traffic rates based on exponential distribution headway intervals. These rates will be used as initial rates by the optimization module and can be updated by the VDC based on the network conditions.

2) Traffic Information From Vehicles: Data collected from connected vehicles are used, along with other data sources (such as road network information, i.e., free-flow speeds, capacities, and lengths) to continuously update the network state conditions, including link travel times $T_{i,j}$ and traffic loads $\zeta_{i,j}$.

In order to compute these parameters, we utilize the GPS in connected vehicles or smartphones. Through the GPS, the front-end application can identify which link it is traversing. Whenever a vehicle enters a new road segment, the application initializes the travel time to the current time, and when the vehicle exits the road segment, it computes the link travel time $T_{i,j}$. Subsequently, the application builds a message containing this information along with the current time. The application then tries to send this message to the TMC using either V2I (such as car A in Fig. 3) or V2V (such as car B in Fig. 3). If the car does not have a connection (such as car C in Fig. 3), it will store the message until reaching an area covered by the network, or finding a vehicle-to-vehicle path to the TMC.

The TMC drops the expired messages based on the message timestamp. Upon receiving an unexpired message, it uses the received travel time to update the link information. The received link travel times may be noisy due to different driver behaviours and random interactions between vehicles. To reduce this noise, the system smooths the received link travel time with the link's previously smoothed value using exponential smoothing (in our case, we used an exponential damping factor $\alpha = 0.2$), as shown in (1).

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

3) Estimating the Current Link Loads: The second important parameter to compute the current traffic load $\zeta_{i,j}$ on each link, which is the difference between the link capacity and its current traffic load. The average traffic load on a link can be estimated using the time-mean speed on the link, as shown by the fundamental relationship shown in Fig. 1. Assuming that connected vehicles on the network is a representative sample for vehicles, their time-mean speed can be used as an estimator for the total link time-mean speed. In literature there are models to represent these fundamental relationships such as the Greenshields [29], Pipes [30], Gipps [31], and Van Aerde [32] models.

In our system, we use the Van Aerde model since it combines both the Greenshields and Pipes models. So, the data processing module uses the smoothed $T^{t}_{i,j}$ and the link length to compute the average link speed $u^{t}_{i,j}$. Then, it plugs this value in Eq. (2) to compute the average heady distance which is the inverse of the vehicle density.

$$h_{i,j} = c_1 + c_3\, u^{t}_{i,j} + \frac{c_2}{u_f - u^{t}_{i,j}} \tag{2}$$

In 2, c_1, c_2, and c_3 are computed based on the road network parameters, as:

$$c_1 = \frac{u_f}{K_c\ u_c^2}(2u_c - u_f) \tag{3}$$

$$c_2 = \frac{u_f}{K_c\ u_c^2}(u_f - u_c)^2 \tag{4}$$

$$c_3 = \frac{1}{Q_{i,j}} - \frac{u_f}{K_c\ u_c^2,} \tag{5}$$

where u_f is the free-flow speed of the link; u_c is its speed-at-capacity; and K_c is the roadway jam density [32]. The headway distance computed in 2 is the reciprocal of the vehicle average density. The traffic load on the link is computed as the multiplication of the link speed by the vehicle density [33], i.e., $\zeta_{i,j} = u_{i,j}/h_{i,j}$.

C. Optimization Module

The optimization module is the core of the proposed system that steers both the VDC and TTSON. Based on the information collected or computed in the previous subsections, the optimization module uses the linear problem described in our previous work [34] to jointly compute allowable vehicle departure rates and optimize their routing.

An Example for Navigation Optimization: We show the idea of optimizing traffic assignment by giving the example in Fig. 4, where a simple network of highway stretch (HW) and an alternative longer travel time arterial road (AR). At point **'A'**, the capacity of the HW is reduced from three to two lanes. Since the HW has a shorter travel time, all vehicles will take it. In the case of high steady-state traffic rate, congestion will take place at point **'A'** and will spill back, creating a queue on the HW, while leaving the AR underutilized. Using the shortest path routing, this congestion will continue and the travel time of the HW will increase until it becomes longer than that of the AR. At this point, vehicles will start switching to the AR, causing congestion on it. This congestion will switch between the HW and the AR.

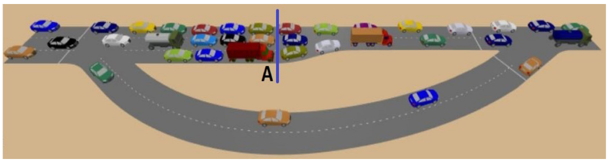


Fig. 4. Illustrative Simple Scenario Example.

In our proposed system, the TTSON will divide the traffic flow between the HW and the AR in such a way that minimizes the network-wide travel time. For instance, if the traffic flow rate is $\mu\ veh/h$, then $\beta.\mu\ veh/h$ will take the HW, and $\gamma.\mu\ veh/h$ will take the AR, where β and $\gamma \in [0,1]$, and in this example, $\beta + \gamma = 1$. We call β and γ the link-flow assignment parameters that must be computed to minimize the network-wide travel time while maintaining the total flow on each road within capacity.

1) Optimization Problem and Constraints: As illustrated by the simple scenario, the optimization module attempts to reduce the network-wide travel time and congestion by utilizing all the available road capacities. To fulfill this objective, a linear optimization model in 6 is formulated. The general idea of the linear program is to divide each flow across a set of alternative routes by computing $q_k^{i,j}$ that minimizes the network-wide travel time while respecting the road capacities. $q_k^{i,j}$ is the portion of this traffic flow k that should go through $L_{i,j}$.

$$\begin{aligned}
&\min_{q_k^{i,j}} \sum_{i=1}^{n}\sum_{j=1}^{n} T_{i,j} \sum_{k=1}^{f} q_k^{i,j}\\
&\text{subject to :}\\
&\sum_{d=1}^{n} q_k^{i,d} - \sum_{s=1}^{n} q_k^{s,i} = 0,\ \text{if } i \text{ is an intermediate node}\\
&\sum_{d=1}^{n} q_k^{i,d} - q_k = 0,\ \text{if } i \text{ is the source of the } k^{th} \text{ flow}\\
&q_k - \sum_{d=1}^{n} q_k^{s,i} = 0,\ \text{if } i \text{ is the destination of the } k^{th} \text{ flow}\\
&\zeta_{i,j} + \sum_{k=1}^{f} q_k^{i,j} \leqslant Q_{i,j} \quad \forall\ L_{i,j} \in \mathcal{L},\\
&q_k^{i,j} \geq 0.
\end{aligned} \tag{6}$$

In this model, $i, s, d \in \mathcal{N}$ such that $L_{i,d}$ is a link exiting node i, and $L_{s,i}$ is a link entering node i. $T_{i,j}$ is the travel time for link $L_{i,j}$. $Q_{i,j}$ is the link capacity, and $\zeta_{i,j}$ is the total traffic load on that link.

The linear program constraints are divided into three sets: route continuity, link capacity, and positive assignment constraints.

The Route Continuity Constraints: This set of constraints is represented by the first three constraints in 6. The objective of these constraints is to ensure that the computed traffic-flow assignment creates a complete and connected route for each flow k from its origin 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 flow k, the summation of the traffic of the k^{th} flow entering this i^{th} node must be equal to the summation that exiting this node.
- For each source (or destination) node, we add (or subtract) the total flow rate q_k. For instance, for the source node i that generates the k^{th} flow whose rate is q_k, we assume there is a fictitious source sending q_k to it, then node i sends these vehicles to other nodes $d \in \mathcal{N}$. And vice versa for the destination nodes.

The Link Capacity Constraints: Constraints in this set are used to control the load on the roads by limiting the total traffic traversing a link $L_{i,j} \in \mathcal{L}$ to its capacity $Q_{i,j}$. The total traffic rate on the link is calculated as the summation of its current load $\zeta_{i,j}$ and all the traffic assigned to this link $\sum_{k=1}^{f} q_k^{i,j}$.

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.

Link costs $T_{i,j}$ and traffic load $\zeta_{i,j}$ are continuously updated using the methodology described earlier. The system periodically solves this problem to cope with the dynamic network condition. This period is called Re-Optimization Interval (ROI).

D. Navigation Module

When a vehicle requests a route (or a route update), the communication module forwards this request to the TTSON. To build a route for a vehicle, TTSON, finds the parameters needed to build the route, namely; the vehicle's destination node d, the vehicle's flow number k (inferred from its destination), its current location from which it can find the route starting node s, and the link-flow assignment $q_k^{i,j}$ for the k^{th} flow. Then, it calls the route construction procedure, shown as Algorithm 1, which takes four input parameters and stochastically creates a route as follows.

The algorithm 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 $\check{L}$. For each link $L_{i,j} \in \check{L}$, it computes p_j, which is the probability of selecting this link as the next link in the route, as

$$p_j = \frac{q_k^{i,j}}{\sum_{L_{i,m} \in \check{L}} q_{i,m}} \tag{7}$$

Then, based on the computed values p_j, the algorithm stochastically selects one of these links to be the next link and adds it to the route. Additionally, its end node j is used as the current node i, and the process is repeated until reaching the vehicle's destination d. Using the individual flow balance as constraints in the optimization problem guarantees that the route built by this algorithm will reach the vehicle's destination, i.e., ensures the route connectivity.

Algorithm 1

```
procedure Build a vehicle route$k$, $q_k^{i,j}$, $s$, $d$
  $R \leftarrow \phi$
  $i \leftarrow s$
  while $i \neq d$ do
    $R \leftarrow (R, i)$
    $\check{L} \leftarrow L_{i,j} : L_{i,j} \in \mathcal{L}$
    for each $L_{i,j} \in \check{L}$ do
      $compute\ p_j$
    $r \leftarrow randomnumber$
    for $j = 1$ to size of $\check{L}$ do
      if $\sum_{\check{j}=1}^{j} p_{\check{j}} \leqslant r$then
        $i \leftarrow j$
        $break$
  return $R$                    △ Return the computed route.
```

E. Departure Control Module

The VDC module is responsible for regulating the traffic entering the network to avoid network congestion. Moreover, in the case of high traffic demand (i.e., when the traffic demand rates exceed the available network maximum flow rate), the VDC computes the maximum allowable traffic rates $\hat{q_k} \leq q_k$, and, accordingly, adjust the vehicle departure times.

1) Computing the Maximum Allowable Traffic Rates: There are existing algorithms to compute the multi-origin multi-destination maximum flow, such as [35], [36]. But these solutions have two main drawbacks. First, the solution they achieve may result in a complete blockage of some traffic flows (i.e., setting $\hat{q_k} = 0$ for some flows), which violates the fairness among the traffic demands. Additionally, this flow blockage will result in differing vehicles in these flows for a long time, consequently, increasing the network clearance time. Secondly, in multi-origin multi-destination maximum flow algorithms, the computed traffic rates do not necessarily minimize the travel time since the maximum flow algorithms do not consider other metrics other than road capacities.

To maintain fairness and minimize travel time, the VDC determines the network maximum flow rates by reducing all the traffic demand rates by the same ratio, and then find if there is a feasible solution for the linear problem in 6 using these rates. It relies on optimization failure as an indication of high traffic demand. To find the maximum allowable departure rates, the VDC applies the algorithm shown in the flowchart in Fig. 5. The algorithm is similar to the binary search in a sorted list. R_{Max} and R_{Min} are the maximum and minimum boundaries of the search space that are updated each iteration. The R_{Max} is initialized to the initial rate computed based on the drivers leaving times. R_{Cur} is the current traffic rates, it is initialized

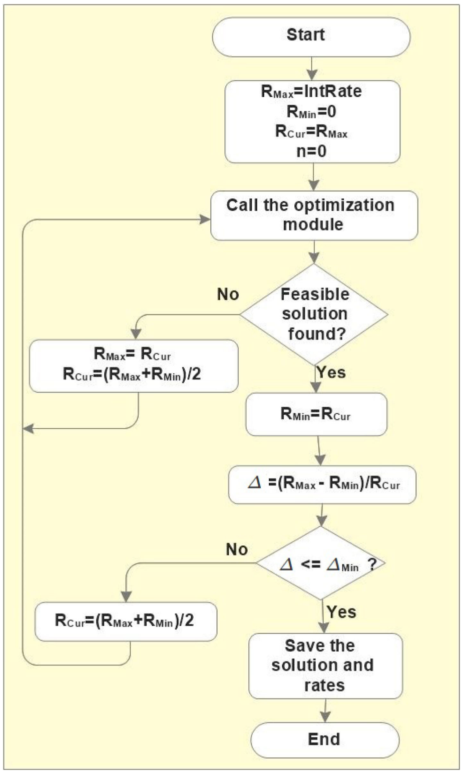


Fig. 5. Computing the minimum cost maximum flow rate procedure.

to R_{Max}. Because of the continuous horizon of the rates, we define a terminating condition, which is the minimum difference ratio Δ_{min} between the size of the search space and the current solution.

The algorithm works as follows. The VDC starts by $R_{Cur} = R_{\text{Max}}$ and calls the optimization module to find a solution for these rates based on the current network state. If a solution is not found, the VDC will receive an optimization failure signal from the optimization module. Consequently, it decreases the R_{Cur} to the middle point between R_{Max} and R_{Min} and recalls the optimization. A successful optimization can occur if the traffic rates are lower than the maximum available capacity. Therefore, when the VDC receives a successful optimization signal, it firsts uses the R_{Cur} as minimum rates $R_{Cur} = R_{\text{Min}}$ and then increases the current R_{Cur} to $(R_{\text{Min}} + R_{\text{Max}})/2$. This process is repeated until one of the terminating conditions is achieved. The returned solution satisfies the two objectives; fairness and minimum travel time. It is also less than or equal to the available network capacity.

An important advantage of this technique is that it is consolidated with the navigation system by using the same optimization problem, once we find a solution, its arguments are directly used by the navigation module to minimize the system-wide travel time. This method achieves $O(log(\frac{R_{\text{Max}}}{\Delta_{\text{Min}}}))$ complexity, i.e., the maximum number of calls to the optimization module is $\leq log(\frac{R_{\text{Max}}}{\Delta_{\text{Min}}})$.

2) Controlling Vehicle Departures: After computing the maximum allowable rate $\hat{q_k}$ for each traffic demand, the VDC uses 8 to adjust the vehicle departure times to match the new rates.

$$\bar{t_v} = T_0 + (t_v - T_0)\frac{q_k}{\hat{q_k}} \tag{8}$$

where, t_v and $\bar{t_v}$ are the previous and the new departure times for vehicle v, T_0 is the current time. This equation shrinks or extends the vehicle inter-departure intervals by the ratio $q_k/\hat{q_k}$.

The departure time is sent to drivers enough time before his/her departure. The VDC uses a token-based model to control the departure of vehicles. So, if the departure time for vehicle v is computed to be t_v, the VDC sends it a token at time $t_v - T_p$, where T_p is the preparation time, i.e., the time needed for the driver to go to the car, to leave the parking lot and go to the gate at designated exit points (i.e., G1, G2,.. G5 in Fig. 3). Once the VDC has sent a token to a driver, the VDC cannot change their departure time.

V. Simulation and Results

To accurately test the proposed model, we developed it within the INTEGRATION software [37], which is an agent-based microscopic traffic simulation and assignment framework. It is characterized by its accuracy in computing travel time and capturing network congestion and its impact on travel time. INTEGRATION can accurately compute travel time because it replicates vehicle longitudinal motion based on the Van Aerde model [37], which produces the best regression for real datasets compared to other models [38]. In INTEGRATION, vehicle speed and acceleration are constrained by the vehicle dynamics [37]. Moreover, its microscopic nature enables it to accurately model vehicle queues and shock-wave at traffic lights [39], which significantly affects the link travel time. It also accounts for traffic lights and their impact on travel time.

We use the CPLEX Optimizer [40] to compute the traffic assignment. To study the efficiency of the proposed framework, we compare it to the Sub-population Feedback Dynamic Time-Based Traffic Incremental Assignment (SFDTIA) [37], which uses the shortest path routing based on the dynamic link travel times since this would best reflect the state-of-practice routing. SFDTIA also tries to overcome the shortest path routing problem and utilize the network resources by dividing the traffic into five sub-populations; each sub-population is routed differently at the same time. This way, it can utilize alternative routes [37].

We use the 2022 FIFA World Cup event, which will be held in Doha, Qatar as a case study. After each match, the number of vehicles that need to leave the stadium will be huge, 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 create a real-world scenario of the 2022 World Cup case study, the Doha road network, shown in Fig. 2, is developed and used for the simulation.

To build the simulation network and to generate the associated parameters (such as the road speed, number of lanes on the road,

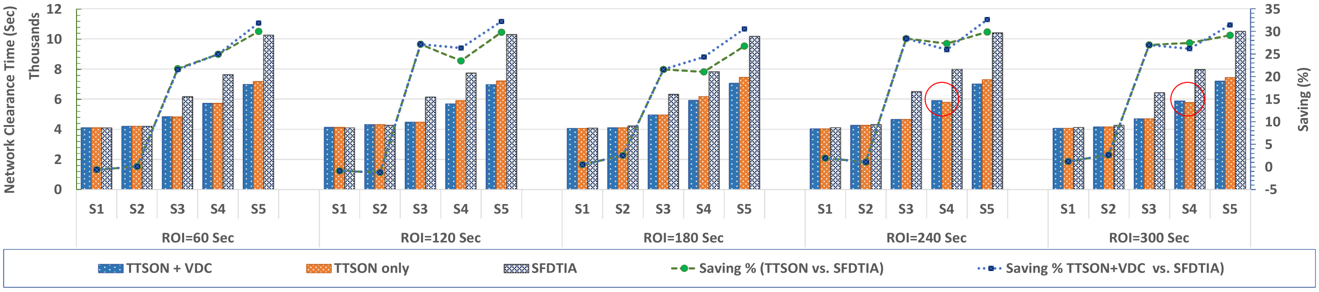


Fig. 6. Network clearance time using TTSON and VDC for all scenarios and different re-optimization intervals.

and traffic lights), we used data from different sources, namely; 1) a road network Geographic Information System (GIS) Shapefile; 2) intersection data from OpenStreetMap (OSM); and 3) Google maps and ArcGIS. The four key traffic stream parameters: free-flow speed, speed-at-capacity, saturation flow rate, and jam density were also derived from different sources. Specifically, the free-flow speed was derived from the roadway speed limits. The speed-at-capacity was set at 80% the free-flow speed as has been demonstrated in the literature [41], [42], the saturation flow rate was derived using the Highway Capacity Manual (HCM) based on the roadway free-flow speed. Finally, the jam density was computed using a typical vehicle length. The Doha city shapefile was used to generate the network nodes and links. OpenStreetMap data were used to extract intersection traffic control information including the traffic control methods (stop sign, yield sign, or traffic signals). The number of phases for each traffic signal and traffic signal timing data was obtained based on field observation and was augmented with real-time traffic signal optimization. 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 signals.

The red area in Fig. 2 represents 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 ten network exit points.

In regards to background traffic (BT), which represents the regular traffic traversing the network, it was calibrated based on car counts data collected from the OpenStreetMaps (OSM) website. The BT rate in the first row ($S1$) in Table II is 10% of the calibrated traffic. In this scenario, $S1$, we assume that there are 2800 cars as the CT. The rates in $S1$ are multiplied by scaling factors 2 through 5 to compute the higher traffic rates in $S2$ through $S5$. 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.

B. Simulation Results

Each of the five traffic levels cases was run using both three methods; the Base case (SFDTIA), the TTSON, and TTSON+VDC. For statistically significant results, each scenario was run 16 times with different seeds and the average parameters computed.

TABLE I
TABLE OF NOTATIONS

Symbol	Description
i, j	The network node number
n	The number of nodes in the network
f	The total number of traffic demands
k	The traffic flow number
$L_{i,j}$	The road link from node i to node j
$Q_{i,j}$	The maximum capacity of the road segment $L_{i,j}$
q_k	The rate of traffic flow number k in veh/h
$T_{i,j}$	The smoothed travel time on link $L_{i,j}$
$\zeta_{i,j}$	The smoothed average traffic load on link $L_{i,j}$
α	The exponential damping factor for travel time
u_f	The the free-flow speed of a road link
u_c	The speed-at-capacity of a link
$h_{i,j}$	The average spacing on a link $L_{i,j}$
$u^t_{i,j}$	The current average speed on a link $L_{i,j}$
$q_k^{i,j}$	The traffic assignment parameter, the portion of the traffic flow k that traverses link $L_{i,j}$
$u^t_{i,j}$	The current average speed on link $L_{i,j}$
$u^t_{i,j}$	The probability of selecting link $L_{i,j}$ as the next link along the route
$\hat{q_k}$	The allowable departure rate for the k^{th} traffic flow

TABLE II
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

1) Network Clearance Time: Fig. 6 compares the network clearance time for the TTSON, TTSON with VDC, and the SFDTIA for the five traffic levels.

It shows that the TTSON can clear the network faster in high traffic demand levels, which are typical cases of its use. In the case of low traffic rates, the differences in the clearance time are not significant. It also illustrates that the higher the traffic rate the better the saving it achieves.

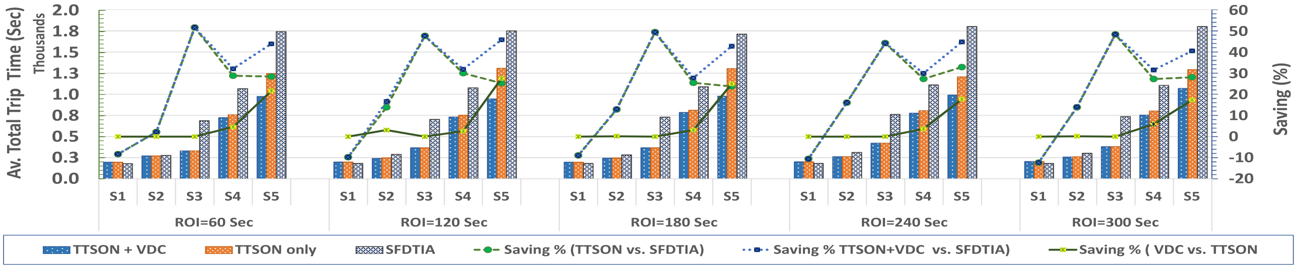
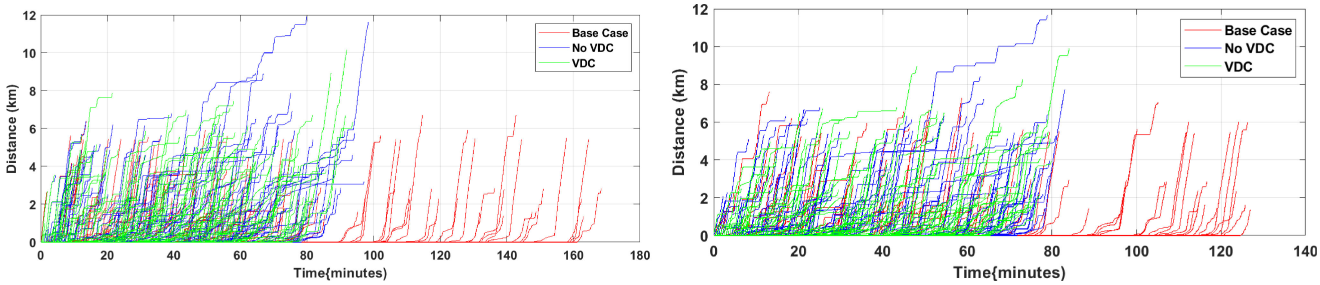


Fig. 7. Average total trip time using TTSON and VDC for all scenarios and different re-optimization intervals.



(a)

(b)

Fig. 8. Time-Space diagram for 0.5% of the cars. (a) S5. (b) S4.

By applying the VDC with the TTSON, Fig. 6 also demonstrates an improvement in the network clearance time in most cases, while in some cases, the VDC slightly increases the network clearance time. The reason is that the final impact of the VDC has two opposing components. The first is that by reducing the traffic load, it reduces the congestion and decrease vehicle travel times, which allows the network to clear early. On the other hand, by decreasing the traffic load, vehicles start their trips late, which may increase the clearance time.

The figure also shows that, at the low traffic levels (S1, S2, and S3), the VDC does not have any impact. The reason is that, at these low rates, the network capacity is sufficient for the traffic. Thus it does not need to decrease the traffic rates.

An interesting observation in Fig 6 is that, in some cases, longer ROIs result in shorter network clearance time. For instance, in the case of S3, increasing the ROI from 60 sec to 120 sec results in 3% reduction in the network clearance time. The reason is that while lowering the ROI helps the CT to arrive early, it may increase the trip time for the BT (as we will show later in Section V-B4 and Fig. 10). Consequently, in some cases, the delay experienced by the BT may increase the total network clearance time.

2) Average Trip Time: The trip time for each individual vehicle is the time it takes from its scheduled departure time to the moment it reaches its destination. Fig. 7 compares the system-wide average travel time, which is computed as the summation of all vehicles' travel times divided by the number of vehicles.

Fig. 7 demonstrates that at low traffic levels, the proposed system increases the average trip time by around 10%. This increase is attributed to the sensitivity of the system to congestion on road segments, which steers the TTSON to route vehicles through longer routes. Consequently increases the travel time for those vehicles.

However, as the traffic rate increases, the proposed system shows significant improvements in the system-wide average trip time that exceed 50% in extreme cases.

3) Time-Space Diagram: Fig. 8 shows the time-space diagram for 0.5% of the cars for the two high traffic demand levels. Each line in this figure represents a vehicle trajectory (the distance it travelled versus time). It shows that for the base case, many vehicles departed very late. Despite the fact that these vehicles attempted to depart in the first hour, we can see some vehicles departed after 160 minutes, implying that these trips were delayed for more than an hour. The reason is that SFDTIA uses a limited number of alternative shortest routes, thus when a driver tries to start their trip and finds its entry road full, it defers its departure. While in our proposed system, a car can be assigned longer routes in which the entry links are almost underutilized. Thus, not requiring the deference of vehicle departure times. This is clear in the cases of TTSON and TTSON with VDC, where the last car departed within 80 minutes of the simulation time. The second point in this figure is that the car speeds in the case of SFDTIA are higher, which is reflected by the steep trajectory slopes, which is attributed to the low vehicle density in the network.

A time-space diagram also shows that in the case of the TTSON without the VDC, there is severe congestion in the network, which can be depicted from the horizontal blue lines. Such horizontal lines demonstrate that a vehicle is stuck in congestion; consequently, it does not move. Because of this congestion, the TTSON system detours some cars through much longer routes (shown by the long blue lines).

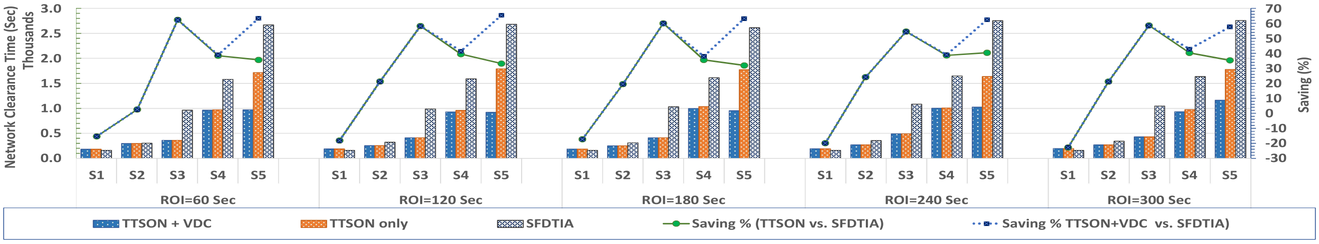


Fig. 9. Average total trip time for the CT.

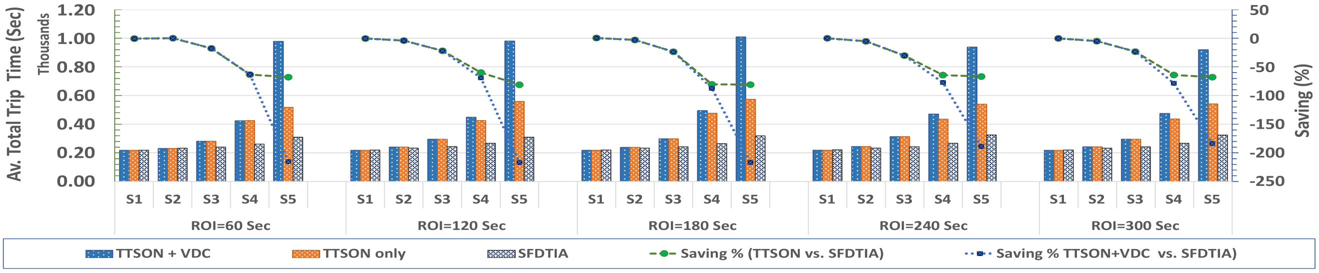
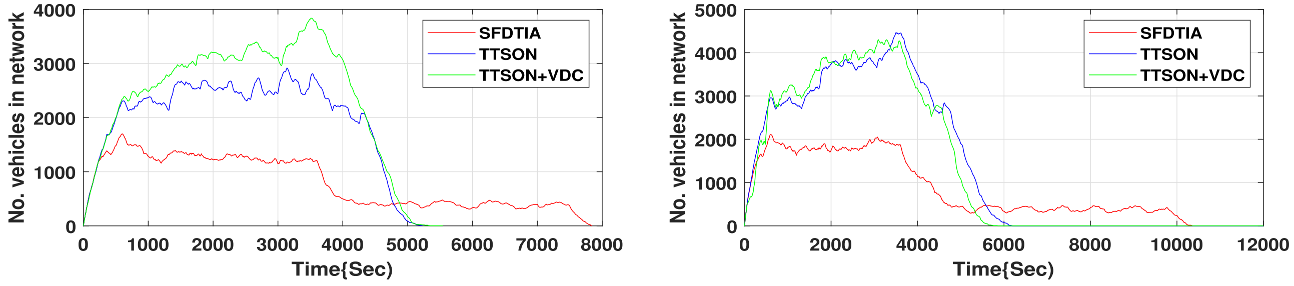


Fig. 10. Average total trip time for the BT.



(a)

(b)

Fig. 11. Number of vehicles in the network versus time. (a) S4. (b) S5.

By employing the VDC with the TTSON, the system is able to reduce the congestion. Therefore, reducing or eliminating the need for longer detours. Consequently, the VDC can reduce the average trip total travel time as shown in Fig. 7.

4) No Free Lunch, CT vs. BT: We also analyzed the traffic class specific performance for the two classes; CT and BT. Fig. 9 and Fig. 10 show the average trip time for CT and BT, respectively. Fig. 9 illustrates the significant impact of the proposed system on the CT average trip time. It shows that the TTSON significantly reduced the CT average trip time. The figure also demonstrates that using the VDC system, contributed to more reduction in the average CT trip time at the high traffic demand levels S4 and S5. It also shows that the impact of the VDC increases with the traffic demand, which is aligned with its purpose of regulating traffic in the congested cases. These improvements are achieved at the cost of the BT, as shown in Fig. 10, where it shows a significant increase in the BT average trip time.

An interesting observation in Fig. 10 is that the VDC results in longer BT average trip time, which contrasts with regulating the vehicle departures. This observation inspired us to investigate this further. By comparing the vehicle density in the network in these different cases, it turned out that regulating the traffic at the gates reduces the congestion at the exit links, consequently allows vehicles to enter the network faster. Thus, resulting in higher vehicle density in the network as shown in Fig. 11 that adversely affects the BT trip time which was doubled in some cases.

This also shows the importance of the TTSON in efficiently controlling the traffic assignment, where the CT can be routed in a better way that decreases its travel time despite the higher vehicle density compared to the SFDTIA.

VI. Conclusion

In this paper, we present a vehicle crowd management system, entitled IoT-VCM, that optimizes the performance of the transportation system in the case of high vehicular demand levels.

To optimize the road network performance and clear the network faster, two conflicting objectives exist, namely; early vehicle departures (which results in higher congestion levels) and short travel times (which requires low traffic congestion). Our proposed system addresses these contradictory objectives by allowing vehicles to depart as early as possible while constraining the traffic load to the maximum network flow rate. Then, to minimize the overall network-wide total travel time, vehicles are routed using a dynamic system-optimum traffic assignment.

The microscopic simulation of the proposed system shows its ability to clear the network in a shorter time while maintaining the travel time as low as possible. The analysis of the vehicle trajectories shows lower congestion when employing the vehicle departure control. Moreover, in some cases, regulating the vehicle traffic entering the network using the vehicle departure control allows vehicles to enter the network faster and increase the vehicle density in the network while still having lower trip times.

References

[1] H. Hassanein, N. Zorba, S. Han, S. S. Kanhere, and M. Shukair, "Crowd management," *IEEE Commun. Mag.*, vol. 57, no. 4, pp. 18–19, Apr. 2019.

[2] X. Zhang, Q. Zhong, and Q. Luo, "Evaluation of transportation network reliability under emergency based on reserve capacity," *J. Adv. Transp.*, vol. 2019, pp. 1–13, 2019.

[3] X. Zhao and Ming-jie Feng and Hui-ying Li and T. Zhang, "Optimization of signal timing at critical intersections for evacuation," *Procedia Eng., Green Intell. Transp. Syst. Saf.*, vol. 137, pp. 334–342, 2016.

[4] N. Geroliminis and C. F. Daganzo, "Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings," *Transp. Res. Part B: Methodol.*, vol. 42, no. 9, pp. 759–770, 2008.

[5] J. G. Wardrop, "Capacity of roads," OR, vol. 5, no. 1, pp. 14–24, 1954.

[6] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[7] Z. Yongjun, Z. Xueli, Z. Shuxian, and G. Shenghui, "Intelligent transportation system based on Internet of Things," in *World Automat. Congr.*, Jun. 2012, pp. 1–3.

[8] S. H. Sutar, R. Koul, and R. Suryavanshi, "Integration of smart phone and IoT for development of smart public transportation system," in *Proc. Int. Conf. Internet Things Appl.*, Jan. 2016, pp. 73–78.

[9] M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali, *Linear Programming and Network Flows*. Hoboken, NJ, USA: Wiley, 2011.

[10] Y. Zhao and C. H. Liu, "Social-aware incentive mechanism for vehicular crowdsensing by deep reinforcement learning," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–12, 2020, doi: 10.1109/TITS.2020.3014263.

[11] INRIX. Accessed: Jan. 2021. [Online]. Avilable: http://INRIX.com

[12] Q. Qin, M. Feng, J. Sun, and B. Sun, "Prediction of road resistance based on historical/real-time information and road quality," in *Proc. 12th Int. Conf. Fuzzy Syst. Knowl. Discov.*, Aug. 2015, pp. 1073–1077.

[13] Y. Regragui and N. Moussa, "Investigating the impact of real-time path planning on reducing vehicles traveling time," in *Proc. Int. Conf. Adv. Commun. Technol. Netw.*, Apr. 2018, pp. 1–6.

[14] J. Pan, M. A. Khan, I. S. Popa, K. Zeitouni, and C. Borcea, "Proactive vehicle re-routing strategies for congestion avoidance," in *Proc. IEEE 8th Int. Conf. Distrib. Comput. Sensor Syst.*, May 2012, pp. 265–272.

[15] G. J. Lim, M. Rungta, and M. R. Baharnemati, "Reliability analysis of evacuation routes under capacity uncertainty of road links," *IIE Trans.*, vol. 47, no. 1, pp. 50–63, 2015.

[16] A. Elbery and H. A. Rakha, "A novel stochastic linear programming feedback eco-routing traffic assignment system," in *Transp. Res. Board 96th Annu. Meeting*, 2017, pp. 1645–1660.

[17] A. Abdelrahman, A. S. El-Wakeel, A. Noureldin, and H. S. Hassanein, "Crowdsensing-based personalized dynamic route planning for smart vehicles," *IEEE Netw.*, vol. 34, no. 3, pp. 216–223, 2020.

[18] B. Yang, Z. Ding, L. Yuan, J. Yan, L. Guo, and Z. Cai, "A novel urban emergency path planning method based on vector grid map," *IEEE Access*, vol. 8, pp. 154 338–154 353, 2020.

[19] Y. Kawamoto, N. Yamada, H. Nishiyama, N. Kato, Y. Shimizu, and Y. Zheng, "A feedback control-based crowd dynamics management in iot system," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1466–1476, Oct. 2017.

[20] W. Gong, B. Zhang, and C. Li, "Task assignment in mobile crowdsensing: Present and future directions," *IEEE Netw.*, vol. 32, no. 4, pp. 100–107, Jul./Aug. 2018.

[21] W. Gong, B. Zhang, and C. Li, "Location-based online task assignment and path planning for mobile crowdsensing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1772–1783, Feb. 2019.

[22] D. M. Levinson and S. Kanchi, "Road capacity and the allocation of time," *J. Transp. Statist.*, vol. 5, no. 1, pp. 25–45, 2002.

[23] R. Margiotta and S. Washburn, "Simplified highway capacity calculation method for the highway performance monitoring system," *U.S. Dep. Transp.: Federal Highway Admin.*, Tech. Rep. PL-18-003, 2017.

[24] M. Abareshi, M. Zaferanieh, and M. R. Safi, "Origin-destination matrix estimation problem in a Markov chain approach," *Netw. Spatial Econ.*, vol. 19, no. 4, pp. 1069–1096, Dec. 2019. [Online]. Available: https://doi.org/10.1007/s11067-019-09447-8

[25] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[26] "IEEE Standard for Information Technology- Local and Metropolitan Area Networks- Specific Requirements- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments," pp. 1–51, Jul. 2010.

[27] G. Naik, B. Choudhury, and J. Park, "IEEE 802.11bd 5G NR V2X: Evolution of radio access technologies for V2X communications," *IEEE Access*, vol. 7, pp. 70169–70184, 2019.

[28] J. Santa, A. F. Gómez-Skarmeta, and M. Sánchez-Artigas, "Architecture and evaluation of a unified V2V and V2I communication system based on cellular networks," *Comput. Commun.*, *Mobility Protocols ITS/VANET*. vol. 31, no. 12, pp. 2850–2861, 2008,

[29] J. Efendi and R. Anwar, "The comparative analysis of the performance of traffic flow using MKJI method, greenshields model, greenberg, and underwood on the way the basis of ULIN banjarbaru km," *CERUCUK*, vol. 3, no. 1, pp. 53–72, 2019.

[30] L. A. Pipes, "Car following models and the fundamental diagram of road traffic," *Transp. Res./U.K./*, 1966.

[31] M. M. Rahman, M. T. Ismail, and M. K. M. Ali, "Simulation of gipps car-following model and validation by China data," in *Proc. AIP Conf.*, 2019, Art. no. 040005.

[32] N. Wu and H. Rakha, "Derivation of van aerde traffic stream model from tandem-queuing theory," *Transp. Res. Rec.*, vol. 2124, no. 1, pp. 18–27, 2009.

[33] L. Immers and S. Logghe, "Traffic flow theory," *Fac. Eng., Dep. Civil Eng., Sec. Traffic Infrastructure.*, vol. 40, p. 21, 2002.

[34] A. Elbery, H. S. Hassanein, N. Zorba, and H. Rakha, "VANET-based smart navigation for vehicle crowds: FIFA world cup 2022 case study," in *Proc. IEEE Glob. Commun. Conf.*, Waikoloa, USA, Dec. 2019, pp. 1–6.

[35] G. Borradaile, P. N. Klein, S. Mozes, Y. Nussbaum, and C. Wulff-Nilsen, "Multiple-source multiple-sink maximum flow in directed planar graphs in near-linear time," *SIAM J. Comput.*, vol. 46, no. 4, pp. 1280–1303, 2017.

[36] G. Borradaile *et al.*, "Multiple-source multiple-sink maximum flow in directed planar graphs in Near-Linear Time," *2011 IEEE 52nd Annual Symposium Foundations Comput. Sci.*, 2010, pp. 170–179, doi: 10.1109/FOCS.2011.73.

[37] H. A. Rakha, K. Ahn, and K. Moran, "INTEGRATION framework for modeling eco-routing strategies: Logic and preliminary results," *Int. J. Transp. Sci. Technol.*, vol. 1, no. 3, pp. 259–274, 2012.

[38] K. Anuar, F. Habtemichael, and M. Cetin, "Estimating traffic flow rate on freeways from probe vehicle data and fundamental diagram," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, 2015, pp. 2921–2926.

[39] M. Di Vaio, G. Fiengo, A. Petrillo, A. Salvi, S. Santini, and M. Tufo, "Cooperative shock waves mitigation in mixed traffic flow environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4339–4353, Dec. 2019.

[40] CPLEX optimizer, IBM. Accessed: Jan. 2021. [Online]. Available: https://www.ibm.com/analytics/cplex-optimizer

[41] H. Rakha and B. Crowther, "Comparison of greenshields, pipes, and van aerde car-following and traffic stream models," *Transp. Res. Rec.*, vol. 1802, no. 1, pp. 248–262, 2002.

[42] H. Rakha and M. Arafeh, "Calibrating steady-state traffic stream and car-following models using loop detector data," *Transp. Sci.*, vol. 44, no. 2, pp. 151–168, May 2010. [Online]. Available: https://doi.org/10.1287/trsc.1090.0297

Ahmed Elbery (Member, IEEE) received the B.Sc. degree (with Hons.) in electrical and computer engineering from the Department of Electrical Engineering, Benah University, Egypt, and the M.Sc. degree in electronic engineering from the Department of Electronic Engineering, Menoufia University, Egypt. He received the Ph.D. degree in computer science form Virginia Tech, USA. He is a Postdoctoral Research Fellow with the School of Computing, Queen's University. His research interests include communication and data networking, wireless and vehicular communication, ITS, simulation and modeling of large-scale vehicular networks in smart cities, eco-routing navigation systems. He was a Senior Network Engineer and has numerous network and communication certificates from different vendors including Cisco, Juniper, and Huawei-3Com. He also has network instructor certificates from Juniper, JNCIA, and JNCIS Junos Enterprise Routing and Switching track. He is a Cisco Academy instructor for CCNA, CCNA Wireless, and CCNA Security.

Hossam S. Hassanein (Fellow, IEEE) is a leading authority in the areas of broadband, wireless and mobile networks architecture, protocols, control and performance evaluation. His record spans more than 500 publications in journals, conferences and book chapters, in addition to numerous keynotes and plenary talks in flagship venues. Dr. Hassanein was the receipient of the several recognition and Best Papers Awards at top international conferences. He is also the Founder and Director of the Telecommunications Research Lab (TRL), Queen's University School of Computing, with extensive international academic and industrial collaborations. He is a Former Chair of the IEEE Communication Society Technical Committee on Ad hoc and Sensor Networks (TC AHSN). Dr. Hassanein is an IEEE Communications Society Distinguished Speaker.

Nizar Zorba (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from JUST University, Jordan, in 2002, and the Ph.D. degree in signal processing for communications from UPC Barcelona, Spain, in 2007. He is a Professor with the Electrical Engineering Department, Qatar University, Doha, Qatar. He has authored five international patents and coauthored more than 120 papers in peer-reviewed journals and international conferences. He is Associate/Guest Editor for the IEEE COMMUNICATIONS LETTERS, IEEE ACCESS, *IEEE Communications Magazine* and IEEE NETWORK. He is Symposium Chair at IEEE Globecom 2021, IEEE VTC 2020 and IEEE ICC 2019; Workshop Chair at ICC 2021, IWCMC 2020 and CSM-PS 2019. He is currently the Vice-Chair of the IEEE ComSoc Communication Systems Integration and Modeling Technical Committee (TC CSIM).

Hesham A. Rakha (Fellow, IEEE) received the Ph.D degree from Queen's University, Kingston, Ontario, in 1993. He is currently the Samuel Reynolds Pritchard Professor of Engineering with the Department of Civil and Environmental Engineering and the Department of Electrical and Computer Engineering (Courtesy), Virginia Tech, and the Director of the Center for Sustainable Mobility with the Virginia Tech Transportation Institute. His research interests include large-scale transportation system optimization, modeling, and assessment. He works on optimizing transportation system operations, including vehicle routing, developing various network and traffic signal control algorithms, developing freeway control strategies (speed harmonization and ramp metering), and optimizing vehicle motion (lateral and longitudinal control of connected automated vehicles (CAVs)) to enhance their efficiency and reduce their energy consumption while ensuring their safety. He has authored or coauthored of six conference Best Paper Awards, namely: 19^{th} ITS World Congress (2012), 20^{th} ITS World Congress (2013), VEHITS (2016), VEHITS (2018), and TRB (2020), received the most cited paper award from the *International Journal of Transportation Science and Technology (IJTST)* in 2018 and received 1st place in the IEEE ITSC 2020 UAS4T Competition. In addition, Dr. Rakha was the recipient of the Virginia Tech's Dean's Award for Outstanding New Professor (2002), the College of Engineering Faculty Fellow Award (2004–2006), and the Dean's Award for Excellence in Research (2007). He is an Editor for *Sensors* (the Intelligent Sensors Section), an Academic Editor for the *Journal of Advanced Transportation*, an Associate Editor for the IEEE TRANSACTIONS OF INTELLIGENT TRANSPORTATION SYSTEMS and the *Journal of Intelligent Transportation Systems: Technology, Planning and Operations*. Furthermore, he is on the Editorial Board of the *Transportation Letters: The International Journal of Transportation Research* and the *International Journal of Transportation Science and Technology*.