A Multiple Linear Regression Model for Predicting Congestion in Heterogeneous Vehicular Networks

Farnoush Falahatraftar, Samuel Pierre, *Senior Member, IEEE*, Steven Chamberland, *Member, IEEE*
Mobile Computing and Networking Research Laboratory (LARIM)
École Polytechnique de Montréal, Montréal, Canada
farnoush.falahatraftar@polymtl.ca, samuel.pierre@polymtl.ca, steven.chamberland@polymtl.ca

***Abstract*—Finite capacity of network resources and enormous data generated by vehicles using safety and comfort applications, have made network congestion a challenge to manage in Heterogeneous Vehicular Network (HetVNET). In this paper, we propose a reliable network congestion model based on a Multiple Linear Regression (MLR), which is a supervised machine learning algorithm to predict network congestion in HetVNET. We have evaluated the performance of our proposed network congestion prediction model using a Cross-Validation test approach. Numerical results show that the proposed linear congestion prediction model is reliable, which can explain and support variability of the response as well. Moreover, we have weighted effectiveness of each considered HetVNET parameters, in association with congestion situation in HetVNET.**

***Index Terms*—Heterogeneous Vehicular Networks (HetVNET), network congestion prediction, supervised machine learning method, Intelligent Transportation System (ITS).**

I. Introduction

In Heterogeneous Vehicular Network (HetVNET) connected vehicle's users can be profited from various services, which are provided by Dedicated Short Range Communication (DSRC) and Long-Term Evolution (LTE) [1]. In vehicular networks, data generated by vehicles can be transmitted via two types of communications: Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I). HetVNET applies DSRC and LTE in V2V and V2I communications respectively [2].

The growing number of smart vehicles and the significant tendency of people to enjoy diverse services from Internet ubiquity generate high resource demands for data transmission through network, which is a notable challenge. In the scenario that a huge volume of data needs to be transmitted, but we do not have enough resources in HetVNET to dedicate, network will experience congestion situations. Unfortunately, congestion can impair network performance and user satisfaction by collapsing the Quality of Services (QoS). When a vehicular user finds out that data transmission is low and has to wait for network respond, then his satisfaction typically will drop. Therefore, congestion in a vehicular network is negatively related to QoS. In a real time situation, when an accident has happened, two actions of warning other vehicles on the way and sending an alarm to immediate health help must be done within an appropriate time and with least latency. In such a scenario, if the HetVNET encounters congestion problem and no congestion mechanisms are used, the direct effect of congestion on QoS could have irreparable harms for human life, time consumption and money expenditures.

Intelligent learning methods help devices and machines to learn from existing data, and then use what they learned for new data, which the device may have never seen those data before. Machine learning algorithms are categorized into supervised and unsupervised learning. An unsupervised learning method is capable to learn and making solutions with no error evaluation. However, a supervised method is an error correction method, and learning will be matured by training [3].

Regarding to this preface, we decided to study current works related to network congestion in vehicular networks, which the authors used machine learning algorithms in them. However, the number of works in this area is limited. That being said, controlling congestion by applying an unsupervised algorithm for clustering information generated by vehicles in Vehicular Ad hoc Network (VANET) has been proposed by Taherkhani et al. [4]. Although their work is not about network congestion prediction, they succeeded in controlling congestion by clustering data using learning K-means algorithm. In many similar works, authors assumed congestion situation from channel busy level and then tried to propose a mechanism to control network congestion. However, for the first time, in this work, we do not use any assumption for network congestion, we propose a method to predict it, and results will show to which extend the proposed prediction method is accurate and reliable.

Moreover, the concept of fracturing control unit and data plane has been emerged by Software-Defined Networking (SDN) architecture [5], [6]. In SDN, the control layer plays administrative roles in the whole network. Therefore, the control layer is able to update, configure and optimize network resources very fast and dynamically thanks to its programmability attribute [5]. SDN is adaptable, manageable, cost-effective, and ideal for dynamic environment like HetVNET. In this regards, a network congestion prediction model at the control layer, can help in forming and boosting an intelligent network management in SDN based architectures of HetVNET. In this paper, we propose a multiple linear network congestion prediction model for HetVNET.

This paper is organized as follows. In Section II, we present a literature review, and in Section III a methodology and the proposed prediction model. Simulation scenario and numerical

results are presented and discussed in Section IV. Concluding remarks are presented in Section V.

II. RELATED WORK

In the literature, several authors decided to cope with network congestion by adjusting the transmission power [7], [8], [9], [10]. For instance, Ali Shah et al. [8] defined a mechanism to reduce the transmission power in order to control traffic load of control channel in VANET. If a vehicle finds out that control channel is congested, it will inform other vehicles that they may be affected by the congestion problem. Then, vehicles are sorted based on their current transmission power. Vehicles formed several groups, and vehicles belonging to each group start to reduce transmission power fairly. In this method, the congestion is alleviated by all vehicles that are impacted by the effects of network congestion. Chakroun et al. [10] tuned transmitting power of each vehicle based on crowding level surrounding of the vehicle. In the proposed algorithm, as far as local communication congestion does not exceed a communication congestion threshold, the transmitting power is increased, and otherwise, the transmitting power is decreased. In this paper, the value of communication congestion threshold is not defined. Rostami et al. [11] compared the performance of two approaches of reactive state based and linear adaptive approaches. In reactive state based approaches three different states are defined for channel: idle, active, and high traffic load. Active level is divided to three sub sets. Each channel occupancy level has a predefined related message transmission policy in terms of time for transmission message and message transmission rate. In linear adaptive approaches message transmission policy is defined to worthy channel utilization. Simulation results illustrate that the message throughput with the linear adaptive approach is higher than the message throughput with the stable reactive approach. Zang et al. [12], used a static and fixed threshold for channel usage. Different congestion control mechanisms are used according to the channel usage, for more than 95% and for more than 70%. However, as found by the authors, congestion may occurred with lower channel usage figures.

Taherkhani et al. [4] improved packet loss, average delay and probability of collision metrics by applying the K-means clustering technique (unsupervised algorithm). Its proposed strategy is divided in to three parts: congestion detection, data control and congestion control. In the congestion detection unit, it is assumed that congestion is happened whenever channel usage comes up to 70%. In the control unit, messages are collected, filtered and then clustered. In the congestion control unit, appropriate communication parameters are assigned to each cluster.

Lu et al. [13] proposed a method, which reduce the bandwidth assigned to delay tolerant data and adding it to the bandwidth used by sensitive delay data. The solution approach was applied where the channel queue length had been grown more than a threshold and congestion happened. Zemouri et al. [14] proposed a model to predict density around a vehicle in the next time window by using beacons' information. Then based on density prediction, the vehicle can adjust its parameters to avoid congestion for the next time window.

Hasanabadi et al. [15] proposed the Synchronized Persistent Coded Repetition (SPCR) algorithm. With SPCR, each active vehicle node broadcasts composition linear coding of messages, which are selected randomly from its queue. If the number of vehicles in a cluster is N, then the congestion control mechanism randomly selects n nodes as active (which defined as $n \leq N$) and abandons all messages from $(N - n)$ inactive nodes. Therefore, the value of n is from 0 to N. If $n = N$, it means that all N vehicles in the cluster are active and can all broadcast messages. On the other hand, if $n = 0$, it means all nodes are passive and all safety messages are dropped which is dangerous especially in critical situation like road hazards. Kolte et al. [16] defined several segments and assigned each vehicle to a segment. In each segment, one node decides that which of the other nodes of the segment can use dedicated bandwidth during specific time interval. Since, segments densities are not equal, bandwidth allocation is not fair, as a node in a denser segment has to wait more to use dedicated bandwidth. Besides, time of using bandwidth for a node in crowded segment is less than a node in a non-crowded segment.

III. METHODOLOGY AND PREDICTION MODEL

A. Designing Structure of Data set

Inspiring of current works, we consider a group of parameters, which each one has effect on creating congestion in vehicular network. Indeed, analyzing a group of different parameters, which have effect on congestion in vehicular network using learning algorithm, helps us to produce prediction model with high accuracy in congestion prediction result. Besides, assigning a weight to each parameter in prediction model can guide us to find out the importance level of parameters in terms of their effects on congestion in the HetVNET. This approach can guide us towards creating a congestion control mechanism based on most effective parameters on congestion occurrence.

We consider following five vehicular network parameters in this work: Number of Vehicles (V), Data Rate (DR), DSRC Transmission Power (TP_{DSRC}), LTE Transmission Power (TP_{LTE}), and LTE Bandwidth (B).

B. Proposing Utility Function

The network throughput is the amount of data successfully received at the destination point per unit of time. The data generation rate is the amount of data generated by the network nodes per unit of time. As a result, in part of creating data set, we propose utility function like $U(V_t)$ as a metric to detect congestion in the HetVNET. Indeed, congestion recognition is based on two factors of network throughput (τ) and data generation rate (α) by V vehicles. Value of $U(V_t) \in [0, 1]$ for time of t and with V vehicles. We can assume that if the value of $U(V_t)$ grows towards one, the network condition in

terms of congestion improves, and if value of $U(V_t)$ collapsed towards zero then congestion is happened in the network:

$$U(V_t) = \frac{(\tau_{DSRC})(V_t) + (\tau_{LTE})(V_t)}{\alpha(V_t)}, \quad (1)$$

where, $\alpha(V_t)$ is the data generation rate by V vehicles for time unit t. Moreover, we consider total throughput in the heterogeneous vehicular network as sum of throughput of DSRC ($\tau_{DSRC}(V_t)$) and throughput of LTE ($\tau_{LTE}(V_t)$), both based on Bytes per second (Bps). This vision helps us to investigate congestion problem based on sensitivity of urban roads. For various scenarios and based on network sensibility, we define a threshold for value of utility function like T (which $T \in (0,1)$) and based on that, we can define three network congestion states in HetVNET:

$$\text{congestion state} = \begin{cases} \text{state1:}safe\text{, if:} \\ (T+\gamma) < U(V_t) \leq 1 \\ \text{state2:}warning\text{, if:} \\ T < U(V_t) \leq (T+\gamma) \\ \text{state3:}congestion\text{, if:} \\ 0 \leq U(V_t) \leq T \end{cases}, \quad (2)$$

where $\gamma \in (0,1)$ is used to define the warning interval and $(T+\gamma) < 1$. For instance, assume that the HetVNET is implemented in a non-safe road with high risk of car accident and the weather is rainy. In such a scenario, as the required emergency and safety services should be provided smoothly, we may set $U(V_t)$ to 0.4 or over, so T= 0.4. If we assume that γ= 0.2, then based on (2), we have:

$$\text{congestion state} = \begin{cases} \text{state1:}safe\text{, if: } 0.6 < U(V_t) \leq 1 \\ \text{state2:}warning\text{, if: } 0.4 < U(V_t) \leq 0.6 \\ \text{state3:}congestion\text{, if: } 0 \leq U(V_t) \leq 0.4 \end{cases}$$

In this example, the congestion prediction model must make a warning when it predicts that the value of $U(V_t)$ will get less to below 0.6. Then, at this moment, the congestion control/avoidance mechanism will be executed before the value of $U(V_t)$ is collapsed to 0.4, and pushes it up to upper level like beyond 0.6. Therefore, we can be assured that in a critical network situation in terms of data traffic, our target HetVNET can provide at least an acceptable level of network services for vehicular users.

C. Multiple Linear Regression Prediction Model

In order to predict quantitative values such as for $U(V_t)$, linear regression is a popular method [17]. According to the Multiple Linear Regression (MLR) method, we use the least squares method in order to generate a best possible fitted prediction model by minimizing predicting error. It means that by using least square approach we attempted to find model coefficients (β) for our prediction model in the manner of minimizing Residual Sum of Squares (RSS). RSS is the difference between observed values of $U(V_t)$ in training data set and response values that are predicted by the prediction model [17], [18]. Based on MLR, if $X = (x_0, x_1, x_2, \ldots, x_m) = (1, V, DR, TP_{DSRC}, TP_{LTE}, B)$ contains our predictor variables, and β is a set including of our model coefficients (β) which $\beta = (\beta_0, \beta_1, \beta_2, \ldots, \beta_m)$, then the quantitative value of $U(V_t)$ (which we call $\hat{y}$) can be predicted as follows:

$$\hat{y} = \beta_0 + \beta_1 V + \beta_2 DR + \beta_3 TP_{DSRC} + \beta_4 TP_{LTE} + \beta_5 B \quad (3)$$

If we suppose that y is the observed value of $U(V_t)$ in the data set, then $e_i = y_i - \hat{y}_i$ is residual error for i^{th} data record [17], [18]. A prediction model can be trustful, where amount of e_i is at minimum value of itself. To achieve this goal, least squares method can help us find a best fitted prediction model in linear regression problems. Therefore, according to the least squares method, we propose different values for our predictor variables and output Y. Therefore, if we consider a (n×6) matrix of X and a (n×1) matrix of Y as follows (which n is number of observed data records in data set):

$$X = \begin{bmatrix} 1 & V_1 & DR_1 & TP_{DSRC1} & TP_{LTE1} & B_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & V_n & DR_n & TP_{DSRCn} & TP_{LTEn} & B_n \end{bmatrix},$$

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} U_1(V_t) \\ U_2(V_t) \\ \vdots \\ U_n(V_t) \end{bmatrix},$$

then, we can calculate ($X^T \times X$) as a (6×6) matrix with subsequent members:

$$X^T \times X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} & x_{26} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} & x_{36} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} & x_{46} \\ x_{51} & x_{52} & x_{53} & x_{54} & x_{55} & x_{56} \\ x_{61} & x_{62} & x_{63} & x_{64} & x_{65} & x_{66} \end{bmatrix}, \quad (4)$$

where:

$x_{11} = n$	$x_{14} = \sum_{i=1}^{n} (TP_{DSRC})_i$
$x_{21} = \sum_{i=1}^{n} V_i$	$x_{24} = \sum_{i=1}^{n} (V_i)(TP_{DSRC})_i$
$x_{31} = \sum_{i=1}^{n} DR_i$	$x_{34} = \sum_{i=1}^{n} (DR_i)(TP_{DSRC})_i$
$x_{41} = \sum_{i=1}^{n} (TP_{DSRC})_i$	$x_{44} = \sum_{i=1}^{n} (TP_{DSRC})_i^2$
$x_{51} = \sum_{i=1}^{n} (TP_{LTE})_i$	$x_{54} = \sum_{i=1}^{n} (TP_{LTE})_i (TP_{DSRC})_i$
$x_{61} = \sum_{i=1}^{n} B_i$	$x_{64} = \sum_{i=1}^{n} (B_i)(TP_{DSRC})_i$
$x_{12} = \sum_{i=1}^{n} V_i$	$x_{15} = \sum_{i=1}^{n} (TP_{LTE})_i$
$x_{22} = \sum_{i=1}^{n} (V_i)^2$	$x_{25} = \sum_{i=1}^{n} (V_i)(TP_{LTE})_i$
$x_{32} = \sum_{i=1}^{n} (DR_i)(V_i)$	$x_{35} = \sum_{i=1}^{n} (DR_i)(TP_{LTE})_i$
$x_{42} = \sum_{i=1}^{n} (TP_{DSRC})_i (V_i)$	$x_{45} = \sum_{i=1}^{n} (TP_{DSRC})_i (TP_{LTE})_i$
$x_{52} = \sum_{i=1}^{n} (TP_{LTE})_i (V_i)$	$x_{55} = \sum_{i=1}^{n} (TP_{LTE})_i^2$
$x_{62} = \sum_{i=1}^{n} (B_i)(V_i)$	$x_{65} = \sum_{i=1}^{n} (B_i)(TP_{LTE})_i$
$x_{13} = \sum_{i=1}^{n} (DR_i)$	$x_{16} = \sum_{i=1}^{n} (B_i)$
$x_{23} = \sum_{i=1}^{n} (V_i)(DR_i)$	$x_{26} = \sum_{i=1}^{n} (V_i)(B_i)$
$x_{33} = \sum_{i=1}^{n} (DR_i)^2$	$x_{36} = \sum_{i=1}^{n} (DR_i)(B_i)$
$x_{43} = \sum_{i=1}^{n} (TP_{DSRC})_i (DR_i)$	$x_{46} = \sum_{i=1}^{n} (TP_{DSRC})_i (B_i)$
$x_{53} = \sum_{i=1}^{n} (TP_{LTE})_i (DR_i)$	$x_{56} = \sum_{i=1}^{n} (TP_{LTE})_i (B_i)$
$x_{63} = \sum_{i=1}^{n} (B_i)(DR_i)$	$x_{66} = \sum_{i=1}^{n} (B_i)^2$

therefore, we can calculate $X^T \times Y$ as follows:

$$X^T \times Y = \begin{bmatrix} \sum_{i=1}^{n} U_i(V_t) \\ \sum_{i=1}^{n} V_i \times U_i(V_t) \\ \sum_{i=1}^{n} DR_i \times U_i(V_t) \\ \sum_{i=1}^{n} (TP_{DSRC})_i \times U_i(V_t) \\ \sum_{i=1}^{n} (TP_{LTE})_i \times U_i(V_t) \\ \sum_{i=1}^{n} B_i \times U_i(V_t) \end{bmatrix}, \quad (5)$$

Finally, β contains the proposed model coefficients is computable using (6), and the multiple linear regression congestion prediction model can be completed:

$$\beta = (X^T \times X)^{-1} \times (X^T \times Y) = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \end{bmatrix}. \quad (6)$$

IV. SIMULATION SCENARIO AND NUMERICAL RESULTS

A. Simulation Scenario

In this paper, as part of data generation and towards generating urban simulation scenario, we use OpenStreetMap (OSM) to create map of boroughs of the city of Montreal in Canada. Then, we used the ".osm" file in Simulation of Urban Mobility (SUMO) 0.26.0 to generate road traffic. Finally, we worked with Veins LTE version 1.3 [19], which is standing on the OMNeT++ (4.6) Network simulator to simulate heterogeneous vehicular network based on IEEE 802.11p and LTE.

TABLE I
PARAMETERS AND CORRESPONDING VALUES USED IN SIMULATION SCENARIO

Parameter	Value
Total road length	1000 m ×1000 m
Number of lanes	4 (two in each direction)
Number of vehicles	30, 50, 100, 150, 200
Number of base station (eNB)	1
Bandwidth (IEEE802.11p)	10MHz
Bandwidth (LTE)	5 MHz, 10MHz, 20MHz
Transmission power (IEEE802.11p)	1 mW, 50 mW, 100 mW
Transmission rate (IEEE802.11p)	6-27 Mbps
Resource Blocks size	25, 50, 100
Message size	400 Bytes
Vehicles speed	0-40 km/h
Propagation model	Nakagami
Simulation time	1000 s
Simulation runs	260

Table I contains attributes and parameters values, which are applied in each of the 260 running simulation scenarios. For each scenario, we set the values of V, DR, TP_{DSRC}, TP_{LTE}, and B and then calculated the value of $U(V_t)$.

After generating data extracted from executing simulation scenarios and putting it in shape of a data set, we use R programming language (using RStudio version 1.1.463) in order to create multiple linear regression congestion prediction models and statistically analyzing their performance to finally find a congestion prediction model most fitted to the observed data.

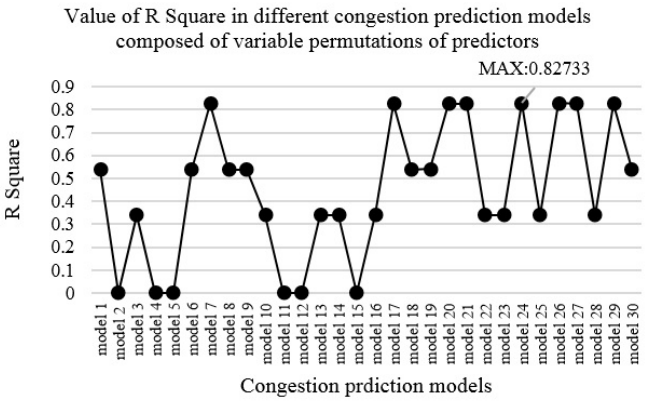

Fig. 1. Value of R^2 for each possible 30 prediction models.

B. Multiple Linear Regression Analysis: Assessing Congestion Prediction Model

In the current HetVNET related works, we did not find any work that could be used as a benchmark (until today), and make comparison with our proposed method. Therefore, in order to evaluate congestion prediction model generated by multiple linear regression method, we will respond to the following questions, which are mainly considered in regression problems:

Q1) How much of variability in amount of $U(V_t)$ can be expressed by predictor variables (V, DR, TP_{DSRC}, TP_{LTE}, and B) in congestion prediction model? Or using a subset of predictor variables is more effective for predicting $U(V_t)$ than having a congestion prediction model contains all five predictor variables? R square parameter can show us that how much changing in value of dependent variable like $U(V_t)$ is determined by independent variables like V, DR, TP_{DSRC}, TP_{LTE}, and B in our problem [17]. Fig. 1 illustrates amount of R^2 for each possible 30 prediction models, which are generated using information of data set and MLR method.

Fig. 1 shows that in model 24, which we apply all five predictor variables to make a congestion prediction model, as we expected, it has highest coefficient of determination (R^2) among all possible 30 congestion prediction models. It confirms that, information from the variables like number of vehicles, data rate, DSRC transmission power, LTE transmission power, and LTE bandwidth are helpful to better predict the network performance in terms of problem in data transmission in HetVNET. The regression congestion prediction model 24 composed of all five considered variables, with coefficient of determination of 0.82733 (which is most close to one among other models), is most capable to express variability in amount of $U(V_t)$. Therefore, in continue we just consider information of all five predictor variables in order to generate congestion prediction models using MLR.

Q2) Is proposed congestion prediction model a reliable model? Based on Cross-Validation test method [17], we considered 80% of our observed data in a training data set and

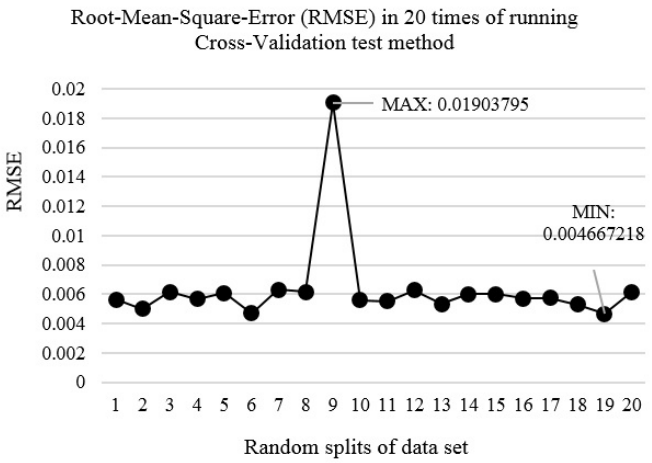

Fig. 2. RMSE in different splits of data set using cross-validation test approach.

remaining 20% are used in a test data set. We performed this approach 20 times and for each time data belonging to training data set and test data set are selected randomly among 260 data records of our observed data. In each split of training data set, the MLR algorithm generates a model based on data of training data set.

TABLE II
STATISTICAL PARAMETERS ABOUT CONGESTION PREDICTION MODEL 19

	Estimate	**Std. Error**	**t value**	**Pr**($>|t|$)
(Intercept)	-1.036e-02	1.943e-02	-0.533	0.594
V	1.555e-04	6.799e-06	22.876	>2e-16
DR	1.921e-06	5.555e-05	0.035	0.972
TP_{DSRC}	1.754e-04	1.036e-05	16.930	>2e-16
TP_{LTE}	2.371e-04	4.482e-04	0.529	0.597
B	3.281e-05	1.033e-04	0.318	0.751

We evaluate accuracy of generated prediction models using test data sets and based on Root Mean Square Error (RMSE) parameter, which is one of the best parameter to show how much the prediction model can be trustful in terms of producing results close to actual data [20]. Fig. 2 shows value of RMSE for different splits of data set using cross-validation test approach. In prediction problems, a model with less RMSE is more accurate than other model with higher RMSE value. As Fig. 2 indicates, the prediction model 19 has minimum amount of RMSE in compare to other congestion prediction models. Table II contains coefficients of congestion prediction model 19 based on least squares in multiple linear regression method. We can see values of matrix β for model 19 (related to formula (6)) in estimate column of Table II. Each of the five predictor variables has its own level of effectiveness on $U(V_t)$. Estimate column in Table II, shows effect of each five predictor variable on smooth data transmission. Based on the estimate column, transmission power of LTE has highest effect on boosting data transmission by increasing value of respond $U(V_t)$. DSRC transmission power has been placed at second level of importance in terms of having contribute on enhancing $U(V_t)$.

Table III provides other information about the congestion prediction model 19, as well. F-statistic close to one indicates no relationship between$U(V_t)$ and five model's predictors [21]. However, as Table III illustrates, F-statistic factor is far from one, which emphasizes that at least one of predictor variables of number of vehicles, data rate, DSRC transmission power, LTE transmission power, and LTE bandwidth has strong relationship with $U(V_t)$.

Even if we propose the most fitted prediction model, we could not say that our model can predict exactly observed value of $U(V_t)$ with hundred percent of accuracy in prediction results. If we apply the congestion prediction model, Residual Standard Error (RSE) helps us to estimate variance of the error (σ^2). Therefore, from Table III, we can say that the proposed congestion prediction model has a variance of error of about σ^2 = 0.00607. Based on this value, we infer that the predicted value of $U(V_t)$) is as much as 0.0.00607 different from exact observed value of $U(V_t)$.

All the assessments in this work is based on prediction models that are made from analyzing and learning of data, which are generated by simulator tools. Having more and more data generated from real HetVNET could help us toward proposing congestion prediction models closer to real situations of HetVNE.

TABLE III
STATISTICAL PARAMETERS ABOUT CONGESTION PREDICTION MODEL 19

Parameter	**Value**
Residual Standard Error	0.00607
R^2	0.801
Mean Square Error	0.000021
F-statistic	162.7

V. CONCLUSION

In the current literature related to congestion problem in vehicular networks, only a few authors applied intelligent methods using machine learning algorithms. The reason for that could be the absence of data needed for analyzing, learning and making congestion prediction models applicable to HetVNET. In this paper, we proposed a utility function to explain how the heterogeneous vehicular network can satisfy its vehicular users in terms of smooth transmitting of data. Besides, we explained about how the proposed utility function can help in having a tolerate HetVNET, which can provide required services even in critical network traffic situation. Afterwards, we move toward generating a congestion prediction model, which can predict the utility function. We generate a data set containing information records extracted from simulation scenarios of HetVNET using Veins LTE 1.3 and SUMO 0.26.0. Moreover, we propose congestion prediction model using multiple linear regression, which is a supervised machine learning method. We evaluate reliability

of the proposed congestion prediction model in terms of accuracy in predicted result by using various statistical metrics such as RMSE, coefficient of determination (R^2), and F-statistic. The approach for predicting congestion in such a proposed tolerable manner with respect to the target network's conditions and based on predicted value of defined utility function, can be applied for 5G based SDN architectures as well.

REFERENCES

[1] K. Zheng, X. Wang, Y. Li, and P. Chatzimisios, "Ieee access special section editorial: Communication, control, and computation issues in heterogeneous vehicular networks," *IEEE Access*, vol. 6, pp. 79 285–79 287, 2018.

[2] K. Zheng, Q. Zheng, P. Chatzimisios, W. Xiang, and Y. Zhou, "Heterogeneous vehicular networking: a survey on architecture, challenges, and solutions," *IEEE communications surveys tutorials*, vol. 17, no. 4, pp. 2377–2396, Jun 2015.

[3] R. Sathya and A. Abraham, "Comparison of supervised and unsupervised learning algorithms for pattern classification," *International Journal of Advanced Research in Artificial Intelligence*, vol. 2, no. 2, pp. 34–38, 2013.

[4] N. Taherkhani and S. Pierre, "Centralized and localized data congestion control strategy for vehicular ad hoc networks using a machine learning clustering algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3275–3285, Nov. 2016.

[5] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network and openflow: From concept to implementation," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 2181–2206, 2014.

[6] J. Liu, Y. Li, M. Chen, W. Dong, and D. Jin, "Software-defined internet of things for smart urban sensing," *IEEE communications magazine*, vol. 53, no. 9, pp. 55–63, 2015.

[7] B. Aygun, M. Boban, and A. M. Wyglinski, "Ecpr: Environment-and context-aware combined power and rate distributed congestion control for vehicular communications," *Computer Communications*, vol. 93, pp. 3–16, 2016.

[8] S. A. A. Shah, E. Ahmed, J. J. Rodrigues, I. Ali, and R. M. Noor, "Shapely value perspective on adapting transmit power for periodic vehicular communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 977–986, 2018.

[9] E. Egea-Lopez and P. Pavon-Mariño, "Fair congestion control in vehicular networks with beaconing rate adaptation at multiple transmit powers," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3888–3903, 2016.

[10] O. Chakroun and S. Cherkaoui, "Overhead-free congestion control and data dissemination for 802.11 p vanets," *Vehicular Communications*, vol. 1, no. 3, pp. 123–133, 2014.

[11] A. Rostami, B. Cheng, G. Bansal, K. Sjöberg, M. Gruteser, and J. B. Kenney, "Stability challenges and enhancements for vehicular channel congestion control approaches," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2935–2948, 2016.

[12] Y. Zang, L. Stibor, X. Cheng, H.-J. Reumerman, A. Paruzel, and A. Barroso, "Congestion control in wireless networks for vehicular safety applications," in *Proceedings of the 8th European Wireless Conference*, vol. 7, 2007, p. 1.

[13] Y. Lu, Z. Ling, S. Zhu, and L. Tang, "Sdtcp: Towards datacenter tcp congestion control with sdn for iot applications," *Sensors*, vol. 17, no. 1, p. 109, 2017.

[14] S. Zemouri, S. Djahel, and J. Murphy, "A short-term vehicular density prediction scheme for enhanced beaconing control," in *2015 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2015, pp. 1–7.

[15] B. Hassanabadi and S. Valaee, "Reliable periodic safety message broadcasting in vanets using network coding," *IEEE transactions on wireless communications*, vol. 13, no. 3, pp. 1284–1297, 2014.

[16] S. R. Kolte and M. S. Madnkar, "A design approach of congestion control for safety critical message transmission in vanet," in *2014 Fourth International Conference on Communication Systems and Network Technologies*. IEEE, 2014, pp. 298–301.

[17] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013, vol. 112, pp. 181–184.

[18] S. Weisberg, *Applied linear regression*, vol. 528.

[19] F. Hagenauer, F. Dressler, and C. Sommer, "Poster: A simulator for heterogeneous vehicular networks," in *2014 IEEE Vehicular Networking Conference (VNC)*. IEEE, 2014, pp. 185–186.

[20] C. J. Willmott, "Some comments on the evaluation of model performance," *Bulletin of the American Meteorological Society*, vol. 63, no. 11, pp. 1309–1313, 1982.