A New Gateway Selection Algorithm Based on Multi-Objective Integer Programming and Reinforcement Learning

Hasanain Alabbas, and Árpád Huszák

***Abstract*—Connecting vehicles to the infrastructure and benefiting from the services provided by the network is one of the main objectives to increase safety and provide well-being for passengers. Providing such services requires finding suitable gateways to connect the source vehicles to the infrastructure. The major feature of using gateways is to decrease the load of the network infrastructure resources so that each gateway is responsible for a group of vehicles. Unfortunately, the implementation of this goal is facing many challenges, including the highly dynamic topology of VANETs, which causes network instability, and the deployment of applications with high bandwidth demand that can cause network congestion, particularly in urban areas with a high-density vehicle. This work introduces a novel gateway selection algorithm for vehicular networks in urban areas, consisting of two phases. The first phase identifies the best gateways among the deployed vehicles using multi-objective integer programming. While in the second phase, reinforcement learning is employed to select a suitable gateway for any vehicular node in need to access the VANET infrastructure. The proposed model is evaluated and compared to other existing solutions. The obtained results show the efficiency of the proposed system in identifying and selecting the gateways.**

***Index Terms*—VANET, gateway selection, multi-objective integer programming, reinforcement learning.**

I. Introduction

THE Vehicular Ad Hoc Networks (VANETs) represent the vital nerve of the Intelligent Transportation System (ITS) as the research, and industrial communities have become increasingly interested in developing VANETs [1]. In general, VANET's infrastructure consists of vehicles equipped with a communication device and Road Side Units (RSUs), which are fixed communication units located near intersections or distributed on the side of the roads [2], [3]. The communication in the VANET environment is divided into two types, namely: Vehicle-to-Vehicles (V2V), which allows the vehicles to communicate directly, and Vehicle-to-Infrastructure (V2I), in which the vehicles are able to make contact with the infrastructure like routers, base stations, and RSUs [4], [2]. The main drive for V2I development is providing the drivers with the necessary information and assistance to increase safety and decrease accidents, as well as providing Internet access for entertainment [5], [6]. However, with the increasing growth of greedy Internet applications, providing the Internet connectivity for vehicular users has become an urgent need, especially in urban areas. As a consequence, new concepts have emerged dedicated to this purpose, like Urban Vehicular Ad Hoc Network (UVANET), which deals with non-safety applications (Internet service, media sharing, and data sharing) [7]. Providing the Internet for vehicles requires finding a suitable gateway. Unfortunately, the implementation of this goal is facing many challenges. The applications with high bandwidth demand can cause network congestion, particularly in urban areas with a high density of vehicles. In addition, the VANET environment is characterized as a high dynamic environment because of the high speed of vehicles that connect or disconnect to the network very frequently, causing unstable network connections [8].

Gateway selection strategies often rely on inquiry and solicitation messages sent and received between the vehicular nodes (VNs) to find a suitable gateway [8]. These kinds of messages overwhelm the networks and can cause broadcast storm problems and overhead when the number of nodes increases [9]. Investing in cloud computing and making it compatible with ITS, provides a valuable opportunity to benefit from cloud computing resources utilized by VANET services [10], [11]. This union produced a new paradigm called Vehicular Cloud (VC) [12] [13]. VC presents many services like network information collection, traffic control optimization, and congestion detection [12]. Because of the massive services and features provided by VC, we will use it to build our gateway selection model, so we can reduce the impact of overhead in the network. The gateways aim to provide the Internet for vehicles that need it. The identified gateways are employed to connect the source vehicles to the infrastructure. The major feature of using the gateways is to decrease the load of network infrastructure resources. Each gateway is responsible for a group of vehicles by handling and multiplexing the traffic amount of the group to send them to the infrastructure. It should be noted; our proposed system is the extension of our previous work entitled "Reinforcement Learning based Gateway Selection in VANETs" . In the previous work, we assumed that the public transport buses are equipped with Internet access and can serve as mobile gateways (MGs) [13]. We used reinforcement learning to select the best gateway for each vehicle that needs Internet access. We are now expanding the scope of our work to include defining the gateways instead of assuming them, and this

Hasanain Alabbas is with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Hungary, and Computer Center Department, Al-Qasim Green University, E-mail: hasanain@hit.bme.hu

Árpád Huszák is with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, and ELKH-BME, Cloud Applications Research Group, Hungary, E-mail: huszak@hit.bme.hu

DOI: 10.36244/ICJ.2022.4.1

leads us to find a mechanism to identify the gateways. Our proposed model now consists of two phases instead of one phase. For the first phase, we utilized the Multi-Objective Integer Programming (MO-IP) to define the best gateways in terms of speed, distance from the base station and geographical distribution. In the second phase, we adopted reinforcement learning to find the best gateway for client vehicles.

II. Related Works

Providing a stable connection to VANET infrastructure is a considerable challenge because of the highly dynamic environment. In [14], the authors proposed a gateway selection strategy to access the information and retrieve the data from the cloud by using epidemic spread routing (ESR). The content accessibility preference (CAP) model has been used to confine the greedy behavior of ESR and minimize the data access delay. In [15], a fuzzy multi-metric qos-balancing gateway selection algorithm (FQGwS) was proposed to provide stable communication and increase the link connectivity duration between the vehicles and LTE infrastructure. The LTE Advanced eNodeBs are employed as fixed gateways. The communication between vehicles and the LTE infrastructure is directly or via a relay gateway. The fuzzy logic is adopted to select the best gateways based on a blend of metrics like signal strength, resources occupation, connection lifetime, and QoS traffic classes. However, this kind of solution uses the reactive approach where vehicles exchange messages to discover an appropriate gateway, thus causing a high amount of overhead. The authors [16] proposed a routing strategy to provide Internet access for vehicles by selecting a suitable mobile gateway. The study utilized the vehicle's characteristics (speed, direction, position) to determine the best mobile gateway. On the other hand, it calculates the trust parameter to determine if the connection is reliable and secure or not. Driss et al. [17] proposed a gateway selection algorithm based on heterogeneous VANET and 4G LTE cellular networks. The study considered that the vehicles fitted with 4G LTE and IEEE 802.11p-based-VANETs interfaces are potential mobile gateways. These possible gateways can provide a reliable connection with the 4G LTE cellular networks to ordinary vehicles. The study took into account several factors for the selection of the mobile gateways, such as signal strength, vehicles' movement, and path length.

However, even though these kinds of solutions show good results in terms of packet delivery when applied in a highway scenario, they are not suitable in urban scenarios. The proactive and reactive strategies used in these algorithms can decrease the throughput when the vehicle numbers increase. Moreover, there is no optimization in the selection procedure.

In [18], the authors suggested a new gateway selection system by using multi-objective optimization to address the issues generated by the previous studies. The system takes into consideration two contradicting objectives. The first objective aims to maximize the number of connected vehicles while the second one aims to minimize the overload of the gateways.

We present a novel model to identify and select the mobile gateways using Multi-Objective Integer Programming (MO-IP) and Reinforcement Learning (RL) in urban scenarios.

III. System Model

Our system model is a hybrid network architecture composed of a VANET, VANET's infrastructure (4G/5G base station, RSU), and Vehicular Cloud (VC). We assume all vehicles in VANET are equipped with an On-Board Unit (OBU). So that they can communicate with each other and with the infrastructure, as stated in Figure 1. We propose a

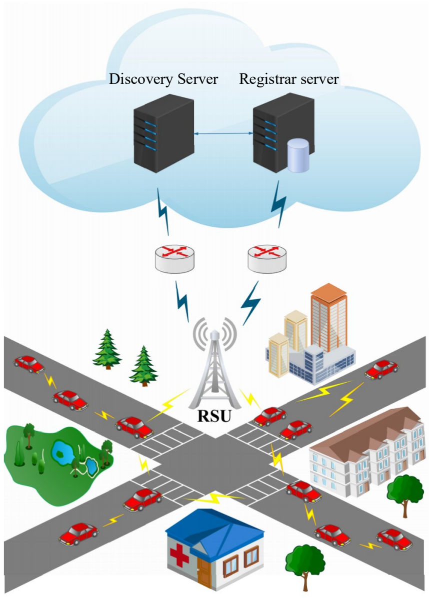

Fig. 1: System architecture.

centralized gateway selection system that aims to identify the gateways and allocate them to vehicles needing of Internet access. Unlike the decentralized strategies in literature, the centralized mechanism reduces the overload situation when there are many nodes in the network.

Our proposed system is based on the VC, which consists of two servers, namely Registrar and Discovery servers. The Registrar Server accumulates the necessary information about vehicles movements and the infrastructure network. It calculates the Link Connectivity Duration (LCD) between them. On the other hand, the proposed system is integrated into the discovery server. Our proposed algorithm is an extension of our previous study [13]. In the previous work, we assumed that the gateways are public transport buses connected directly to the Internet. Based on this assumption, we used reinforcement learning to discover a suitable gateway for source vehicles. In our current work, we aim to make our algorithm more general and comprehensive. The development and expansion is the use of a mechanism to identify the gateways instead of assuming their existence. Therefore, our proposed system consists of two phases:

1) Gateways Identification: we use Integer Programming (IP) to identify the gateways.
2) Gateway Selection: we adopt reinforcement learning to allocate a suitable gateway for ordinary vehicles.

A. Gateways Identification Phase

In this phase, we propose an algorithm seeking the gateways based on different objectives. These objectives ensure that the gateways are close to the infrastructure with lowest relative speed and the highest number of neighbors.

Despite the good results achieved by reinforcement learning to select the best gateways in the previous study compared to MOO, it cannot be applied to the gateway identification phase. In the gateway identification phase, all gateways must be determined by a single decision. Since each gateway has its own criteria for identifying it, the final outcome of the reward function becomes too complex and as a result, the agent becomes confused and unable to learn.

Regarding the run-time complexity, IP is considered NP-complete and can be affected by the number of variables and constraints while RL agent should be trained in a simulated environment. Once the neural network is set up, the decision is very fast.

The MO-IP technique is employed to optimize the gateways discovery. It aims to find optimal solution based on different objectives. The main challenge faced by this type of optimization is finding a compromise solution among Pareto optimal solutions. Pareto optimal solution refers to a non-dominated solution, which means none of objective functions can be improved without making the other objective values degrade. In the rest of this subsection, we formulate the gateways discovery problem. VANET consists of a set of vehicular nodes (VNs), which is represented by $\mathcal{VN}$ and a set of Base Stations (BSs), which is represented by the $\mathcal{BS}$. The distance between a VN $i \in \mathcal{VN}$ and a BS $j \in \mathcal{BS}$ is represented by d_{ij}, while the distance between VNs is denoted by $d_{i_1 i_2}$ where $i_1 \in \mathcal{VN}$ and $i_2 \in \mathcal{VN}$. Let $\mathcal{V}_i$ denotes the velocity of the VN i. $\mathcal{D}_i$ is the direction of a VN i and Nie_i is the number of neighbors located under the VN i range. $\mathcal{U}(\mathcal{VN})$ denotes a binary vector where $\mathcal{U}(i)$ = 1 if the VN i is selected as a gateway, else $\mathcal{U}(i)$ = 0. The relationship between VNs and BSs is represented through the binary matrix $\mathcal{X}(\mathcal{VN},\mathcal{BS})$. If and only if the VN i is selected as a Gateway (GW) to the BS j, then $\mathcal{X}(i,j)$ = 1, otherwise $\mathcal{X}(i,j)$ = 0. The binary symmetric matrix $\mathcal{Y}(\mathcal{VN},\mathcal{VN})$ is defined, if and only if $i_1 \in \mathcal{VN}$ and $i_2 \in \mathcal{VN}$ are Gateways, then $y(i_1,i_2)$ = 1, otherwise $y(i_1,i_2)$ = 0. The gateway identification problem is expressed by the integer program as follow.

$$f = \alpha \sum_{j=1}^{M}\sum_{i=1}^{N} d_{ij}\mathcal{X}(i,j) - \beta \sum_{j=1}^{M}\sum_{i=1}^{N} \mathcal{V}_i \mathcal{X}(i,j) - \gamma \sum_{j=1}^{M}\sum_{i=1}^{N} Nie_i \mathcal{X}(i,j) \quad (1a)$$

Subject to

$$\forall i \in \mathcal{VN}, \forall j \in \mathcal{BS}, d_{ij}\mathcal{X}(i,j) \leq r \quad (1b)$$

$$\forall i \in \mathcal{VN}, \sum_{i \in \mathcal{VN}} \mathcal{U}(i) \leq \mathcal{N} \quad (1c)$$

$$\forall i \in \mathcal{VN}, \forall j \in \mathcal{BS}, \sum_{j \in \mathcal{BS}} \mathcal{X}(i,j) = \mathcal{U}(i) \quad (1d)$$

$$(100 - d_{i_1 i_2})\mathcal{U}(i_1)\mathcal{U}(i_2) \leq \mathcal{M}\mathcal{Y}(i_1,i_2) \quad (1e)$$

$$(45 - |\mathcal{D}i_1 - \mathcal{D}i_2|)\mathcal{U}(i_1)\mathcal{U}(i_2) \leq \mathcal{M}(1 - \mathcal{Y}(i_1,i_2)) \quad (1f)$$

$$\forall i_1 \in \mathcal{VN}, \forall i_2 \in \mathcal{VN}, \mathcal{Y}(i_1,i_2) = \mathcal{Y}(i_2,i_1) \quad (1g)$$

$$\forall i \in \mathcal{VN}, \forall j \in \mathcal{BS}, \mathcal{X}(i,j) \in \{0,1\} \quad (1h)$$

$$\forall i \in \mathcal{VN}, \mathcal{U}(i) \in \{0,1\} \quad (1i)$$

$$\forall i_1 \in \mathcal{VN}, \forall i_2 \in \mathcal{VN}, \mathcal{Y}(i_1,i_2) \in \{0,1\} \quad (1j)$$

The integer programming model consists of three objective functions. The first objective aims to find a GW having the minimum distance with the BS. The second one is used to find a GW with the highest number of neighbors, while the third one aims to identify the lowest speed VN as a GW. Therefore, the utility function (1a). α, β, and γ are the objectives weights so that $\alpha + \beta + \gamma = 1$. The set of constraints are explained as follow:

- Constraint (1b) is used to ensures that VN i, if selected as a GW to a BS j then the distance between i and j must not exceed the range r.
- Constraint (1c) is used to restrict the number of GWs, where $\mathcal{N}$ is the number of the GWs.
- Constraint (1d) is used to ensure that every GW is connected to only one BS.
- Constraint (1e) and constraint (1f) represent if-then constraint which ensure that if the difference in direction between GW i_1 and GW i_2 is less than 45, then the distance between them must be greater than 100 m. $\mathcal{M}$ is a large number enough to bound the difference. These two constraints ensure that the GWs moving in the same direction are not concentrated in a certain area more than the others.
- Constraint (1g) is used to ensure that the matrix $\mathcal{Y}$ is symmetric.
- Constraints (1h), (1i), and (1j) are integrality constraints.

B. Reinforcement Learning

Reinforcement Learning (RL) is an area of machine learning inspired by human interaction with the environment to learn skills. The main parts of the RL system are the agent and the environment. RL is modeled by a Markov decision process. The concepts (state (S), action (A), reward (R)) represent the interaction of the agent with its environment. At each time (t), the agent senses the environment state (s_t) and takes action (a_t) from the set of available actions causing a state transition to a new state (s_{t+1}). The agent obtains a reward (r_t) that indicates whether the decision taken is correct or not.

The mapping between the action (a) and the state (s) is denoted by the policy $\pi(a, s)$. The policy $\pi(a, s)$ reflects the behavior of agent during sensing its environment.

The agent seeks the optimal policy $\pi^*(a, s)$ by maximizing accumulated discounted reward for each $s \in S$ and $a \in A$ expressed in Equation (3):

$$\pi^*(a \mid s) = \underset{\pi(a|s)}{\arg\max} \sum_{t=t_0}^{t_{end}} \gamma^{t-t_0} r_t \tag{2}$$

where $\gamma \in (0, 1)$ is the discount factor and t is the time horizon. Policy optimization algorithms can be categorized into two groups which are value-based algorithms and policy-based algorithms. Although the policy-based algorithms have better convergence and are more convenient for large action spaces but they have some shortcomings. Proximal Policy Optimization (PPO) [19] combines the value-based and policy-based features by using two neural networks called actor-critic. The first one, named $actor$, takes the state (s) as entries and outputs the policy $\pi(a, s)$, while the second one, named $critic$, optimizes $V(s)$ that measures the goodness of the action (a). PPO uses the advantage function $A(s, a)$ to reduce the estimation variance.

PPO uses the Trust Region Policy Optimization(TRPO) strategy to ensure that the new updated policy never goes far away from the old policy, making it more stable and reliable. For these reasons, we adopt PPO algorithm in our proposed gateway allocation phase, namely RL-agent.

C. Gateway Allocation Phase

After electing a specific set of VNs to be gateways to the infrastructure in the first phase, the second phase is concerned with assigning an appropriate GW to each VN that needs access to the Internet or infrastructure network services. RL mechanism is adapted to achieve this goal. Three main parts must be accurately identified to enable the RL agent to sense the VANET environment and make the right decision: state, action, and reward.

1) Definition of Observation State: The state (s) will be created for each VN i that needs access to the infrastructure network and looks for a connection to a suitable GW j. It represents the relationship between the VN and all the identified GWs in terms of geographical location, speed, and available bandwidth. The state is expressed by the entries as follow:

$$X = \begin{bmatrix} Lo_{i1} & Lat_{i1} & V_{i1} & \theta_{i1} & C_1 \\ Lo_{i2} & Lat_{i2} & V_{i2} & \theta_{i2} & C_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Lo_{ij} & Lat_{ij} & V_{ij} & \theta_{ij} & C_j \end{bmatrix} \tag{3}$$

- Lo_{ij}, Lat_{ij}, V_{ij}, and θ_{ij} represent the difference in longitude, latitude, velocity, and direction between VN i and GW j, respectively.
- C_j denotes the available capacity of a GW j.

Since the relationship of the VN to each MG is represented by five parameters $S = (Lo, Lat, V, \theta, C)$, the total number of entries to represent the state is $|S| \cdot |GW|$,where $|S|$ is the number of parameters used to describe a state, while $|GW|$ is the number of MGs.

2) Definition of Agent Action and Rewards: The action space represents all possible actions. Since the agent's action is to assign an appropriate GW to each VN trying to reach the infrastructure, the action space represents all GWs. Action $a = \{a_1 a_2 a_3 \ldots a_n\}$, where a_1 represents the selection of GW_1 and a_n stands for the selection of GW_n. The reward function is assigned based on two metrics: the first one is the link connectivity duration between the VN and GW, whereas the second one is the GW capacity. The first metric motivates the agent to find a GW with the highest LCD for each VN, while the second one aims to reduce the VNs connected to the same GW. Multi-Objective Reinforcement Learning (MORL) is adapted to set the reward function by finding a compromise between the objectives. The reward function is expressed as:

$$R = w_1 \cdot lcd_{ij} + w_2 \cdot C_j \tag{4}$$

where lcd_{ij} denotes the link connectivity duration value between VN i and MG j, while C_j represents the available GW j capacity. Parameters w_1 and w_2 take values between 0 and 1 depending on the importance of the objectives so that w_1 + w_2 = 1. The reward value is positive when the action is valid otherwise, the reward is negative. The positive reward ranges in value between 0 and 20, while the negative reward is (-4). The negative reward is applied in two cases:

1) The GW allocated to a VN is out of its coverage range.
2) The allocated GW does not have enough traffic amount.

As mentioned above, the reward value was adopted after training the agent several times with different reward values because the assumed value showed a faster response from the agent to learn.

3) Agent parameters: The GW allocation system depends mainly on the dataset collected by the registrar server. The dataset consists of a huge number of snapshots collected from the VANET environment. The process of adding a snapshot to the dataset takes place after defining the GWs in the first phase. The snapshot is divided into a number of entries so that the number of entries in each snapshot is equal to the number of VNs. Each entry represents the relationship between each VN with all the GWs in terms of the Cartesian coordinates, speed, LCD, and the available bandwidth for each GW. PPO is employed to maximize the GWs selection return. The reward r is a multi-objective reward in which the agent tends to find a GW for a VN with the maximum LCD and minimum number of VNs connected to it. The dataset is employed to train the RL agent.

IV. RESULTS

The simulation results are presented to show the efficiency of our proposed algorithm. Simulations are implemented by combining Python programming language, Urban Mobility simulator (SUMO), and Open Street Map (OSM). SUMO is used to simulate the vehicles' mobility [20], while OSM is used to extract real-world map data, which makes the simulation more realistic [21]. Gurobi Optimizer is executed to solve the MO-IP problem used in the gateway identification phase [22], while Baseline3 library is used to implement the RL problem used in gateway allocation phase [23]. The entire simulation parameters are listed in Table I.

TABLE I
SIMULATION PARAMETERS.

Parameters	Setting
Simulation area	1500 m X 1500 m
Transmission Range	500 m
Vehicles speed	0-20 m/s
Vehicles Number	120-200

The nature of the roads in urban areas in terms of the spread of intersections, speed limitation, and traffic jams makes them more challenging than the highway environment. It can be considered a true measure of algorithms performance. The number of VNs deployed in the simulation network is 120-220. The MO-IP algorithm used in the gateways identification phase is evaluated and compared with the Fuzzy Multi-metric QoS-balancing Gateway Selection Algorithm (FQGwS) [15] which uses Received Signal Strength (RSS) metric between the VNs and the infrastructure to discover the potential gateways candidates. Three metrics have been used for the performance evaluation in which metric 1 represents the number of GWs' neighbors, metric 2 represents the velocity of GWs, and metric 3 represents the connection lifetime duration between GWs and VANET's infrastructure. Figure 2 which represents the relationship between metric 1 and metric 2 shows that our proposed algorithm has a good trade-off compared to FQGwS algorithm by finding GWs with low speed and a high number of neighbors. On the other hand, Figure 3 which represents the relationship between metric 1 and metric 3 shows our proposed algorithm has better results in terms of choosing GWs with the lowest speed in comparison with FQGwS, but for the connection lifetime metric, the results are approximately similar. In Figure 4, the 3D diagram is depicted, which combines all the metrics. It should be noted, the results plotted in these figures are collected from 10 times of executions for different scenarios.

In the gateway allocation phase, Reinforcement Learning agent (RL-agent) performance is evaluated based on the number of connected VNs and the VNs distribution among GWs. All approaches are simulated and executed under the same conditions. Each scenario is executed and evaluated multiple times so that each point in the plot shows the mean of 10 executions with a variance representing the error in the error-bar plots. Two case studies are applied to make sure our algorithm is efficient under different conditions so that the GWs are either with a bandwidth limitation constraint or without.

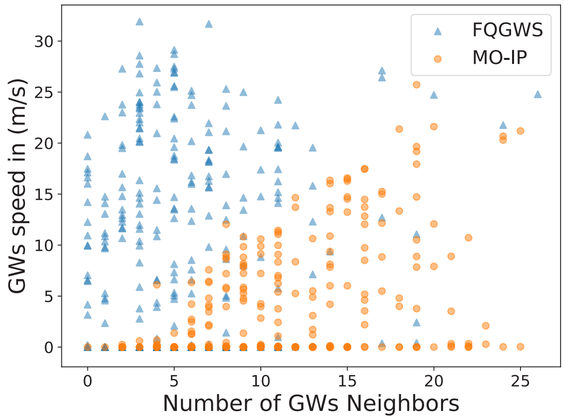

Fig. 2: Number of GWs neighbors.

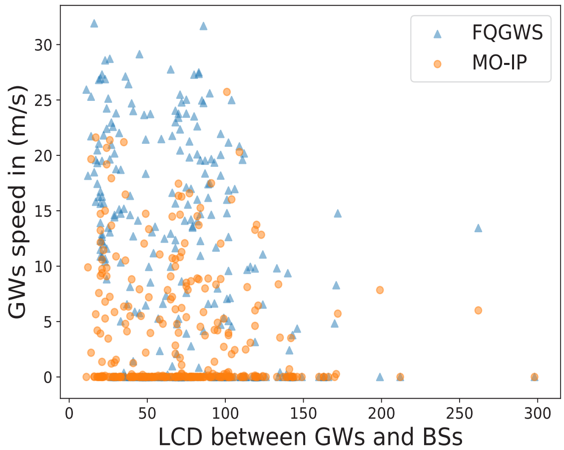

Fig. 3: connection lifetime between GWs and BSs.

The bandwidth constraint limits number of VNs per GW. We set the GW capacity number to 10. Figure 6 and 5 show that the RL-agent has better performance in increasing the number of connected VNs in comparison with FQGwS and DIS-based algorithms. Without bandwidth limitation constraint, RL-agent causes inequality and a wide variation in the distribution of VNs over the MGs, as shown in Figure 7. Figure 8 shows that the RL-agent is not affected by capacity constraint, and it has a higher efficiency in distributing VNs compared to other solutions.

Finally, the connection lifetime between VNs and the infrastructure is evaluated. The connection lifetime is the minimum of $(CON_{VN2GW}, CON_{GW2BS})$ where CON_{VN2GW} represents the connection lifetime between the VN and the GW and CON_{GW2BS} denotes the connection lifetime between the GWs and the infrastructure. In figure 9, the connection lifetime rate of the proposed algorithm (GWS-MORL) is higher than in case of other algorithms. It is also not affected by the limitation constraint of GWs capacity when the number of vehicles increases, unlike the other algorithms in which the connection lifetime rate decreases, as presented in figure 10.

V. CONCLUSION

In this paper, a new gateway selection algorithm based

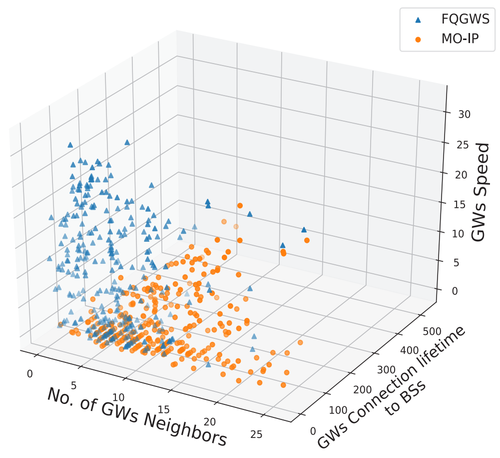

Fig. 4: Projection of metric 1, metric 2, and metric 3.

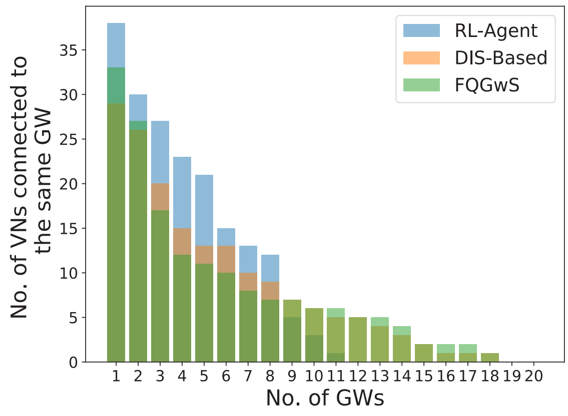

Fig. 7: VNs distribution among GWs.

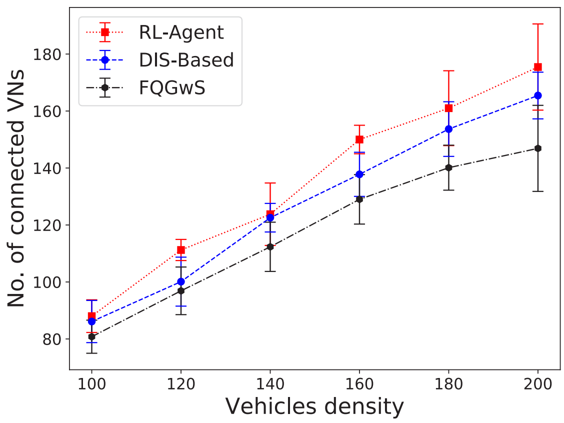

Fig. 5: Number of VNs connected to GWs.

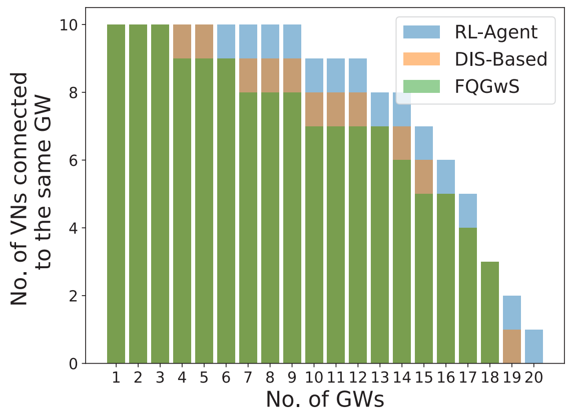

Fig. 8: VNs distribution among limited bandwidth GWs.

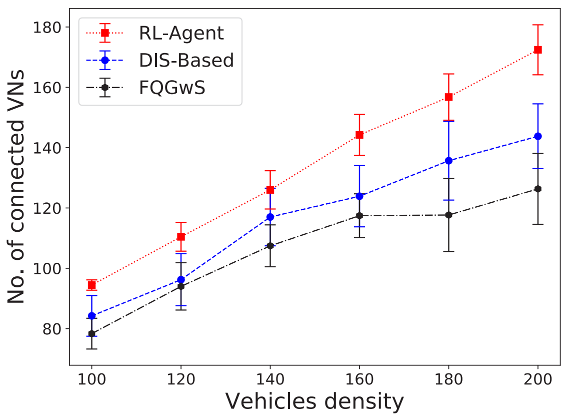

Fig. 6: Number of VNs connected to limited bandwidth GWs.

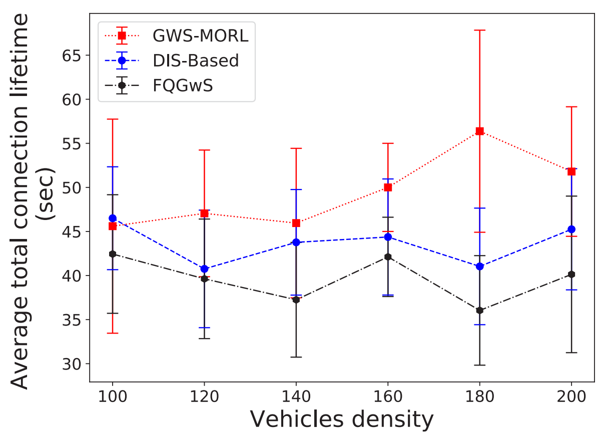

Fig. 9: VNs distribution among GWs.

on multi-objective integer programming and reinforcement learning is presented. The proposed system is a central algorithm assisted by vehicular cloud. System architecture consists of two phases. In the first phase, multi-objective Integer programming is used to elect the best gateways depending on their speed, direction, and proximity to the base stations. The reinforcement learning technique is employed in the second phase to allocate one of elected gateways for each vehicle in need of the infrastructure services. Two agents are created based on the objectives' preferences. Compared with the existing mobile gateway selection algorithms, the simulation results show that the proposed approach is effective in terms of increasing the number of connected vehicles, distributing the vehicular nodes among gateways, and increasing the connection lifetime between the source vehicles and the infrastructure.

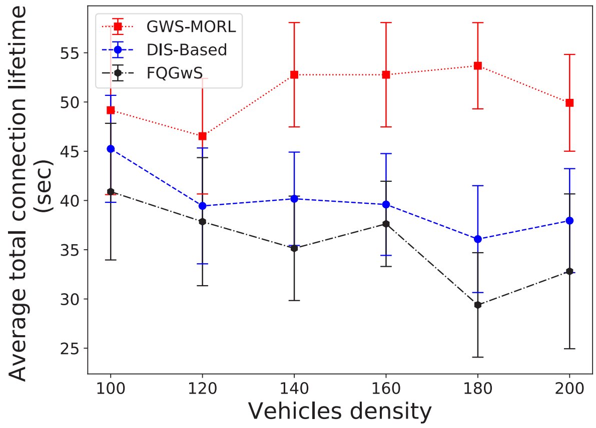

Fig. 10: VNs distribution among limited bandwidth GWs.

REFERENCES

[1] M. Lee and T. Atkison, "Vanet applications: Past, present, and future," *Vehicular Communications*, vol. 28, p. 100310, 2021, DOI: 10.1016/j.vehcom.2020.100310.

[2] J. Jeong, Y. Shen, T. Oh, S. Céspedes, N. Benamar, M. Wetterwald, and J. Härri, "A comprehensive survey on vehicular networks for smart roads: A focus on ip-based approaches," *Vehicular Communications*, vol. 29, p. 100334, 2021, DOI: 10.1016/j.vehcom.2021.100334.

[3] H. Alabbas and Á. Huszák, "Camvc: Stable clustering algorithm for efficient multi-hop vehicular communication on highways," in *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2020, pp. 149–152, DOI: 10.1109/TSP49548.2020.9163488.

[4] F. Cunha, L. Villas, A. Boukerche, G. Maia, A. Viana, R. A. Mini, and A. A. Loureiro, "Data communication in vanets: Protocols, applications and challenges," *Ad Hoc Networks*, vol. 44, pp. 90–103, 2016, DOI: 10.1016/j.adhoc.2016.02.017.

[5] H. Hejazi and L. Bokor, "A survey on the use-cases and deployment efforts toward converged internet of things (iot) and vehicle-to-everything (v2x) environments," *Acta Technica Jaurinensis*, 2021, DOI: 10.14513/actatechjaur.00627.

[6] D. A. Mahmood and G. Horváth, "Analysis of the message propagation speed in vanet with disconnected rsus," *Mathematics*, vol. 8, no. 5, p. 782, 2020, DOI: 10.3390/math8050782.

[7] S. Bhoi, P. Khilar, M. Singh, R. Sahoo, and R. Swain, "A routing protocol for urban vehicular ad hoc networks to support non-safety applications," *Digital Communications and Networks*, vol. 4, no. 3, pp. 189–199, 2018, DOI: 10.1016/j.dcan.2017.08.003.

[8] M. Alawi, E. Sundararajan, R. Alsaqour, and M. Ismail, "Gateway selection techniques in heterogeneous vehicular network: Review and challenges," in *2017 6th International Conference on Electrical Engineering and Informatics (ICEEI)*. IEEE, 2017, pp. 1–6, DOI: 10.1109/ICEEI.2017.8312425.

[9] H. Shahwani, S. A. Shah, M. Ashraf, M. Akram, J. P. Jeong, and J. Shin, "A comprehensive survey on data dissemination in vehicular ad hoc networks," *Vehicular Communications*, p. 100420, 2021, DOI: 10.1016/j.vehcom.2021.100420.

[10] Y.-W. Lin, J.-M. Shen, and H.-C. Weng, "Gateway discovery in vanet cloud," in *2011 IEEE international conference on high performance computing and communications*. IEEE, 2011, pp. 951–954, DOI: 10.1109/HPCC.2011.138.

[11] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, "Rethinking vehicular communications: Merging vanet with cloud computing," in *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings*. IEEE, 2012, pp. 606–609, DOI: 10.1109/CloudCom.2012.6427481.

[12] S. Bitam, A. Mellouk, and S. Zeadally, "Vanet-cloud: a generic cloud computing model for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 22, no. 1, pp. 96–102, 2015, DOI: 10.1109/MWC.2015.7054724.

[13] H. Alabbas and Á. Huszák, "Reinforcement learning based gateway selection in vanets," *International journal of electrical and computer engineering systems*, vol. 13, no. 3, pp. 195–202, 2022.

[14] A. Tolba, "Content accessibility preference approach for improving service optimality in internet of vehicles," *Computer Networks*, vol. 152, pp. 78–86, 2019, DOI: 10.1016/j.comnet.2019.01.038.

[15] G. El Mouna Zhioua, N. Tabbane, H. Labiod, and S. Tabbane, "A fuzzy multi-metric qos-balancing gateway selection algorithm in a clustered vanet to lte advanced hybrid cellular network," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 2, pp. 804–817, 2014, DOI: 10.1109/TVT.2014.2323693.

[16] B. Sharef, R. Alsaqour, M. Alawi, M. Abdelhaq, and E. Sundararajan, "Robust and trust dynamic mobile gateway selection in heterogeneous vanet-umts network," *Vehicular communications*, vol. 12, pp. 75–87, 2018, DOI: 10.1016/j.vehcom.2018.02.002.

[17] D. Abada, A. Massaq, A. Boulouz, and M. B. Salah, "An adaptive vehicular relay and gateway selection scheme for connecting vanets to internet," *Emerging Technologies for Connected Internet of Vehicles and Intelligent Transportation System Networks: Emerging Technologies for Connected and Smart Vehicles*, vol. 242, p. 149, 2019.

[18] S. Retal and A. Idrissi, "A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks," *Computers & Electrical Engineering*, vol. 73, pp. 289–303, 2019.

[19] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "*Proximal policy optimization algorithms,*" arXiv preprint arXiv:1707.06347, 2017.

[20] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: https://elib.dlr.de/124092/

[21] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org , 2017.

[22] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual," http://www.gurobi.com, 2021.

[23] A. Raffin, A. Hill, M. Ernestus, A. Gleave, A. Kanervisto, and N. Dormann, "Stable baselines3," https://github.com/DLR-RM/stable-baselines3, 2019.

[24] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. *MIT press*, 2018.

[25] H. Dong, H. Dong, Z. Ding, S. Zhang, and Chang, *Deep Reinforcement Learning*. Springer, 2020.

[26] C. Liu, X. Xu, and D. Hu, "Multiobjective reinforcement learning: A comprehensive overview," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 3, pp. 385–398, 2014, DOI: 10.1109/TSMC.2014.2358639.

Hasanain Alabbas received his MSc. in Computer Engineering from Al-Nahrain University, Iraq in 2014 and received BSc. from University of Technology, Iraq in 2007. He is currently a PhD student in the Department of Networked System and Services at Budapest University of Technology and Economics, Hungary. His current research interests are in the area of vehicular networks, routing protocols and clustering algorithms.

Árpád Huszák received his M.Sc. degree in 2003 as an Electrical Engineer from the Budapest University of Technology and Economics (BUTE) at the Department of Telecommunications (Dept. of Networked Systems and Services since 2013). He received Ph.D. degree in electrical and computer engineering in 2010. Currently he is with the Department of Networked Systems and Services as assistant professor, but he also joined the Mobile Innovation Center Hungary (MIK). He is member of the IEEE and HTE. Recently, he has been involved in many European (FP6-IST, FP7-ICT, EIT and Celtic) and domestic projects. His research interests focus on network protocols, mobile computing and machine learning.