Ensemble SARSA and LSTM for User-centric Handover Decisions in 5G Vehicular Networks

Mubashir Murshed, Glaucio H.S. Carvalho, and Robson E. De Grande

***Abstract*—5G and vehicular networks have enabled Intelligent Transportation Systems (ITS) with better safety and infotainment services where connected vehicles are critical components for data sharing. However, a stable connection is mandatory to transmit data successfully across the network. The 5G technology enhances bandwidth, stability, and reliability but suffers from low communication ranges, which results in frequent and unnecessary handovers and connection drops. In this paper, we introduce a user-centric approach, Factor-distinct SARSA Reinforcement Learning (FD-SRL), which combines a time series data-oriented model LSTM and adaptive method SARSA Reinforcement Learning for Virtual Cell (VC) and handover (HO) management. Our proposed approach maintains stable connections by reducing the number of HOs, given the fast-paced changes due to mobility, network load, and communication conditions. Realistic simulations demonstrated that FD-SRL reduced the number of HOs and the average cumulative HO time, showing potential improvements in connection stability for 5G-based ITS.**

***Index Terms*—5G, SARSA, LSTM, Virtual Cell, Handover, Mobility, Ultra-dense Networks**

I. Introduction

Intelligent Transportation Systems (ITS) in urban and highway scenarios aim to improve vehicular safety and mobility, as well as to provide comfort services [1]. Vehicular Networks (VN) enable vehicles to communicate with each other, infrastructures, and devices, sharing data and information produced in vehicular On Board Units (OBU) for safety, traffic management, and entertainment [2] [3].

The mobility of vehicles on road segments is highly dynamic [4]. Connected vehicles rely on wireless communication, requiring switching between transmission points to remain connected by performing handover (HO) [5]. However, vehicles' highly dynamic behaviour often results in lost connections, negatively disrupting the delivery of services and even compromising safety. Cellular management's delayed decision-making often causes connection dropouts; in metropolitan settings, network ultra-density is fairly prevalent.

The 5G mmWave (millimetre wave) frequencies provide more bandwidth and higher data transfer rates than traditional networks, including benefits, such as high bandwidth, ultra-low latency, enhanced security, and increased energy consumption [6] [7]. However, the high frequency also leads to more significant signal loss due to absorption and scattering, limiting the communication range and leading to a smaller coverage area than traditional networks. The limited coverage area of 5G networks operating at mmWave frequencies is an open issue where approaches attempt to mitigate it, such as merging multiple cells [8]. The constrained communication range causes user nodes to perform frequent HO to remain connected to the cellular towers. Frequent and unnecessary HO affect connection stability for dynamic high mobility vehicles [5] [9]. Efficient decision-making for HO is essential for enabling reliable and stable vehicular service and applications.

A user-centric virtual cell (VC) strategy helps to deal with connection stability problems due to the low coverage range of 5G networks [9] [8]. A user-centric network performs network and decision-making operations on the user's side rather than the base station to deal with the high mobility of vehicles and ultra-density, reducing decision-making costs and complexity. By the definition of the user-centric VC, a vehicle remains connected to multiple transmission points or cellular towers and virtually forms a cell to get service [9] [8]. Redundant lightweight connections in a VC consume negligible bandwidth and support this user-centric strategy to help stabilize communication even under adverse network conditions by ensuring the benefits of 5G networks. Additionally, by leveraging a time series and adaptive learning approaches, the design of HO management mechanisms for VCs can lead to intelligent systems that effectively and efficiently adjust to user behaviour and network conditions in a dynamic manner based on historical data. For these systems, the time series models lead to stable network conditions and enhanced user experience, while adaptive learning approaches promote real-time decision-making HO process. Moreover, time series data-oriented learning can provide a highly customized and efficient user experience while maintaining stable network connections; adaptive learning offers more precise decision-making in a real-time manner.

Therefore, our proposed approach is designed to be highly adaptive and robust, offering broad applicability across various network settings. Unlike fixed-trained models, it learns in real-time from the environment, enabling it to effectively manage high mobility and ultra-density in 5G networks. Figure 1 shows an example of VC where the solid black line denotes a VC with multiple cellular towers for a vehicle.

Many works have dealt with connection stability. Several techniques, such as HetNet, SDN, Fog, Hybrid, and VC, contribute to reducing HOs and better connection stability [5] [10] [9] [8] [11]. However, there is still a need for optimization for ensuring stable connections in 5G networks, particularly exploring user-centric networks and VC technologies. Previous works have handled HOs in 5G with a probabilistic estimation-based approach or have yet to consider historical and adaptive approaches together [11]. Combining an adaptive approach and time series data-oriented learning can more precisely ensure connection stability.

The main contribution of our work is successfully introduc-

Mubashir Murshed and Robson E. De Grande are with the Department of Computer Science at Brock University, St. Catharines, ON, Canada.

Glaucio H.S. Carvalho is with the Department of Computer Science and Department of Engineering at Brock University, St. Catharines, ON, Canada. E-mails: {mmurshed, gdecarvalho, rdegrande}@brocku.ca.

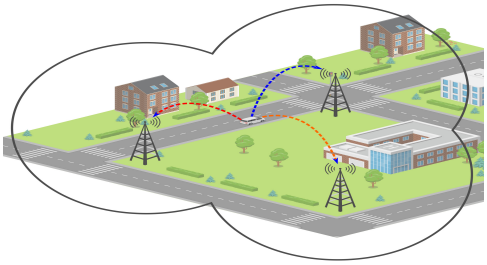


Fig. 1. Virtual cell scenario.

ing adaptability to the communication management of vehicles in dynamic urban ultra-dense vehicular networks:

- We presented an adaptive approach based on State-Action-Reward-State-Action (SARSA) Reinforcement Learning for VC management and HO decision-making.
- We incorporated a time series data-oriented Bidirectional Long Short-Term Memory (BLSTM) approach to better cope with the long- and short-term dependencies. It allowed more stable connections by reducing the number of HO and HO times.
- We conducted performance analyses with different 5G tower deployments, vehicular densities, and dynamic vehicular mobility in realistic, real-time simulations. The results demonstrated the efficiency of the proposed mechanism in reducing the HO management overhead.
- We compared our work with established SOTA approaches and showed that our proposed approach performs better in connection stability.

The rest of this paper is organized as follows. Section II presents related works for connection stability problems. Section III describes the system model. Section IV contains virtual cell management using time series data-oriented LSTM. Section V presents our proposed HO decision-making using adaptive SARSA RL. Section VI discusses the performance analysis. Finally, Section VII concludes by summarizing and providing future work directions.

II. Related Works

Connection stability is a concern in vehicular networks where efficient HO management is a prominent solution [10]. Reducing the number of HOs, decreasing HO times, and managing communication cells can render more stable connections. SDN, HetNet, Fog, and hybrid-based approaches are explored to address HO management efficiency [10]. To the best of our knowledge, not all real-world 5G V2X scenarios are holistically and thoroughly addressed (high mobility and high density), whereas most works were base-station-centric.

A. Heterogeneous Network

HetNets comprise integrating many wireless communication technologies, enabling more connection opportunities and leading to stable communication. A HetNet strategy helped deal with HO for connection stability based on Markov Decision Process rewards in 5G networks to reduce frequent HOs [12]. It avoided excessive HOs while maintaining the stability of connections, but it ignored intolerable HOs compared to other benchmark schemes. In another approach, a dynamic fuzzy Q-Learning method attempted to support mobility management for small cell networks [13]. Even though the work aims at mobility, its analyses observed UE movement with an average speed of 10km/h - low mobility to match vehicular systems. Stochastic analysis is a common approach to handling mobility management where works [14] have observed parameters, such as HO cost and channel busy times, where fixed network measurement assumptions may lead to issues in dynamic scenarios.

B. Software Defined Network

Software-defined Networks (SDN) introduce flexible network management that allows remote control of rules and policies. An SDN-based MEC-enabled approach was devised for service-aware HO management in 5G vehicular networks [15], reducing the number of HOs for seamless coverage. Similarly, SDN has been used in a multi-level view for handover management to optimize the HO times of vehicular networks by separating its support into core and edges [16]. SDN supported Unmanned Aerial Vehicles (UAVs) as on-demand forwarders, where UE's acted as terminals [17], signaling overhead and enabling HO decisions.

C. Fog-based Network

The fog-based network uses macro-cell base stations and shifts network functions closer to end-users, reducing latency. A feed-forward neural network determined an optimal fog node where the distance between fog and vehicle and time were parameters [2]. However, the approach was limited to HO between fogs, did not consider heterogeneous networks, and assumed known vehicle trajectories. A two-tier Machine Learning-based scheme was introduced for HO management in intelligent vehicular networks [18]. This work predicted signal strength using a recurrent neural network model to estimate the receiving signal strength for a handover decision.

D. Virtual Cell

In base station-centric approaches, the network selection procedure could face latency issues as it needs to be served from a remote base station [19]. A user-centric approach reduces latency problems and can migrate computational tasks to the user end in highly congested networks. Moreover, recent works have turned to Virtual Cell (VC) technologies to solve 5G networks' ultra-density and limited coverage range [9]. VCs have served to constrain effective data dissemination for a group of vehicles called hotspots (HSs), maximizing the number of served HSs and minimizing the total power radiated in 5G vehicular networks [8]. Since the size of a VC matters, a method determined the optimal radius of a VC to maximize the system downlink capacity using distance and remote radio head density [20]. VC helped support a scheme that selects a static cluster of small cells with local mobility to anchor in high-density scenarios [21]. In the presence of high vehicular mobility, a VC management approach for V2X communications in 5G networks focused on low energy consumption and high reliability [22]. Also, a dynamic user-centric scheme allowed updating VCs through the mobile tracking of the vehicles in V2X communication [23].

A probabilistic approach has dealt with VC formation and update [24]. It adopted user-centric VC management to allow

TABLE I
SUMMARY OF PREVIOUS WORKS.

Work	HO	V2X	5G	UC	Conn	HM	Approach	Target	Process
[12]	✓		✓		✓	✓	HetNet	Reducing HO	Markov Decision Process
[13]	✓				✓		HetNet	Mobility Mng	Dynamic Fuzzy Q-Learning
[14]	✓		✓		✓	✓	HetNet	HO, Overhead	Probabilistic Analysis
[15]	✓				✓	✓	SDN	Mobility Mng	Multi-access Edge Comp
[17]	✓				✓	✓	SDN	Reducing HO	UAV, On-demand forwarding
[2]	✓				✓	✓	Fog	Min HO time	Feed Forward NN
[24]			✓	✓	✓	✓	VC	VC formation	Probabilistic Estimate
[8]		✓	✓	✓	✓		VC	Limit broadcasting	Probabilistic Estimate
[9]	✓	✓	✓	✓	✓	✓	VC	VC formation, HO	Probabilistic Estimate
[20]	✓		✓	✓	✓	✓	VC	Optimal VC radius	Scheduling Algorithm
[21]	✓		✓	✓	✓		VC	Reducing HO	Local Anchor Cell Control
[22]		✓	✓	✓	✓	✓	VC	Min broadcasting	Probabilistic
[18]	✓	✓	✓		✓	✓	ML	HO Decision	Stochastic Markov model
This	✓	✓	✓	✓	✓	✓	**VC**	**VC & HO mng**	**RL, LSTM, Policy-oriented**

HO: Handover; UC: User-centric; Connectivity: Conn; HM: High Mobility; VC: Virtual Cell; NN: Neural Network, Machine Learning: ML; Reinforcement Learning: RL; Historical Model Learning: HML

decentralized decision-making and lower network complexity. The VC paradigm helped to deal with handovers in 5G V2X networks, using similar probabilities and parameters to make decisions [9]. However, these probabilistic approaches cannot cope with highly dynamic vehicular environments. Our previous work used adaptive SARSA Reinforcement Learning to maintain stable connection [11]. This work shows a promising solution to reduce HO and HO time and ensure connection stability. However, time series data-oriented learning along with adaptive learning can ensure more connection stability.

E. Remarks

As summarized in Table I, most earlier works were base station-centric and often failed to deliver stable connections. In addition, there are problems with power consumption, network complexity, ultra-dense networks, failure in HO, and HO time. A user-centric strategy may more effectively address stable connections, which would address all issues. Although studies have addressed user-centric VC, these works are based on a straightforward probabilistic methodology that lacks adaptation to reliably match the real-time, fast-paced changes in vehicular environments.

III. SYSTEM MODEL

Both trace analysis and direct observation of the environment in real-time impact the understanding of vehicular mobility: routes and movement patterns. Depending on different road segments, time, weather, and road-side events, they are quite uncertain. Vehicle direction, speed, and value of signal measurement parameters vary due to the dynamic vehicular movement. Sometimes, only adaptive learning or a trace-based learning model may not be efficient for decision-making in a dynamic environment. Time series data-oriented learning for historical model-based learning is an effective approach for dynamic environments. It is also true that a predefined model is inefficient for predicting with an old dataset and requires high computational performance and time, which is unsuitable for vehicles running on the road. Considering all the challenges, we propose a user-centric approach FD-SRL consisting of lightweight time-series learning and adaptive learning, adjusting in real-time to effectively manage high mobility and ultra-density. FD-SRL is an improvement of CO-SRL [11], where we introduce adaptive learning along with time-series data-oriented learning to make it more efficient in terms of decision-making.

FD-SRL consists of two phases *(i)* time-series data-oriented Virtual Cell (VC) management and *(ii)* adaptive learning-based handover (HO) decision-making. In phase *(i)*, a virtual cell is formed using time series data-oriented Bidirectional LSTM (BLSTM). BLSTM extends LSTM by processing sequences bidirectionally, capturing both past and future context, typically implemented by stacking two LSTM layers and concatenating their outputs. HO decision is performed in phase *(ii)* using SARSA Reinforcement Learning (SRL). SRL is a Reinforcement Learning algorithm where an agent learns a policy by updating its Q-values based on the state, action, reward, next state, and next action tuples. Both phase *(i)*, VC management, and phase *(ii)*, HO decision-making, are user-centric. We use time series-oriented historical data for VC management because VC consists of multiple towers, and they are updated less frequently. Moreover, a predefined model based on its traversing path and historic signal values can help to find suitable cellular towers to provide services [25]. For this reason, we adopt time-series data-oriented learning for VC management. On the other hand, HO management mostly depends on a real-time environment, which efficiently deals with adaptive learning. By considering the dynamic behaviour, we propose a blend of model-based and model-free learning methods to enhance performance in dynamic scenarios. The output of phase *(i)* time-series data-oriented VC management is a set of cellular towers for a virtual cell, which is used as the input of phase *(ii)* adaptive learning-based handover management. The outcome of phase *(ii)* is the selection of a serving tower, i.e. intra-VC or inter-VC HO. The architecture of our proposed FD-SRL is given in Figure 2. The VC for a vehicle encompasses multiple cellular towers, and the VC is updated over time. The components of BLSTM and SARSA RL are illustrated by zooming out from the vehicle to highlight the user-centric operation of our proposed FD-SRL.

Figure 3 illustrates the overview of the system's decision flowchart of the FD-SRL approach. The flowchart depicts each module of FD-SRL as a rectangular box, showing the system's progression from phase *(i)* to phase *(ii)*. Each box details the breakdown of its respective modules. The exchanged beacon messages contain data that is directed to the time-series data-oriented VC management using BLSTM. Subsequently, adaptive learning-based HO decision-making is performed using SARSA RL.

The system under assumption considers a scenario with a set of vehicles $V = \{v_1, v_2, ..., v_i\}$, a set of cellular towers $TCT = \{tct_1, tct_2, ..., tct_j\}$, and a set of in-range towers $CT = \{ct_1, ct_2, ..., ct_j\}$, where $ct_j \in CT$ and $CT \subset TCT$. The set of in-range towers relates to a single vehicle v_i; when vehicle v_i comes to the communication range of cellular towers of TCT at time k, CT is updated with those in-range towers.

IV. TIMES SERIES-ORIENTED VIRTUAL CELL MANAGEMENT

In phase *(i)*, we use Bidirectional LSTM (BLSTM) for Virtual Cell (VC) management. BLSTM uses time series data to predict future value [26] [27]. BLSTM comprises a cell,

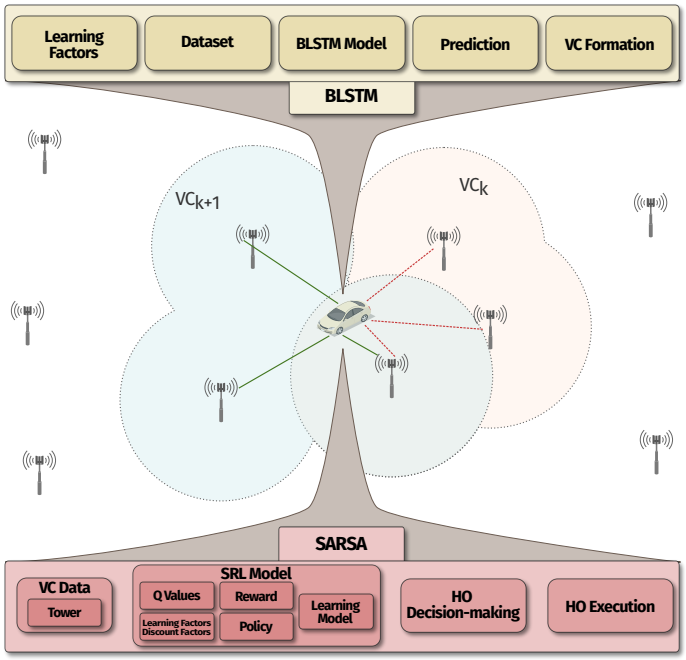


Fig. 2. Architecture of FD-SRL.

an input gate, an output gate, and a forget gate. The cell remembers values over arbitrary time intervals, and the three gates regulate the flow of information into and out of the cell. BLSTM has two sequence processing models, one taking the input in the forward direction and the other in the backward direction. Though BLSTM is not that kind of lightweight model, we designed the simple-lightweight structure of BLSTM and the training and testing data to run faster within the timeframe. The BLSTM module in Figure 3 outlines the stages of the process, including layers, model training, and VC building using the prepared dataset, leading in scalar value prediction and decision-making for VC management.

A. Learning Factors

We consider RSSI, SINR, RSRP, distance, tower load, and speed as the learning factors of FD-SRL. In our user-centric FD-SRL approach, we combine learning factors together to a scalar value denoted as ι^j. This scalar value ι^j is used as the feeding value of time-series data-oriented BLSTM. We use speed for determining the cell size, which is discussed in Section V-E of this section.

In Equation (1), we combine the weighted sum of all of the learning factors to calculate the scalar value ι^j. Here $\hat{w}_{rssi}$, $\hat{w}_{sinr}$, $\hat{w}_{rsrp}$, $\hat{w}_{dist}$ and $\hat{w}_{load}$ are weights of RSSI, SINR, RSRP, distance, and tower load respectively.

$$\begin{aligned}\iota^j = &(\chi^i_{rssi_{ct_j}} * \hat{w}_{rssi}) + (\chi^i_{sinr_{ct_j}} * \hat{w}_{sinr}) \\ &+ (\chi^i_{rsrp_{ct_j}} * \hat{w}_{rsrp}) + (\chi^i_{dist_{v_i,ct_j}} * \hat{w}_{dist_{v_i,ct_j}}) \\ &+ (\chi^i_{load_j} * \hat{w}_{load})\end{aligned} \tag{1}$$

B. Execution of BLSTM

We design the BLSTM model according to the context of dynamic vehicular networks. A BLSTM processes the data forward and backward with two separate hidden layers. The

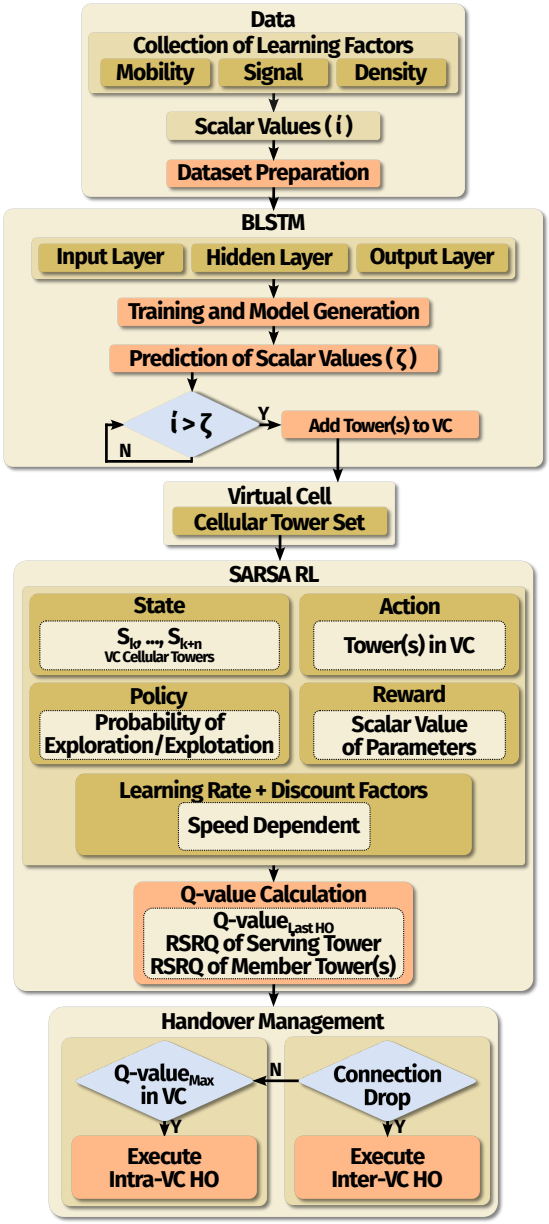


Fig. 3. Overview of system's decision-flowchart of FD-SRL.

input layer of BLSTM contains the input vector of scalar value ι^j at time k. The feeding values $\iota^j = \iota^j_1, \iota^j_2, ..., \iota^j_n$ are propagated to the hidden layer. The hidden layer updates its hidden layers status $h_1, h_2, ..., h_n$ at each time k. The output g_n is computed based on the input ι^j as an output vector sequence $g = g_1, g_2, ..., g_n$ [27]. The hidden layer function $\mathcal{H}$ of BLSTM is implemented using a logistic sigmoid function, input gate, forget gate, output gate, and cell activation vectors; all have the same size as the hidden vector h. This vector feeds forward to the same output layer. A BLSTM computes the forward hidden sequence $\overrightarrow{h}$, the backward hidden sequence $\overleftarrow{h}$ and the output sequence g by iterating the backward and forward layer and then updating the output layer with $\mathcal{W}$ terms (denote weight matrices), the b terms denote bias vectors by following the Equations (2), (3) and (4).

$$\overrightarrow{h} = \mathcal{H}(\mathcal{W}_{\iota\overrightarrow{h}}\iota_k + \mathcal{W}_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{h}_{k+1} + b_{\overrightarrow{h}}) \tag{2}$$

$$\overleftarrow{h} = \mathcal{H}(\mathcal{W}_{\iota\overleftarrow{h}}\iota_k + \mathcal{W}_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{k+1} + b_{\overleftarrow{h}}) \tag{3}$$

$$g = \mathcal{W}_{\overrightarrow{h}\,g}\,\overrightarrow{h}_k + \mathcal{W}_{\overleftarrow{h}\,g}\,\overleftarrow{h}_k + b_g \tag{4}$$

We experiment on a single-layer BLSTM. The input layer is fully connected to the hidden layer, and the hidden layer is fully connected to the output layer. BLSTM blocks use the logistic sigmoid for the cell's input and output squashing functions. We modify the traditional BLSTM to make it lightweight and follow the requirements of network connectivity parameters. We use 50 neurons, activation function $ReLU$, optimizer $Adam$, and 100 epochs in the training and test process.

C. Dataset

The real-time captured data is used to determine the size of VC to ensure stable connectivity. To remain updated with the dynamic environment, we only store data or enrich datasets for a short period. In the high dynamic mobility scenario, a pre-defined model using an old dataset for a long period might lead to a wrong result. However, our proposed user-centric FD-SRL performs its training and testing operations in the OBU of each v_i frequently to be updated with time.

We assume that vehicle v_i receives a sequence of the scalar values of parameters ι^j from towers CT. We train our BLSTM model in an offline manner and use the saved model to predict a new scalar value sequence ζ^i. We define a period $k_1, k_2, ..., k'_t$ to perform training and testing of BLSTM. A window size $k' - k$ is defined for the prediction of ζ^i where the size of the input vector is $\iota^j_{k'}$.

Vehicle v_i generates and captures RSSI, SINR, RSRP, distance, and tower load and stores the data in its storage area by converting them to a scalar value ι^j. v_i maintains a timeframe $k' - k$ to store data in a dataset $\mathcal{D}^i$ for training purposes and testing purposes. In this timeframe, the stored scalar value reached the window size. The timeframe $k' - k$ is determined in such a way that is efficient for training and testing. The generated and captured data is divided into 70% and 30% for training data $\mathcal{DTN}^i$ and testing data $\mathcal{DT}^i$ purposes.

D. Prediction of Scalar Value

At each time interval $k' - k$, the BLSTM model $\mathcal{M}^i$ is generated with the training data $\mathcal{DTN}^i$ and testing data $\mathcal{DT}^i$. After generating $\mathcal{M}^i$, FD-SRL generates the predicted scalar value ζ^i for vc_i. This predicted scalar value ζ^i is used for the next time interval $k' - k$ for selecting towers of vc_i. If the scalar value ι^j for a tower ct_j is greater than the predicted scalar value ζ^i, that tower is selected for the vc_i of vehicle v_i. In every time interval $k' - k$, this process repeats to update VC.

Mean Square Error (MSE) is employed to evaluate the performance of BLSTM. MSE provides a quantitative measure of the model's prediction accuracy by measuring the average difference between the actual and predicted values. We use a predefined maximum error rate to identify the model's performance. The time interval $k' - k$ is selected by focusing on the error measurement report. In a time interval $k' - k$, $\mathcal{M}^i$ is generated. When the MSE of $\mathcal{M}^i$ is less than our predefined maximum error rate, $\mathcal{M}^i$ is considered for performing further operations. Whenever $\mathcal{M}^i$ MSE exceeds the predefined maximum error rate, $\mathcal{M}^i$ is discarded and trained again. by increasing the time interval $k' - k$. The previous $\mathcal{M}^i$ performs operations till new $\mathcal{M}^i$ has less MSE than the predefined maximum error rate.

ALGORITHM 1: Algorithm FD-SRL - Phase *(i)*

Data : $rssi_{ct_j}$, $sinr_{ct_j}$, $rsrp_{ct_j}$, (x, y), $dist$, $load_{ct_j}$
Result: vc_i, ζ^i, ct^{ϑ}_j

1 $\iota^j \leftarrow rssi_{ct_j}$, $sinr_{ct_j}$, $rsrp_{ct_j}$, (x, y), $dist$, $load_{ct_j}$;
2 **for** k_1 **to** k'_{t-1} **do**
3 $\mathcal{D}^i \leftarrow \iota^j$;
4 **if** $k == k'_{t+1}$ **then**
5 forget $\mathcal{D}^i$ at $(k' - k)_0$;
6 **if** $k == k'_{t-1}$ **then**
7 $\mathcal{DTN}^i$, $\mathcal{DT}^i \leftarrow$ split $\mathcal{D}^i$ into 70% and 30%;
8 **if** $k == k'_t$ **then**
9 $\mathcal{M}^i \leftarrow seqBiDirectional$;
10 $\zeta^i \leftarrow \mathcal{M}^i.predict()$;
11 **if** $\iota^j >= \zeta^i$ **then**
12 $vc_i \leftarrow ct_j$;
13 $ct^{\vartheta}_j \leftarrow AlHoMgmt(vc_i)$;

Algorithm 1 describes the phase *(i)* of VC management. Learning factors are stored and updated in a dataset as train and test data from line 2...7. Then, the BLSTM model is trained, and prediction is made in each time interval $k' - k$, which is mentioned in line 8...10. The scalar value is compared with the predicted scalar value to form VC in line 11...12. In the end, in line 13, adaptive HO management is called for handover decision-making.

V. Adaptive Handover-decision Making

We define the components of the proposed FD-SRL model as vehicular connectivity requirements to ensure network stability in dynamic and high-mobility vehicular networks. FD-SRL performs its phase *(ii)* operation each time k. In phase *(ii)* of FD-SRL, vehicle v_i decides for the HO management. v_i selects a suitable tower ct^{ϑ}_j for getting service among $ct_j \in vc_i$. FD-SRL takes the HO decision, i.e. intra-VC or inter-VC handover, to maintain stable connections using the SRL algorithm. The phase *(i)* of FD-SRL provides a set of towers for vc_i, which is used as the input of phase *(ii)*. When v_i registers in a network, vc_i is determined with a default value of ζ^i, suitable for 5G network connectivity. It chooses the state, action, and reward of the phase *(ii)* according to $ct_j \in vc_i$ with max ι^j. With the progress in time, the FD-SRL measures the Q-value of FD-SRL by applying an action in a state with its corresponding reward. The SARSA RL module stems from phase *(i)* of the VC management module in FD-SRL as shown in Figure 3. It illustrates the workflow of adaptive SARSA RL components—state, action, reward, and policy—coupled with Q-value calculation, ultimately guiding HO decision-making.

We consider Reference Signal Received Quality (RSRQ) to be the Q-value of FD-SRL. RSRQ is calculated using RSRP and RSSI as described in CO-SRL [11]. The higher the RSRQ is, the better the signal. Thus, the higher Q-value indicates that a cellular tower is better for connection stability. After learning for a certain time during the initialization, FD-SRL starts making HO decisions using the adaptive SRL algorithm.

A. State-Action

We define state-action according to the serving tower and available actions in vc_i. Every vehicle v_i calculates its state and actions. The state-action is independent of each v_i in our proposed user-centric FD-SRL.

1) State: The state s_k is the cellular tower that serves v_i at time k. It resembles the current serving tower ct_j^{ϑ}. The state s_{k+1} can be any cellular tower with max ι^j at time $k+1$ from vc_i.

2) Action: There are several possible actions $a_k \in \{a_{k_1}, a_{k_2}, ..., a_{k_m}\}$ in a state s_k. The number of available actions is the same as the tower in vc_i. Each v_i has $|vc_i|$ number of available actions. The possible available migrations from the serving cellular tower to available cellular towers in vc_i at time k are defined as action $a_{k,p}$. In a state s_k, available actions are $a_{k,p} ct_j^{\vartheta} \rightarrow ct_j \in vc_i$. After taking an action $a_{k,m}$, the state changes to s_{k+1}. Thus, the serving tower ct_j^{ϑ} changes as the cellular tower of state s_{k+1}.

B. Reward

In SRL, the reward r_{k+1} is obtained by taking action a_k of transition s_k to s_{k+1}. Reward impacts on selecting the best action. The scalar combination of $\chi^i_{rssi_{ct_j}}$, $\chi^i_{sinr_{ct_j}}$, $\chi^i_{rsrp_{ct_j}}$, $\chi^i_{dist(v_i,ct_j)}$, $\chi^i_{load_{ct_j}}$ and $\chi^i_{speed_i}$ of an action a_k calculated as the reward $\mathcal{R}_{k+1}$ using Equation (5). We calculated their sum by weighted multiplication while computing the reward.

$$\begin{aligned}\mathcal{R} = &(\chi^i_{rssi_{ct_j}} * \hat{w}_{rssi}) + (\chi^i_{sinr_{ct_j}} * \hat{w}_{sinr}) \\ &+ (\chi^i_{rsrp_{ct_j}} * \hat{w}_{rsrp}) + (\chi^i_{dist_{v_i,ct_j}} * \hat{w}_{dist_{v_i,ct_j}}) \\ &+ (\chi^i_{load_j} * \hat{w}_{load}) + (\chi^i_{speed_i} * \hat{w}_{speed})\end{aligned} \tag{5}$$

C. Policy Function

FD-SRL uses the well-known ϵ-greedy policy of RL [28]. It handles trade-offs between exploration and exploitation.

At time k, ϵ_k is calculated from the predicted scalar value ζ^i of a time interval among the towers in vc_i [11]. Initially, an action is taken randomly. So, at time k, the probability of exploration is ϵ_k. After iterations, the action that has the best reward is selected with the exploitation probability $(1 - \epsilon_k)$.

D. Execution of SRL

FD-SRL calculates the Q-value for every tower in vc_i. The number of Q-values matches the number of available towers in vc_i. The reduced number of towers within a VC for vehicle vc_i ensures that SRL only consumes a little computation time and resources. We define the update of Q-value calculation as in Equation (6).

$$Q(s_k, a_k) \leftarrow Q(s_k, a_k)^{\vartheta} + \alpha[\mathcal{R}_{k+1} + \gamma rsrq^{mbr}_{ct_j} - rsrq^{srv}_{ct_j}] \tag{6}$$

Assume that at time k, an intra-VC handover has occurred, which is the switching of the serving cellular tower. At that period, the updated Q-value of the newly determined served cellular tower is $Q(s_k, a_k)^{\vartheta}$. Let the Q-value of a cellular tower in vc_i be represented as $Q^i_j(s_{k+1}, a_{k+1})$. It is the same as the Q-value of the next state. Thus, the Q-value of all available towers $ct_j \in vc_i$ is defined as $Q^i_1(s_{k+1}, a_{k+1}) = rsrq^{mbr_1}_{ct_1}, Q^i_2(s_{k+1}, a_{k+1}) = rsrq^{mbr_2}_{ct_2}, ..., Q^i_j(s_{k+1}, a_{k+1}) = rsrq^{mbr}_{ct_j}$.

ALGORITHM 2: Algorithm FD-SRL - Phase *(ii)*

Data : vc_i, $rssi_{ct_j}$, $sinr_{ct_j}$, $rsrp_{ct_j}$, $rsrq_{ct_j}$, (x, y), $dist$, $load_j$, $speed_i$
Result: ct_j^{ϑ}

1 $InitializeCondition$;
2 **while** $vc_i \neq \emptyset$ **do**
3 $a_k \leftarrow vc_i$;
4 $\mathcal{R}_{k+1} \leftarrow rssi_{ct_j}, sinr_{ct_j}, rsrp_{ct_j}, dist, load_{ct_j}, speed_i$;
5 $\epsilon_k \leftarrow computePolicy()$;
6 *set* α, γ; //depending on $speed_i$
7 $Q(s_k, a_k) \leftarrow Q(s_k, a_k)^{\vartheta} + \alpha[\mathcal{R}_{k+1} + \gamma rsrq^{mbr}_{ct_j} - rsrq^{srv}_{ct_j}]$;
8 **if** *(*$\arg\max_{ct_j}(Q(s_k, a_k))$ **then**
9 execute switching serving tower to new ct_j^{ϑ};
10 $Q(s_k, a_k)^{\vartheta} \leftarrow \arg\max_{ct_j}(Q(s_k, a_k))$;
11 **if** *(*$connectionDrop$ *in* vc_i*)* **then**
12 execute inter-VC HO to neighbouring cellular tower;
13 break;

In Equation (6), $rsrq^{mbr}_{ct_j}$ is the RSRQ of a cellular tower in vc_i, and $rsrq^{srv}_{ct_j}$ is the RSRQ of the serving tower in vc_i. These values are updated in every period k. Note that the $rsrq^{srv}_{ct_j}$ is the updated RSRQ at each time k, and $Q(s_k, a_k)^{\vartheta}$ is the RSRQ during the last switching of the serving tower or intra-VC HO. Whenever a Q-value gets to max than the previous for an action a_k, i.e. ct_j, is selected as the serving tower and the value of $Q(s_k, a_k)^{\vartheta}$ is updated.

There are $|vc_i|$ number of possible actions at a time k. For every available action a_k, v_i updates its respective Q-value considering state transition s_k to s_{k+1}, storing all updated Q-values in a local table. Thus, the number of computed Q-values equals the number of available cellular towers in vc_i. The a_k related to the maximum Q-value gained among the actions at time k within vc_i, i.e. ct_j^{ϑ}, is selected as the serving tower.

The phase *(ii)* of FD-SRL is discussed in Algorithm 2. The algorithm starts working from the calling of the function $AlHoMgmt(vc_i)$ in Algorithm 1. In line 3, the algorithm assigns VC towers to the SRL action. Then, reward, policy, learning rates, and discount factors of SARSA RL are assigned in line 4...6. The Q-value generation and intra-VC HO from line 7...10. Connection drop leads to inter-VC HO and redirects to the initial condition, which is followed as mentioned at the end.

E. Cell Size Selection

We use speed as another connectivity parameter to determine the size of the virtual cell. We have considered pico (small), micro (medium), and macro (large) size cells in our approach [9]. Speed has an impact on determining virtual cell size. We consider three categories of speeds: faster (121km/h to 180km/h), fast (61km/h to 120km/h), and medium (0km/h to 60km/h) [11].

The time interval $k' - k$ in the FD-SRL is calibrated based on the speed of the vehicles. Vehicles travelling at higher speeds have a larger time interval than those travelling at lower speeds. This calibration is performed to guarantee that sufficient data is collected to train the BLSTM model of FD-SRL. Moreover, when a vehicle moves faster, the macro cell is chosen with a slightly lower value of α and γ [11]. Accordingly, the fast- and medium-speed vehicles choose a

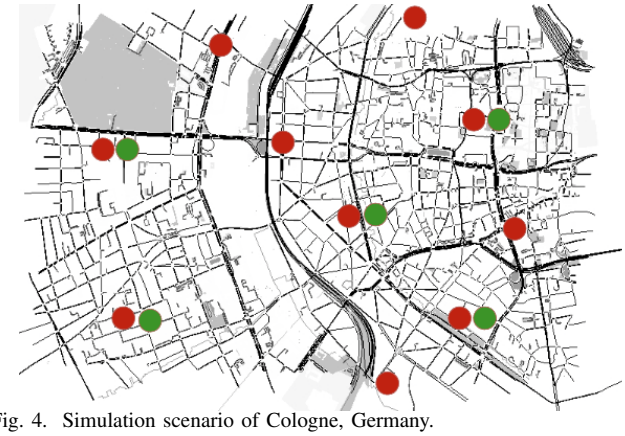


Fig. 4. Simulation scenario of Cologne, Germany.

microcell and pico cell with comparatively higher values of α and γ than faster speed as given in Table II.

VI. Performance Analysis and Results

We have conducted extensive simulated analyses of our proposed approaches utilizing VEINS, OMNet++, SUMO, and Simu5G. VEINS facilitates ITS simulation execution, allying vehicular mobility and network [29]. OMNet++ supports the simulation of networking and communication [30]. Simu5G supports simulating 5G and vehicular networks [31]. SUMO is a road traffic simulator [32].

A. Network Topology

We use the Cologne, Germany map in our simulations. It projects real-world traffic movement into traffic data. The selected region is a dense urban area with standard road layouts and highways, allowing variable mobility patterns across the region. We have conducted simulations in two different parameter settings with five and ten 5G gNodeB cellular towers. They are randomly placed on the simulation playground. We have positioned the cellular towers to mimic real-world conditions, such as (i) covering the majority of the region, (ii) having some areas outside the coverage of cellular towers, and (iii) having some areas where many cellular towers overlap. Figure 4 shows the simulated network map.

In our simulation scenario, each vehicle v_i has an independent traversing route from a starting point A to a destination D. Vehicles follow traffic rules, speed limits, priority, and patterns of a specific zone as our selected region. There are multiple routes for each v_i, source A and destination D can be different. In each seed of our batch run for the simulation, v_i follows different routes.

B. Simulation Parameters

Our proposed approaches are evaluated using a combination of parameter settings defined in Table II. The results are averages from 30+ runs with different seeds, with confidence intervals of 95%. Each simulation is 100s simulation-time long to execute operations. We evaluate our work in different vehicle densities [100..3100] and have conducted an analysis observing different speed ranges in intervals of 20km/h (min-max km/h): $0-20$ km/h to $160-180$ km/h. The different densities of vehicles and speeds impact the HO in terms of connectivity.

TABLE II
Simulation Parameters.

Parameter	Value Range
Simulation Area	$5 * 5$ km^2
Vehicle Density	$100-3100$
Vehicle Speed	0 km/h - 180 km/h
Num of Cellular Tower	5 and 10
Distribution of Cellular Tower	Random
Comm. Range of Cellular Tower	1000m
Simulation Time	100s
PHY Model	5G
Transmission Power (gNodeB)	46dbm
Transmission Power (Vehicle)	26dbm
Pico, Micro, Macro (α)	0.8, 0.5, 0.3
Pico, Micro, Macro (γ)	0.8, 0.5, 0.1

We have devised two separate parts of performance analyses. In our first part of the performance analysis, five 5G cellular towers (gNodeB) are placed randomly, and a finite number of vehicles traverse the map. In the second part of the performance analysis, ten 5G cellular towers (gNodeB) are employed with the same number of vehicles. The number of vehicles is the same in both analyses. The cellular towers are positioned in the same place for all compared approaches. We have defined the routes of vehicles and scenarios so that the second part is an extensive scenario that prompts more connection stability issues than the regular scenario of the first part. The green dots in Figure 4 represent the approximate location of the simulation with five 5G cellular towers (gNodeB). In comparison, the red dots indicate the approximate location of the simulation with ten 5G cellular towers (gNodeB). The coverage area for each gNodeB is defined as a constant radius of 1000m [33] [34], regardless of whether the environment is urban or highway. Our proposed approach features dynamic characteristics designed to adapt to varying conditions in each scenario, including communication range, road segment topology, 5G tower deployment, and vehicle mobility.

C. Performance Metrics

Our research reduces the overall number of HO and average cumulative HO duration to improve network stability [35] [9]. In our user-centric approach, the computing costs of training and decision-making are minimized due to the reduced scope of a virtual cell size [24] [21]. The effectiveness of our technique has been tested using a variety of performance parameters on vehicles with various densities and ranges of mobility.

1) Number of Intra-VC HO: The number of switching serving towers or intra-VC HO estimates the total number of switching of the serving tower within VC. As per the definition of VC, the vehicle already has information related to the switched tower in its VC, so there is no hard HO or connection loss, and connection stability is still maintained [9] [8]. The transition between the serving tower should be imperceptible to the user without facing connection drops while exchanging data.

Though the number of intra-VC HO does not directly affect connection stability, fewer HOs provide more stable service delivery. It is also true that a decent number of intra-VC of HO is necessary to have a better user experience.

2) Number of Inter-VC HO: The number of hard-HO or inter-VC HO represents the total number of HO performed outside VC. A vehicle faces this HO event while traversing on road segments. This type of HO requires the same amount of time as standard HO [9]. The lower the number, the better the network stability. A lower number of inter-VC HO shows better connection stability without interruptions in service delivery.

3) Average Cumulative HO Time for Intra-VC HO: The average cumulative HO time for intra-VC HO describes the time used for intra-VC HO. The cumulative HO time includes HO time inside VC and a latency including HO attachment-detachment time. This time is short when compared to inter-VC because all necessary information is already in the vehicle's OBU. The time taken to switch from one serving tower to another is minimal, so it does not affect the performance of user experience and safety service delivery. The average cumulative HO time is calculated by dividing the cumulative HO time by the total number of HO for intra-VC HO.

4) Average Cumulative HO Time for Inter-VC HO: The average cumulative HO time for inter-VC HO describes the time used for inter-VC HO. It is calculated by dividing the cumulative HO time by the total number of HO for inter-VC HO. In this case, the cumulative HO time consists of latency, including HO decision-making, attachment-detachment time, and a time for exchanging beacon messages. The latency for beacon messages and packet loss might be another reason. In contrast to intra-VC HO, the vehicle completes the HO operation in a longer time because it needs more information for inter-VC HO. A lower cumulative inter-VC HO time shows higher connection stability, facing less disruption in service and data delivery.

The metric average cumulative HO time for intra-VC and inter-VC HO shows the network stability by providing a lower average cumulative time required for intra-VC than inter-VC HO.

5) Percentage of Intra-VC: We calculate the percentage of intra-VC HO for different speeds. The percentages are obtained from the total intra-VC HO divided by the number of vehicles that reached a defined speed range. This metric describes the variation of HO occurrence in different speed ranges. The lower the percentage of intra-VC HO, the better the performance is. This metric indicates that a consistently low percentage of intra-VC HOs across all speed ranges signifies a stable and ideal handover system.

6) Percentage of Inter-VC HO: For different speeds, we determine the percentage of inter-VC HO. The percentages are calculated by dividing the number of vehicles that reached a specified speed range by the total number of inter-VC HO. This measure informs how inter-VC HO recurrence varies throughout speed ranges. Performance improves with a decreased inter-VC HO proportion. A low percentage of inter-VC HOs indicates high performance, suggesting that the HO system is more efficient and stable at maintaining connections as vehicles move through different speeds.

The smaller proportion of intra-VC HO than inter-VC HO shows that approaches perform well in terms of maintaining stable connections by executing HOs inside VC. Intra-VC and inter-VC HO percentages should retain stability to deliver better results regarding speed ranges.

7) Size of VC: The size of VC indicates the computational cost of network and HO management. The metric is the average VC size among a set of n vehicles randomly picked following a uniform distribution at the beginning of the simulation.

Growing the number of cellular towers in a virtual cell raises the number of connections that need to be managed by the network, increasing the computation required to maintain the network. Larger virtual cell sizes can lead to more costs and resources to manage communication. The system minimizes computational cost by selecting the most appropriate cellular towers, thus reducing their number. This metric describes the computational cost efficiency of a perfect handover system.

D. Compared Approaches

We have thoroughly examined numerous approaches in these analyses. The probabilistic approach, known as FiVH from the work in [9], has been implemented for the comparison. We have presented a performance study and comparison of the following approaches to demonstrate the improvement of our suggested methodologies.

- FiVH: Probabilistic approach of a previous work [9].
- M-FiVH: A modified probabilistic approach of FiVH [9].
- CO-SRL: Connectivity-oriented adaptive learning approach using SRL, a previous work [36].
- FD-SRL: Time series data-oriented adaptive learning approach using BLSTM and SRL.

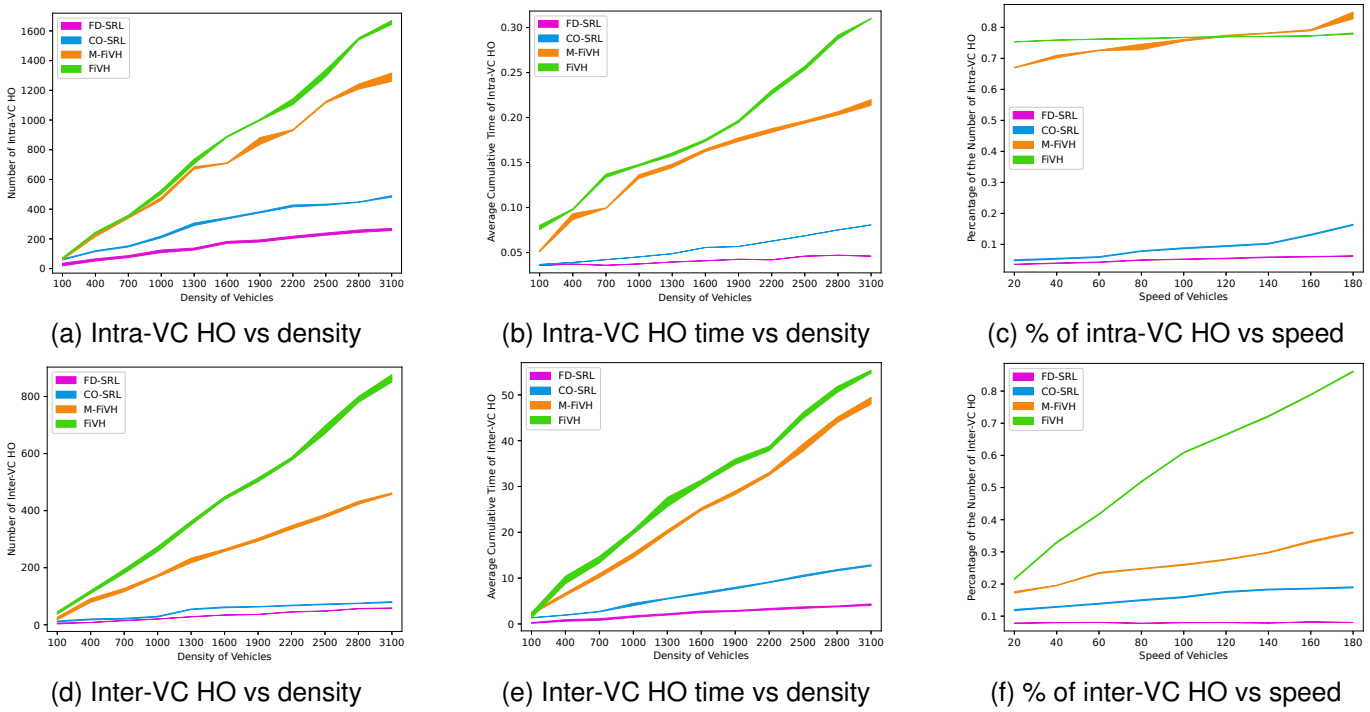
E. Results

We have conducted the performance analysis in two distinct parts. This progression has been aimed at validating the effectiveness of our proposed approach across different scenario setups. The first part compares FD-SRL, CO-SRL, M-FiVH, and FiVH using five 5G cellular towers (gNodeB). The second part compares FD-SRL and CO-SRL using ten 5G cellular towers (gNodeB). FD-SRL and CO-SRL are exclusively included in the second part of the comparison because they have demonstrated superior performance in the first part, particularly in scenarios with fewer cellular towers. This approach streamlines the comparison process by focusing on the best-performing algorithms from the outset.

1) Five-Towers Scenario: The performance of FD-SRL is compared with three previous works CO-SRL [36], FiVH [9], and M-FiVH. The results are averages of multiple runs, along with 95% confidence intervals, obtained from conducting over 30 experiments with different seeds. We evaluate our work in different vehicle densities $[100...3100]$.

It is important to mention that the thicknesses in the curves communicate the confidence intervals of the plotted performance metrics. Therefore, the thicker the curve is, the larger the confidence interval is. Adopting this modern data visualization technique enables the communication of the major trend and the confidence interval simultaneously in a concise and unequivocal manner.

Figure 5a represents the total number of intra-VC HO, which increases with the density of vehicles. The number of intra-VC HO for FD-SRL stays in the range $[15...250]$ while CO-SRL, M-FiVH, and FiVH are in the respective



(a) Intra-VC HO vs density (b) Intra-VC HO time vs density (c) % of intra-VC HO vs speed

(d) Inter-VC HO vs density (e) Inter-VC HO time vs density (f) % of inter-VC HO vs speed

Fig. 5. Handover management performance analysis in five-towers scenario.

ranges [50...500], [65...1300] and [76...1700], respectively. The number of intra-VC HOs follows a linear upward trend for all methods. However, the FD-SRL and CO-SRL display smaller slopes than the baseline methods, which shows that they are more efficient in reducing the amount of intra-VC. Notably, FD-SRL outperforms CO-SRL as the number of intra-VC HO is lower, highlighting its superior performance.

Figure 5d shows the total number of inter-VC HO increases linearly with the densification of the vehicle population. Unlike the baseline approaches, which present steep slopes, the curves for FD-SRL and CO-SRL present gentle inclines. This result demonstrates that the proposed methods deliver connection stability for sparse and dense scenarios. Again, FD-SRL outperforms CO-SRL, which is indicated by its smaller slope. When it comes to the number of inter-VC HOs, it turns out that for FD-SRL, CO-SRL, M-FiVH, and FiVH, it stays in the respective ranges [5...60], [10...82], [40...460], and [60...850].

Figure 5b and 5e describes the average cumulative time required for intra-VC and inter-VC HO. Intra-VC HO requires at most 0.05ms, 0.08ms, 0.24ms, and 0.31ms for FD-SRL, CO-SRL, M-FiVH, and FiVH, respectively. The average cumulative time increases sharply for M-FiVH and FiVH with the increasing number of densities and slightly increases for FD-SRL and CO-SRL. FD-SRL exhibits a lower average cumulative time for intra-VC HO compared to others. The average cumulative time of inter-VC HO for high-density 3100 vehicles is 4ms, 12ms, 47ms, and 60ms for FD-SRL, CO-SRL, M-FiVH and FiVH, respectively. For inter-VC HO, FD-SRL requires less time than CO-SRL, M-FiVH, and FiVH. Moreover, Figure 5b and 5e also show that intra-VC HO requires less time than inter-VC HO.

It should be noted that the average cumulative time for intra-VC and inter-VC HO increases with the densification of the vehicle population. The reason behind this trend is likely the corresponding increase in the number of handover candidates that overloads the network, causing latency inflation and ultimately extending the completion of the handover process. Furthermore, vehicle densification is intrinsically associated with interference in such a way that the denser the area is, the higher the interference is. This increase in interference also slows down the handover process. Notably, an increase in this metric is more critical for inter-VC HO than for intra-VC HO because the handover occurs outside the VC.

Moreover, M-FiVH showed better performance than FiVH. We can understand from Figures 5a, 5d, 5b and 5e that adding signal measurement RSSI as a parameter to FiVH improved the VC and HO management performance. From Figures 5a, 5d, 5b and 5e, we can infer that thanks to its real-time adaptive feature, CO-SRL achieve a more effective HO decision-making than M-FiVH and FiVH. Remarkably, the joint use of adaptive and time series learning renders FD-SRL even better performance.

As follows, we analyze the impact of speed in the HO process. In doing so, we consider different speed ranges in intervals of 20 km/h (min-max km/h): $0-20$ km/h to $160-180$ km/h. Figure 5c presents the percentage of intra-VC HO for different speed ranges. FD-SRL has an almost flat pattern concerning speed while CO-SRL has a slightly increasing pattern. It is expected because high-mobility vehicles need to perform more intra-VC HO to maintain stable connections. M-FiVH and FiVH have fluctuations in the percentage of the intra-VC HO, but they still have a massive difference from FD-SRL and CO-SRL. Since we are considering the maximum value of RSSI for the formation of VC in M-FiVH, the increases in speed lead to frequent intra-VC HO. It helps to reduce the number of inter-VC. On the other hand, FiVH remains stable as no signal parameter is considered, which affects the increasing number of inter-VC HO for FiVH rather

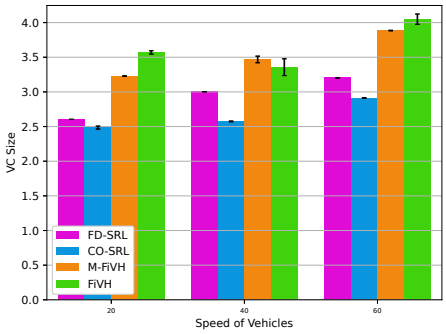


Fig. 6. The size of VC over speed.

than M-FiVH. Nonetheless, compared to the baselines, CO-SRL is significantly superior. For instance, regardless of the speed range, FD-SRL and CO-SRL maintain values below 0.07 and 0.16, respectively, whereas the ranges for M-FiVH and FiVH fluctuate between 0.67 and 0.85.

The percentage of the number of inter-VC HO is shown in Figure 5f. Similar to the analysis of intra-VC HO, the results illustrate that FD-SRL and CO-SRL are more robust to vehicle velocities, and the metric remains close to 0.08 and 0.2, respectively, while the baselines present a sharp increase in the percentage of the number of inter-VC HO. Notably, FD-SRL showcases superior performance compared to CO-SRL, particularly in scenarios where high mobility does not affect the number of hard HOs or connection drops.

The VC size serves as a measurement of the computational costs of the employed approaches. A hundred vehicles were monitored at three different speeds to determine the VC size. In Figure 6, FD-SRL and CO-SRL have a smaller VC size of $[2.6...3.1]$ and $[2.5...2.9]$, respectively, while M-FiVH and FiVH have a larger size of $[3.2...3.8]$ and $[3.5...4.1]$. FD-SRL and CO-SRL maintain lower computational costs than M-FiVH and FiVH because their VCs are reduced in size. However, FD-SRL incurs a slightly higher cost than CO-SRL. CO-SRL adds slightly fewer towers with policy-based learning, but FD-SRL adds more towers by combining time-series data-oriented learning with adaptive learning. According to the definition, FD-SRL requires slightly more computation cost than CO-SRL.

2) Ten-Towers Scenario: We have evaluated FD-SRL in comparison to our proposed CO-SRL, which is our previous contribution. We consider a new, more complex scenario in which more HOs may occur. The vehicle densities range from $[100...3100]$. The results are averages with 95% confidence intervals obtained from more than 30 seed-varying runs.

Figure 7a shows the total number of intra-VC HO against the increase in the densification of the vehicle population. As we can see, this metric grows for CO-SRL and FD-SRL almost linearly, with the latter outperforming the former. To quantify the difference, for the worst-case scenario, the total number of intra-VC HO is 36000 for FD-SRL against 57000 for CO-SRL. Since FD-SRL has a slightly smaller slope than the others, it works better in lowering the number of intra-VC HO than CO-SRL.

Figure 7d resembles the same upward trend reported in Figure 7a, but with smaller HO quantities. The increase for FD-SRL and CO-SRL is linear over the increase in vehicle density. Despite growing linearly, they continue to exist in smaller quantities. There are situations when HOs are necessary. While the number of inter-VC HOs for CO-SRL is $[1800...12000]$, for FD-SRL, it is $[200...8300]$ HOs.

Figures 7b and 7e summarize the average cumulative time needed for intra-VC and inter-VC HO. Intra-VC HO fluctuates between $[0.160...0.200]$ ms for FD-SRL. The average cumulative time increases slightly $[0.275...0.350]$ ms for CO-SRL as a function of the densification of the vehicle population. The average cumulative time of inter-VC HO is $[0.50...0.55]$ ms for CO-SRL, while FD-SRL requires less time, $[0.05...0.40]$ ms. Moreover, Figures 7b and 7e also show that intra-VC HO is faster than inter-VC HO. The average cumulative time for CO-SRL varies from our previous analysis. This result is likely caused by the modification in the settings of simulation parameters and environmental variations.

Like our previous results in Figure 5, Figures 7b and 7e show that the average cumulative time for intra-VC and inter-VC HO is highly affected by the concentration of vehicles. As we can see, CO-SRL is more vulnerable to densification in the vehicle population than the FD-SRL.

Figures 7c and 7f display the impact of speed on the number of intra- and inter-VC HOs. In agreement with the previous analysis in the Figure 5 counterparts, an increase in velocity provokes a rise in the number of HOs, regardless of the HO type. Thanks to the real-time adaptive feature, the proposed approaches keep this metric under control. Considering the worst-case scenario of the highest speed and CO-SRL, this metric peaks less than 0.9 and 0.45 for intra- and inter-VC HO, respectively. Since FD-SRL also leverages historical data, it gets the best results. To illustrate its benefits, considering the worst-case scenario, FD-SRL keeps the metric under 0.83 and 0.32 for intra- and inter-VC HOs.

Though FD-SRL performs better than CO-SRL, there are more fluctuations in FD-SRL than CO-SRL. The BLSTM model in FD-SRL is designed to learn long-term dependencies in data sequences, while the SRL algorithm in CO-SRL is designed to learn a policy for making decisions. Moreover, BLSTMs are sensitive to the amount of data; with the variation of captured data, the BLSTM model can perform differently and fluctuate more than SRL.

The VC size defines the computational cost of the implemented techniques. A hundred vehicles were observed at three different speed ranges to determine the VC size. Figure 8 shows that CO-SRL presented a smaller VC size $[3.2...4]$ than FD-SRL did $[4.2...5.1]$. The reason is that CO-SRL introduces slightly fewer towers through policy-based learning, while FD-SRL adds more towers by integrating time-series data-oriented learning with adaptive learning. Consequently, FD-SRL incurs slightly higher but acceptable computational costs compared to CO-SRL while delivering better performance in terms of reducing the number of HOs.

F. Remarks

Our proposed approaches, M-FiVH, CO-SRL, and FD-SRL, outperform FiVH, according to our analyses. With various tactics, the performance gradually becomes better. It is noteworthy that FD-SRL and CO-SRL produce the most remarkable outcomes. Regarding the number of handovers, average cumulative handover time, and percentages of handovers, FD-SRL outperforms CO-SRL, while CO-SRL has a lower

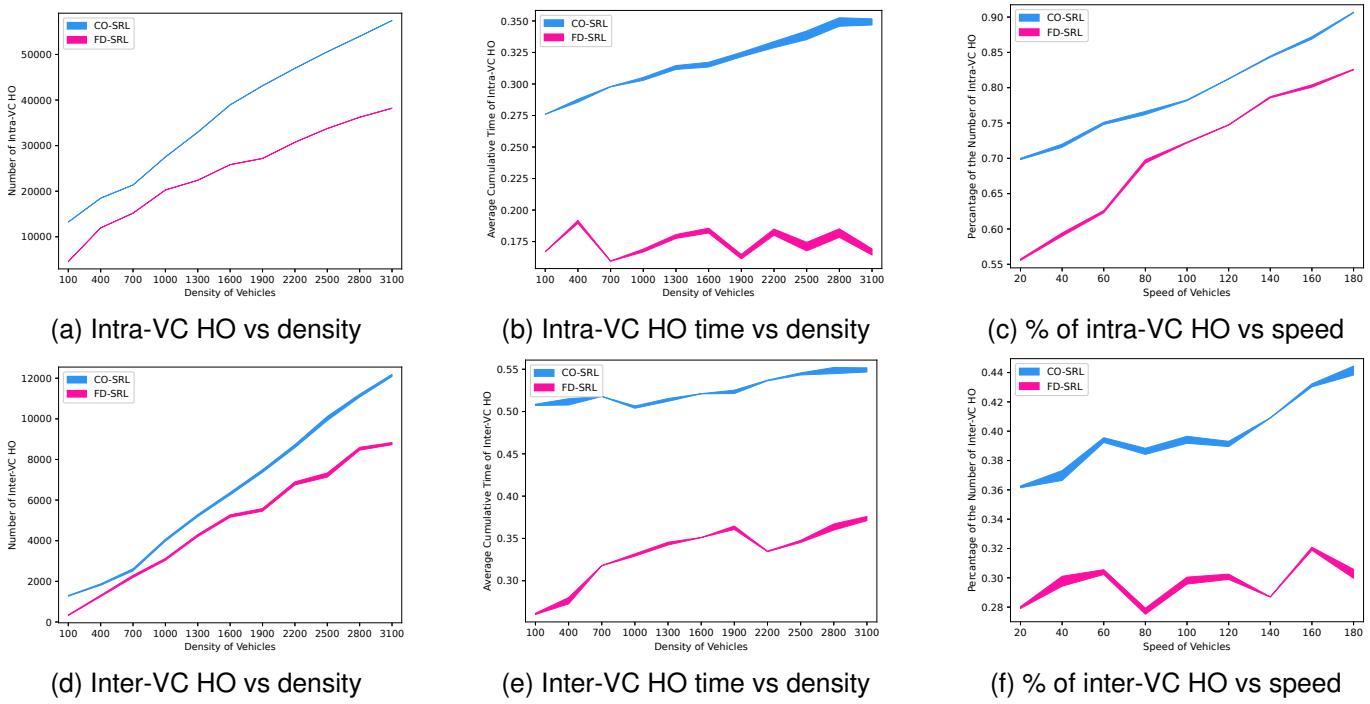


Fig. 7. Handover management performance analysis in ten-towers scenario.

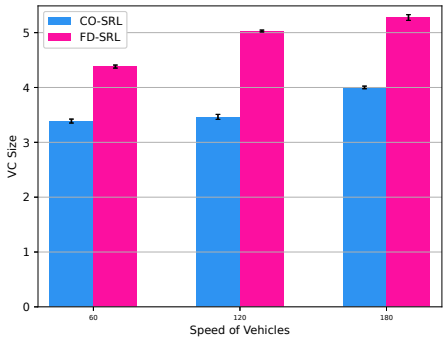


Fig. 8. The size of VC over speed.

computational cost than FD-SRL. Our analysis demonstrates that connection stability is maintained for high-mobility and ultra-density vehicular networks.

VII. Conclusion

In this paper, we have addressed the vehicular network stability problem in 5G vehicular networks for ultra-dense networks and high mobility of vehicles. Initial analysis of a user-centric HO management approach showed the importance of adding signal parameters to enhance algorithmic decision-making and enable dynamic adjustments (M-FiVH). This analysis enabled the development of the first contribution of this work, CO-SRL, an adaptive SARSA-based reinforcement learning (SRL) strategy for lowering the number of HO and average cumulative HO time, which adjusted mobility parameters and connectivity factors in real time for more stable connections. We then extended the model with a time-series data-oriented BLSTM learning. Time series data supported the management of virtual cells, while SRL supported handover management. This extension showed the significance of time in the management for supporting connection stability. Performance analyses showed that CO-SRL and FD-SRL ensured stable connectivity of networks while reducing HO overhead, outperforming FiVH and M-FiVH, VC-based approaches.

In future work, we will investigate different parameter settings, learning rates, and discount factors in terms of HO management efficiency and model complexity. Loss calculation will be studied more precisely in our future work. Moreover, we will incorporate a more comprehensive mobility model into the decision-making process, considering the complexity and cost of the system. We will explore HO management in the context of a multi-RAT scenario and investigate traffic steering to ensure effective HO decision-making and data flow.

Acknowledgments

This work was partially supported by NSERC Discovery, Canada and the Government of Canada.

References

[1] F. Cunha, G. Maia, H. S. Ramos, B. Perreira, C. Celes, A. Campolina, P. Rettore, D. Guidoni, F. Sumika, L. Villas, R. Mini, and A. Loureiro, *Emerging Wireless Communication and Network Technologies: Principle, Paradigm and Performance*. Springer, 2018, ch. Vehicular Networks to Intelligent Transportation Systems, pp. 297–315.

[2] Z. Mahmood, *Connected Vehicles in the Internet of Things: Concepts, Technologies and Frameworks for the IoV*. Cham: Springer Int. Publishing, 2020, ch. Connected Vehicles in the IoV: Concepts, Technologies and Architectures, pp. 3–18.

[3] T. S. Gomides, R. E. De Grande, A. M. de Souza, F. S. H. Souza, L. A. Villas, and D. L. Guidoni, "An adaptive and distributed traffic management system using vehicular ad-hoc networks," *Elsevier Computer Communications*, vol. 159, pp. 317–330, 2020.

[4] A. Boukerche and R. E. De Grande, "Vehicular cloud computing: Architectures, applications, and mobility," *Elsevier Computer Networks*, vol. 135, pp. 171–189, 2018.

[5] N. Aljeri and A. Boukerche, "Mobility management in 5g-enabled vehicular networks: Models, protocols, and classification," *ACM Computing Surveys*, vol. 53, no. 5, pp. 1–35, 2020.

[6] Y. Niu, Y. Li, D. Jin, L. Su, and A. V. Vasilakos, "A survey of millimeter wave communications (mmwave) for 5g: opportunities and challenges," *Springer Wireless Networks*, vol. 21, pp. 2657–2676, 2015.

[7] P. Lang, D. Tian, X. Duan, J. Zhou, Z. Sheng, and V. C. Leung, "Blockchain-based cooperative computation offloading and secure handover in vehicular edge computing networks," *IEEE Transactions on Intelligent Vehicles*, 2023.

[8] T. Sahin, M. Klugel, C. Zhou, and W. Kellerer, "Virtual cells for 5g v2x communications," *IEEE Communications Standards Magazine*, vol. 2, no. 1, pp. 22–28, 2018.

[9] C. R. Storck, E. E. de O. Lousada, G. G. de O. Silva, R. A. Mini, and F. Duarte-Figueiredo, "Fivh: A solution of inter-v-cell handover decision for connected vehicles in ultra-dense 5g networks," *Elsevier Vehicular Communications*, vol. 28, p. 100307, 2021.

[10] L. Tuyisenge, M. Ayaida, S. Tohme, and L.-E. Afilal, "Handover mechanisms in the internet of vehicles (iov): Survey, trends, challenges, and issues," in *Global Advancements in Connected and Intelligent Mobility: Emerging Research and Opportunities*. IGI Global, 2020, pp. 1–64.

[11] M. Murshed, G. H. Carvalho, and R. E. de Grande, "Adaptive user-centric virtual cell handover decision-making in 5g vehicular networks," in *ICC 2023-IEEE International Conference on Communications*. IEEE, 2023, pp. 3983–3988.

[12] S. Zang, W. Bao, P. L. Yeoh, B. Vucetic, and Y. Li, "Managing vertical handovers in millimeter wave heterogeneous networks," *IEEE Transactions on Communications*, vol. 67, no. 2, pp. 1629–1644, 2018.

[13] J. Wu, J. Liu, Z. Huang, and S. Zheng, "Dynamic fuzzy q-learning for handover parameters optimization in 5g multi-tier networks," in *Proc. of the IEEE Int. Conf. on Wireless Communications & Signal Processing (WCSP)*, 2015, pp. 1–5.

[14] M. F. Tuysuz and M. Ucan, "Energy-aware network/interface selection and handover application for android-based mobile devices," *Elsevier Computer Networks*, vol. 113, pp. 17–28, 2017.

[15] S. D. A. Shah, M. A. Gregory, S. Li, R. Fontes, and L. Hou, "Sdn-based service mobility management in mec-enabled 5g and beyond vehicular networks," *IEEE Internet of Things Journal*, vol. 9, no. 15, pp. 13 425–13 442, 2022.

[16] A. Gharsallah, F. Zarai, and M. Neji, "Exploring software-defined network (sdn) for seamless handovers in future vehicular networks: Mobility management approach," *Int. Journal of Smart Security Technologies (IJSST)*, vol. 9, no. 1, pp. 1–16, 2022.

[17] V. Sharma, F. Song, I. You, and H.-C. Chao, "Efficient management and fast handovers in software defined wireless networks using uavs," *IEEE Network*, vol. 31, no. 6, pp. 78–85, 2017.

[18] N. Aljeri and A. Boukerche, "A two-tier machine learning-based handover management scheme for intelligent vehicular networks," *Elsevier Ad Hoc Networks*, vol. 94, p. 101930, 2019.

[19] N. Mouawad, R. Naja, and S. Tohme, "User-centric vs network-centric vertical handover algorithms in 5g vehicular networks," in *Proc. of the Springer Int. Conf. on Internet of Vehicles*, 2018, pp. 46–59.

[20] Y. Zhang and Y. J. Zhang, "User-centric virtual cell design for cloud radio access networks," in *Proc. of the IEEE Int. Workshop on Signal Processing Advances in Wireless Communications*, 2014, pp. 249–253.

[21] N. Meng, H. Zhang, and B. Lin, "User-centric mobility management based on virtual cell in ultra-dense networks," in *Proc. of the IEEE/CIC Int. Conf. on Communications in China (ICCC)*, 2016, pp. 1–6.

[22] T. Şahin, M. Klügel, C. Zhou, and W. Kellerer, "Multi-user-centric virtual cell operation for v2x communications in 5g networks," in *Proc. of the IEEE Conf. on Standards for Communications and Networking (CSCN)*, 2017, pp. 84–90.

[23] H. Xiao, X. Zhang, A. T. Chronopoulos, Z. Zhang, H. Liu, and S. Ouyang, "Resource management for multi-user-centric v2x communication in dynamic virtual-cell-based ultra-dense networks," *IEEE Transaction on Communications*, vol. 68, no. 10, pp. 6346–6358, 2020.

[24] C. R. Storck, G. G. de O. Silva, and F. Duarte-Figueiredo, "A vehicle-centric probabilistic approach to virtual cell management in ultra-dense 5g networks," in *Proc. of the IEEE Symposium on Computers and Communications (ISCC)*, 2020, pp. 1–7.

[25] N. Aljeri and A. Boukerche, "An efficient handover trigger scheme for vehicular networks using recurrent neural networks," in *Proc. of the ACM Int. Symposium on QoS and Security for Wireless and Mobile Networks*, 2019, pp. 85–91.

[26] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Elsevier Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, 2020.

[27] Z. Yu, V. Ramanarayanan, D. Suendermann-Oeft, X. Wang, K. Zechner, L. Chen, J. Tao, A. Ivanou, and Y. Qian, "Using bidirectional lstm recurrent neural networks to learn high-level abstractions of sequential features for automated scoring of non-native spontaneous speech," in *Proc. of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2015, pp. 338–345.

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

[29] C. Sommer, D. Eckhoff, A. Brummer, D. S. Buse, F. Hagenauer, S. Joerer, and M. Segata, "Veins: The open source vehicular network simulation framework," in *Springer Recent Advances in Network Simulation*. Springer, 2019, pp. 215–252.

[30] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proc. of the Int. Conf. on Simulation Tools and Techniques for Communications, Networks and Systems*, 2010, p. 10.

[31] G. Nardini, D. Sabella, G. Stea, P. Thakkar, and A. Virdis, "Simu5g–an omnet++ library for end-to-end performance evaluation of 5g networks," *IEEE Access*, vol. 8, pp. 181 176–181 191, 2020.

[32] 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 *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2018, pp. 2575–2582.

[33] E. I. Adegoke, E. Kampert, and M. D. Higgins, "Channel modeling and over-the-air signal quality at 3.5 ghz for 5g new radio," *IEEE Access*, vol. 9, pp. 11 183–11 193, 2021.

[34] M. E. Diago-Mosquera, A. Aragón-Zavala, and M. Rodriguez, "Testing a 5g communication system: Kriging-aided o2i path loss modeling based on 3.5 ghz measurement analysis," *Sensors*, vol. 21, no. 20, p. 6716, 2021.

[35] M. Cicioğlu, "Performance analysis of handover management in 5g small cells," *Computer Standards & Interfaces*, vol. 75, p. 103502, 2021.

[36] M. Murshed, G. H. S. Carvalho, and R. E. de Grande, "Adaptive user-centric virtual cell handover decision-making in 5g vehicular networks," in *Proc. of the IEEE ICC*, 2023, pp. 3983–3988.

Mubashir Murshed is doing his Ph.D. in Computer Science (Intelligent Systems and Data Science) at Brock University, Canada. He received his M.Sc. degree in Computer Science from Brock University, Canada, in 2023. He obtained his B.Sc. degree in Computer Science from the University of Chittagong, Bangladesh in 2019. His research interests include vehicular networks, 5G and 6G networks, cloud computing, adaptive learning, algorithm and protocol design, meta-heuristic algorithms, intelligent transport systems, and smart cities.

Glaucio H.S. Carvalho is an Assistant Professor of Computer Science and Engineering at Brock University, ON, Canada. He holds two Ph.D. degrees: Computer Science from the Toronto Metropolitan University, Canada (2021) and Electrical Engineering from Federal University of Pará, Brazil (2005). His research interests are the cybersecurity, privacy, and dependability of intelligent critical infrastructure systems such as Telecommunications (5G/6G), Intelligent Transportation Systems, and Smart Grids. Dr. Carvalho is an IEEE member and serves the IEEE Future Networks Tech Focus Editorial Board and Security and Privacy Working Group.

Robson E. De Grande is an Associate Professor in the Department of Computer Science at Brock University, Canada. He received his Ph.D. degree in Computer Science from the University of Ottawa, Canada, in 2012. His research interests include large-scale distributed and mobile systems, cloud computing, performance modelling and simulation, computer networks, vehicular networks, intelligent transportation systems, and distributed simulation systems, actively contributing to these areas.