Reinforcement Learning-based User-centric Handover Decision-making in 5G Vehicular Networks

Mubashir Murshed

Submitted in partial fulfillment
of the requirements for the degree of

Master of Science

Department of Computer Science
Faculty of Mathematics and Science
Brock University
St. Catharines, Ontario

Abstract

The advancement of 5G technologies and Vehicular Networks open a new paradigm for Intelligent Transportation Systems (ITS) in safety and infotainment services in urban and highway scenarios. Connected vehicles are vital for enabling massive data sharing and supporting such services. Consequently, a stable connection is compulsory to transmit data across the network successfully. The new 5G technology introduces more bandwidth, stability, and reliability, but it faces a low communication range, suffering from more frequent handovers and connection drops. The shift from the base station-centric view to the user-centric view helps to cope with the smaller communication range and ultra-density of 5G networks. In this thesis, we propose a series of strategies to improve connection stability through efficient handover decision-making. First, a modified probabilistic approach, M-FiVH, aimed at reducing 5G handovers and enhancing network stability. Later, an adaptive learning approach employed Connectivity-oriented SARSA Reinforcement Learning (CO-SRL) for user-centric Virtual Cell (VC) management to enable efficient handover (HO) decisions. Following that, a user-centric Factor-distinct SARSA Reinforcement Learning (FD-SRL) approach combines time series data-oriented LSTM and adaptive SRL for VC and HO management by considering both historical and real-time data. The random direction of vehicular movement, high mobility, network load, uncertain road traffic situation, and signal strength from cellular transmission towers vary from time to time and cannot always be predicted. Our proposed approaches maintain stable connections by reducing the number of HOs by selecting the appropriate size of VCs and HO management. A series of improvements demonstrated through realistic simulations showed that M-FiVH, CO-SRL, and FD-SRL were successful in reducing the number of HOs and the average cumulative HO time. We provide an analysis and comparison of several approaches and demonstrate our proposed approaches perform better in terms of network connectivity.

Acknowledgements

I would like to express my gratitude to my parents, family, and friends for their support during my Master's studies. A special thanks to my supervisor, Dr. Robson E. De Grande, for his guidance and encouragement to make this journey true and valuable. I am appreciative of Brock University for giving me such a prosperous time in my life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Intelligent Transportation Systems (ITS) in urban and highway scenarios aims to improve vehicular safety, and mobility, as well as to provide comfort services [9]. Vehicular Network (VN) is reached a new era of communication where vehicles communicate with each other, infrastructures, devices, and people. They share various ranges of data and information including safety, traffic, comfort, and entertainment [16] [11]. Connected vehicles are an important component of the ITS. ITS have an important aspect to build a successful smart city. Vehicles produce large amounts of data with their On Board Unit (OBU) and exchange information with other vehicles, traffic signals, and smart devices within or outside of the vehicles. With the advancement of technologies, it becomes a basic need to have more reliability, high bandwidth, and low latency connection to provide services [4].

The movement and mobility of vehicles on various road segments are dynamic [7]. Maintaining stable connections for this type of dynamic behaviour of vehicles is an open area for research. Vehicles need to switch between transmission points to remain connected. This event is called Handover (HO). User experience can be influenced by the frequent number of HOs. Moreover, unnecessary HO, frequent HO, and HO time have a massive impact on a reliable connection. When vehicles share safety services, connection loss is not tolerable [12] [6]. HO leads to failure in sharing high-priority safety data. Additionally, in our technological age, entertainment has surpassed services as one of the fundamental needs. An unnecessary HO that results in a connection loss might have an impact on the user's experience.

Connected vehicles transmit enormous quantities of data which could be utilized to build better technologies that result in higher efficiency, improved safety, and a better driving experience along with stable connection [16]. The 5G technology is a promising solution for vehicular networks to deal with the challenges with its high

scalability, ultra-low latency, reduced energy consumption, and ultra-dense network. VN has a broad area of communication, including Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I), Vehicle to Pedestrian (V2P), and Vehicle to Everything (V2X) [16]. Together with VN and 5G technologies provide enormous improvement in the field of vehicular connectivity, which leads to better user experience and improvement in ITS [4].

Many works have dealt with connection stability. Several techniques, such as HetNet, SDN, Fog, Hybrid, and Virtual Cell (VC), contribute to reducing HOs and better connection stability [4] [31] [28] [23]. 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. In this thesis, we introduce an adaptive learning approach to cope with highly dynamic vehicular environments. We thus propose a user-centric approach for virtual cell management using adaptive learning to ensure stable connection by reducing handovers in the 5G vehicular networks. Vehicles continuously communicate with cellular towers while running on different road segments to learn and adjust the adaptive learning model with real-time parameters. Our proposed algorithm determines suitable towers for VC and selects a serving tower to get service.

1.1 Motivation

The high mobility and dynamic behavior of vehicles often result in lost connections, negatively impacting the user experience and compromising the safety services offered by the vehicular network, which is intolerable. Additionally, connection dropouts are brought on by the cellular management's delayed decision-making. In metropolitan settings, network ultra-density is fairly prevalent.

The 5G mmWave (millimeter wave) frequencies in the range of $30GHz$ to $300GHz$, provide more bandwidth and higher data transfer rates compared to traditional networks [20]. The higher frequencies allow for multiple cellular towers or antennas to be used in both the transmitter and receiver, leading to more robust data transmission and higher capacities. However, the higher frequency also leads to greater signal loss due to absorption and scattering by atmospheric gases and obstacles such as buildings, trees, and people. As a result, mmWave signals have a limited range, which leads to a smaller coverage area compared to traditional networks.

The 5G network has several benefits including high bandwidth, ultra-low latency,

and increased energy consumption [20]. User nodes need to perform frequent HO to remain connected to the cellular towers for the low coverage range of 5G mmWave. Frequent and unnecessary HO affect connection stability for dynamic high mobility vehicles [4] [28]. The limited coverage area of 5G networks operating at mmWave frequencies creates a new field of research, such as the merging of multiple cells [23]. Efficient decision-making for HO is essential for a better user experience.

Network virtualization helps to deal with connection stability problems due to the low coverage range of 5G networks [28] [23]. A user-centric virtual cell (VC) strategy enhances network stability, addressing the challenge posed by the limited coverage range of 5G networks. 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 [28] [23]. It helps to maintain stable connections because a vehicle redundantly is aware of the multiple in-range cellular towers, minimizing network instability under sudden drops in signal quality or connection. In addition, by applying adaptive learning techniques to VC and HO management, the network can dynamically adjust to changes in user behavior and network conditions in real-time, by improving network stability. Moreover, time series data-oriented learning can provide a highly personalized and efficient experience for users, while also maintaining stable network connections.

1.2 Objectives

Summarily, the primary goal of this work consists of efficiently reducing the number of handovers and overall handover time for 5G communication in a highly dynamic vehicular environment. This reduction promotes less communication management overhead and less impact on data service delivery. Also, we adopt a virtual cell management strategy, which eases the handover decision-making. In order for us to achieve our goal, we have the following sub-objectives:

- Investigating and developing methodologies for selecting the optimal virtual cell configurations based on connection connectivity requirements and constraints.
- Adaptive decision-making that can be efficiently applied to the dynamic vehicular environment, including factors such as mobility, network congestion, and

signal measurements, to improve network performance and ensure optimal use of resources.

- Efficient HO management based on the current status of the network, by considering real-time scenarios.
- Considered 5G ultra-dense networks and high mobility of vehicles. The 5G network's constraints are taken into account including low coverage range.
- Vehicular trajectory prediction for determining the direction of vehicles to ensure future connectivity.

1.3 Contribution

The main contribution of our work consists of successfully introducing adaptability to the communication management of vehicles in dynamic urban ultra-dense vehicular networks. Our contributions are described as follows:

- We devised an improved static VC and HO management based on FiVH [28]. This improved approach has considered additional mobility and communication parameters and an algorithmic redesign.
- We introduced an adaptive learning-based approach with SARSA RL for efficiently managing VC and HO to reduce the number of HOs and HO time. The approach learns online and is on-policy, which brings flexibility and adaptability to an urban setup. This contribution has been accepted and recognized as an excellent contribution by reviewers in **IEEE International Conference on Communications**.
- We enhanced our learning approach by incorporating time series data-oriented LSTM with adaptive SARSA RL for better performance on VC management and HO operations. This enhanced version allows more stable connections by reducing the number of HO and HO times.
- We conducted extensive performance analyses with diverse 5G tower deployments, vehicular densities, and vehicular mobility in realistic, real-time simulation scenarios. The analyses proved the adaption efficiently reduced the HO management overhead in 5G vehicular networks.

1.4 Outline

The rest of this thesis is organized as follows. Chapter 2 contains the background of our work. Chapter 3 presents related works for connection stability problems. Chapter 4 defines the problem formulation. Chapter 5 contains our proposed modified probabilistic approach. In the beginning, we start implementing the work in [28], then add parameters and modify the algorithm. Chapter 6 presents our proposed adaptive approach using SARSA RL. Chapter 7 discusses our extended version of the adaptive approach, where we introduce both time series data-oriented LSTM and adaptive SARSA RL on the same page. By altering the settings and method, we stick to the majority of the concepts from the adaptive approach. Chapter 8 discusses performance analysis of several approaches. Finally, Chapter 9 concludes the thesis, summarizing it and providing future work directions.

Chapter 2

Background

Technological advancement in vehicular networks shows us a new paradigm of research. Though several novel strategies for communication, such as DSRC, 4G, and 5G have been introduced and many research can be found that deal with highly dynamic environments, many shortcomings of these paradigms have already been identified [4]. Most works have been done for considering a subset of the area and did not consider real-world road traffic scenarios. Moreover, 5G networks have a lower coverage range compared to previous generations of mobile networks due to the higher frequency bands they use to transmit data. However, 5G technology offers many benefits over previous generations, such as higher speeds, lower latency, and lower energy consumption. There is an open research direction of continuing to work on improving the technology to increase coverage. There are some solutions for handling low communication range are introduced, such as small cell deployment, repeaters, directional antenna placement, and network virtualization. Network virtualization is proving to be a particularly promising solution among them due to its benefits [23]. It is important to study and investigate the architecture of such technologies like Vehicular Networks, 5G New Radio, Virtual Cell, and Adaptive Learning techniques. It is valuable to discuss the main aspects and technological advancements in support of the research paradigm.

2.1 Vehicular Networks

Vehicular Network (VN) is a type of wireless network that is used for communication between vehicles, infrastructure, and other connected devices [13]. VN is designed to support safety-critical and non-safety-critical applications in the transportation sector, such as ITS, connected vehicles, and autonomous vehicles. It provides a

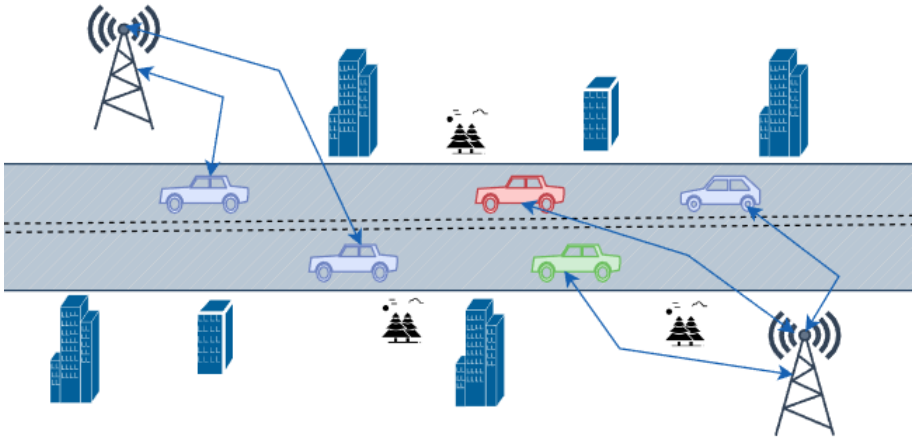


Figure 2.1: Scenario of Vehicular Network.

platform for exchanging information between vehicles, infrastructure, and other connected devices. By sharing information, VN has the potential to increase road safety, alleviate traffic congestion, and improve the overall driving experience. Vehicular Ad-hoc Network (VANET) is the spontaneous version of the VN for the domain of vehicles [9]. It focuses Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), Vehicle-to-Pedestrian (V2P), and Vehicle-to-Everything (V2X) communications architectures to provide roadside services.

With the increasing demand of users and technological advancement, VANET directs to a new paradigm, where vehicles equipped with sensors, software, and technologies that communicate with other vehicles, roadside infrastructures, pedestrians, machines, and devices with the aim of connecting and exchanging data over the network [5]. Vehicular Network is improving in such a way that soon it will be one of the enablers for autonomous, connected, and smart vehicles. Figure 2.1 is the scenario of a vehicular network, where vehicles are connected to cellular towers and get different services.

2.2 5G New Radio

The communication era is directed to a new world with 5G New Radio network [1]. It is the new generation mobile network that shows the world a new direction which comes to global wireless standards after 4G networks. 5G networks enable a new paradigm of the network that is designed to connect a very large amount of users including machines, objects, and devices. It has features like high data speeds, reliability, ultra-low latency, increased availability, massive network capacity, and a

uniform user experience. 5G opens a new window for vehicular networks in terms of ultra-dense network, ultra-reliability, and low latency for safety and comfort services experienced in vehicles.

Though 5G New Radio has lots of advancements, it faces some shortcomings including communication range. 5G millimeter waves use frequencies from $30 - 300$ gigahertz and frequency waves are able to travel a short distance. Moreover, 5G frequency can be interrupted by physical obstructions such as trees, towers, walls, and buildings. Extending the number of existing cell towers to increase the communication range could be a solution and but it increases cost. To counter this drawback, researchers are working to direct a solution to overcome this limitation by ensuring other prospects.

2.3 User-centric Network

In the traditional communication network system, nodes are connected to base stations or transmission towers. The decision-making and computation are done on the base station side. The base station-centric network can face computational overload for ultra-density of user nodes. This network paradigm has some issues such as high latency, connection drop for unbalanced network load, etc. Authors in [21] and [39] show the benefit of the user-centric network over the base station-centric network.

User-centric Network is a network architecture where users are the center of the network [23]. It is the opposite of the base station-centric network. It is an emerging concept that can overcome all the drawbacks of the base station-centric network. In the user-centric network, nodes are responsible for decision-making and computational operations [40]. A personalized network experience offers by the user-centric network where users have greater convenience in making decisions. It is useful for unbalanced network loads by shifting network decisions to nodes. It has lots of advantages for throughput and latency for 5G vehicular networks.

2.4 Handover

Handover (HO) is a process in wireless communication networks where a user node switches from one network connection to another while maintaining its network session. This enables the device to maintain its communication without interruption as it moves from one coverage area to another. In cellular communication, cellular towers have a limited communication range. For high mobility of user nodes, it is not

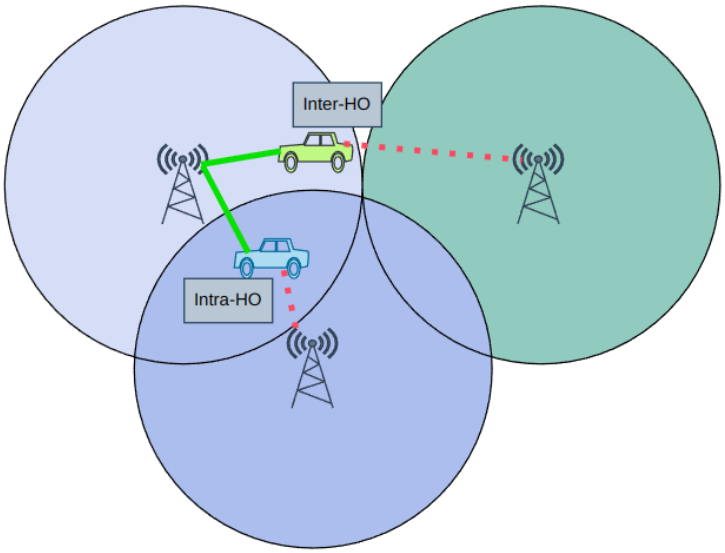


Figure 2.2: Scenario of Intra-Handover and Inter-Handover.

possible to get connected with the same cellular tower. The procedure of transferring data and services from one tower is necessary to ensure connection stability. Though the HO procedure takes time to execute, it plays a core element in planning and deploying cellular networks.

HOs can be classified into several types. There are two types of HO, vertical and horizontal in terms of networks. Horizontal HO occurs when a node moves within the same type of network technology. On the other hand, vertical HO is the HO between different network technologies. Horizontal HO and vertical HO are also defined as intra-HO and inter-HO, respectively. There is another kind of HO classification based on link establishment, soft-HO, and hard-HO. Soft-HO makes a connection establishment before disconnecting with the previous cellular tower. Hard-HO drops the connection to connect with another tower. These classifications vary from network requirements. A scenario of intra-HO and inter-HO is provided in Figure 2.2.

2.5 Virtual Cell

Virtual Cell (VC) is a new concept that shifts the base station-centric point of view to the user-centric for Vehicular Networks [23]. In VC, vehicles are connected to multiple transmission cellular tower(s) to form a virtual cell [23] [39]. When a vehicle moves to a certain direction, it updates the VC by adding and releasing the appropriate trans-

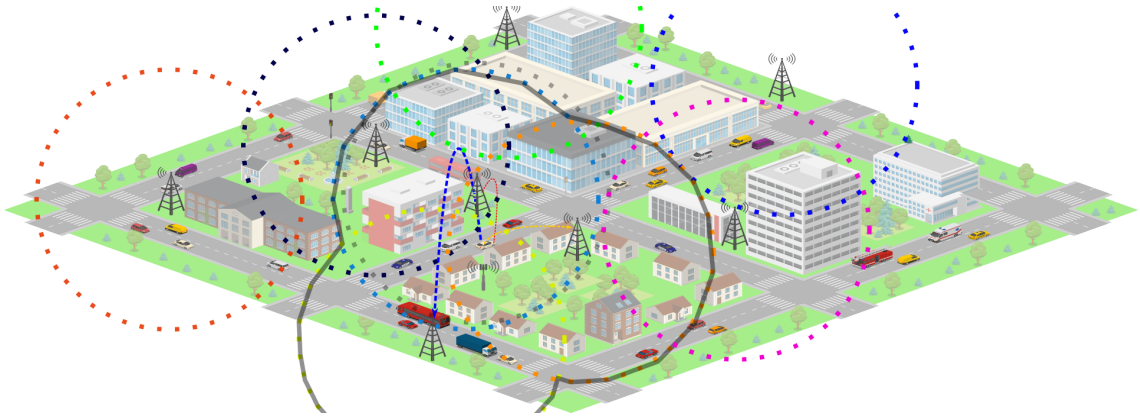


Figure 2.3: Scenario of Virtual Cell.

mission cellular tower(s). This helps to enlarge the communication range of 5G New Radio [28]. As vehicles are connected to multiple transmission cellular tower(s), they can get relief from frequent HOs. In this way, stable connections can be maintained in the dynamic high mobility vehicular environment. Moreover, vehicle is responsible to compare and compute its performance parameters, so there is less time required to perform the HO operation. The limitations of ultra-dense networks can be overcome using VC [23]. The scenario in Figure 2.3 illustrates VC, where multiple vehicles form VCs with multiple cellular tower(s). In VC, the size of the VC cell is determined by the combined communication range of all cellular tower(s) that are within the VC of a vehicle.

2.6 State-Action-Reward-State-Action Reinforcement Learning

Reinforcement Learning (RL) is the subdomain of machine learning where agents learn and take actions to the environment [30]. The goal is to maximize the cumulative reward by learning based on agent faces from the environment. RL focuses to find a balance between current knowledge and the goal of what agents learn from the environment.

State-Action-Reward-State-Action (SARSA) Reinforcement Learning algorithm is a variation of the Q-Learning algorithm. In SARSA Reinforcement Learning (SRL) algorithm, the learning agent learns the value function according to the current action derived from the current policy. On each step of interaction, the learning agent receives input from the environment as a form of indication of the state of the envi-

ronment, therefore deciding on an action to get as output. The action changes the state of the environment depending on the reward for the action. The main function for updating the Q-value depends on the current state and action of the agent. For choosing an action it gets a reward. The agent enters the next state after taking that action and determines the next action.

SRL is suitable for decision-making cellular network connectivity because of its ability to handle complex and dynamic environments. It updates the value of an action based on the expected future reward for that action given the current state and the next state and action. This makes it suitable for modeling the behavior of cellular networks, which are subject to constantly changing conditions and require real-time decision-making. Additionally, SRL can handle large state spaces, making it well-suited for optimizing network connectivity in large-scale cellular networks.

SRL *state:* describe the current situation of a system for a given environmental condition; *action:* next state is obtained by applying an action on available actions on the present state; *policy:* set of rules that are followed by the RL agent to determine the action for the current state; *reward:* a return value that is given by the environment for changing the state of the system. In SRL, when an agent is in a state s_k, an action a_k is taken for policy π with reward r_{k+1}, taking it to the next state s_{k+1} [30]. An action a_{k+1} is taken in the state s_{k+1} with policy π. SRL works on the current policy, defined through the tuple $(s_k, a_k, r_{k+1}, s_{k+1}, a_{k+1})$, where Q-values are updated with state-action transitions following α learning rate and γ discount factor [38]. The Q-value is updated using Equation 2.1 with the state-action transitions. Figure 2.4 presents the architecture of the SRL algorithm.

$$Q(s_k, a_k) \leftarrow Q(s_k, a_k) + \alpha[r_{k+1} + \gamma Q(s_{k+1}, a_{k+1}) - Q(s_k, a_k)] \tag{2.1}$$

2.7 Long Short-Term Memory

Long Short-Term Memory (LSTM) is a type of Recurrent Neural Network (RNN) architecture used for processing sequential time series data [26]. It has been designed in a such way that it completely solves the vanishing gradient problem while maintaining the integrity of the training model. LSTM networks are well-suited to classifying, processing, and making predictions based on time series data, since there can be lags of unknown duration between important events in a time series. In some situations, long-time delays are overcome using LSTMs, which can also handle noise, dispersed representations, and continuous input. The large range of parameters of-

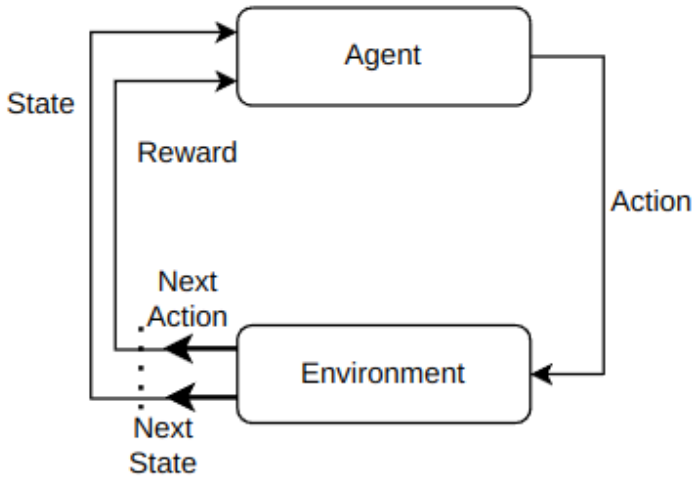


Figure 2.4: Architecture of SARSA RL.

fered by LSTMs includes learning rates, and input and output biases. Thus, there is no need for precise modifications. There is an advantage in that the complexity of the LSTM is decreased to $\mathcal{O}(1)$ for updating each weight. LSTM is composed of a cell, 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.

Bidirectional Long Short-Term Memory (BLSTM) is a type of LSTM where the processing of input data is in two directions: forwards and backward [36]. The forward processing captures information from past events in the sequence, while the backward processing captures information from future events. This bidirectional processing allows BLSTM to capture information from both the past and future context, leading to improved performance compared to traditional LSTM. BLSTM has the ability to remember important information from earlier time steps, making them suitable for tasks where the order of the input data matters and a change in the network state at one-time step can have an impact on the network state at later time steps.

BLSTM can be chosen for cellular network connectivity prediction tasks because of its ability to handle sequential data effectively. In such tasks, the network needs to consider past and future inputs in order to make accurate predictions. The bidirectional aspect of the BLSTM allows it to capture both past and future context, while the LSTM component provides the ability to capture long-term dependencies in the data. This makes BLSTM well-suited for tasks such as predicting cellular network connectivity, where the network needs to consider both past and future data in order to make accurate predictions.

Chapter 3

Related Works

Connection stability is an ongoing concern in vehicular networks where several works have attempted to support HOs efficiently [31]. In such works, more stable connections can be achieved by reducing the number of HOs, decreasing HO times, and managing communication cells. Although not all works deal with HO and VC management directly, they have also contributed in terms of connection stability.

SDN, HetNet, Fog, Hybrid based approaches are well-explored areas to deal with HO [31]. There is not a lot of research that addressed situations of 5G networks in the actual world. Some of them performed effectively in a variety of situations, but not every situation was looked at once. In order to guarantee a steady connection in actual urban and highway traffic scenarios, improvements are required. Research with 5G networks' high mobility and ultra-density simultaneously is difficult. High vehicular mobility was overlooked in certain works that addressed ultra-dense networks, whereas the contrary was true in others. Additionally, not all real-world V2X scenarios are thoroughly addressed.

In our study, we have discovered that the majority of earlier works are base station-centric. There are several studies that utilize probabilistic methods, but comparatively fewer studies employ adaptive learning mechanisms for connection stability.

3.1 Heterogeneous Network

HetNet is a heterogeneous network that integrates cellular networks with different types of wireless communication technologies. It is considered as a potential solution for stable communication and mobility management in vehicular networks. It combines two or more network technologies together to overcome network stability issues.

In [37], authors proposed a HetNet-based solution to deal with HO for connection stability. They used Reward-based Markov Decision Process in 5G networks to reduce frequent HO by using parameters, such as connection signal, power, bandwidth, location, and velocity. They optimized the overall service delivery to user nodes by avoiding excessive HOs while maintaining the stability of connections. Their works ignored intolerable HOs compared to other benchmark schemes.

Wu et al. in [34] proposed a dynamic fuzzy Q-Learning method for small cell networks for mobility management. They simulated by considering a UE movement with an average speed of $10km/h$ which is not high mobility. HetNet provides direction to many problems, but the mostly seen issue of HetNet is the deployment and transition between different cells and ping-pong events.

Tuysuz et al. in [32] proposed a probability analysis by considering parameters such as HO cost, channel busy times, and traffic class. They aim to reduce the number of cells, HO, and reduce scanning overhead. They considered fixed assumptions of network measurements which is not efficient for dynamic scenarios.

3.2 Software Defined Network

Software Defined Networks (SDN) is a network architecture where it is divided into control and data planes. The data plane is responsible for forwarding packets to connected wireless or wired networks and the control plane is responsible for determining the path as SDN controllers. In the SDN-based method, each controller has an open Application Programming Interface (API) to support the programming capability of network infrastructure. The API enables administrators to control the data and control plane rules and policies remotely.

Authors in [24] proposed an SDN-based MEC-enabled approach to a service-aware HO management phase in 5G vehicular networks. Their work focused on continuous service and seamless coverage by reducing the number of HO.

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 [10].

Sharma et al. in [25] proposed an SDN-based method using utilized Unmanned Aerial Vehicles (UAVs) as an on-demand forwarding scheme, where UE's act as terminals to UAVs. They proposed two approaches, a centralized approach, and distributed approach. The RSSI, RSRP, RSRQ, and channel quality indicators are used as measurement reports and signaling overhead. They used HO latency and delay as

performance evaluation matrices.

3.3 Fog-based Network

The fog-based network shifts network functions closer to end-users. It reduces latency and improves the Quality of Service. Fog uses macro-cell base stations and WiFi access points as fog servers. It shifts necessary control methods to the Fog Access point.

Authors in [16] presented a machine learning model for HO scheme using vehicular fog to minimize the disruption time during the HO process. They used a feed-forward neural network to determine an optimal fog node and distance between fog and vehicle and time as parameters. However, their work was limited to HO between fogs. Moreover, it was not tested on heterogeneous networks and was built on the assumption of known vehicle trajectories. The most common issues the Fog-based method faces are when and where to offload high and low-speed vehicles in different coverage areas.

Machine learning has also been used for HO management. A two-tier Machine Learning-based scheme was introduced for HO management in intelligent vehicular networks [3]. This work predicted signal strength using a recurrent neural network model to predict the receiving signal strength for a handover decision, and a new AP was determined with a stochastic Markov model.

3.4 Virtual Cell

The majority of earlier studies on connection stability focus on base station-centric networks. [4]. Road safety and traffic monitoring messages are high-priority messages which do not allow connection loss and latency. In the base station-centric approach, the network selection procedure could face latency issues as it needs to be served from a remote base station [18]. A user-centric approach can perform the network selection by itself, it does not face latency problems. Moreover, the user-centric approach is useful in highly congested networks by migrating computational tasks to the user end. Decision-making does not face latency issues in user-centric networks.

Researchers have recently become aware of the user-centric strategy for reliable connections. Another promising solution for connection stability issues is network virtualization. Researchers' attention has turned to Virtual Cell (VC) technologies

as a solution to 5G networks' ultra-density and limited coverage range [28]. Research is ongoing, although there aren't many works that deal with virtual cells.

Several works have explored VC management for maintaining stable connections in 5G networks. VCs have served to constrain effective data dissemination for a group of vehicles called hotspots (HSs) to maximize the number of served HSs and minimize the total power radiated in 5G vehicular networks [23]. The approach consists of three main stages, including VC-admission control which is responsible for balancing network load, and intra-VC optimization, inter-VC power control.

An approach for V2X communications in 5G networks focused on forming a VC with low energy consumption and high reliability by sharing the same channel [22].

A probabilistic approach has been proposed to deal with VC formation and update [29]. The approach follows a user-centric perspective to facilitate VC management, allowing more decentralized decision-making and lowering the network complexity. Probabilities represented the status of the network – base station degree, betweenness, and distances – where these parameters are used to determine virtual cell size. Similarly, the VC paradigm also helped to deal with handovers in 5G V2X networks, using similar probabilities and parameters to make handover decisions [28]. However, these probabilistic approaches cannot guarantee efficient HO management in highly dynamic vehicular environments.

Focusing on an individual device, a method has been introduced to determine the optimal radius of a virtual cell to maximize the system downlink capacity using distance and Remote Radio Head (RRH) density [39].

VC helped support a scheme that selects a static cluster of small cells with local mobility to serve as an anchor in high-density scenarios, attempting to enable stable connections [17].

A dynamic user-centric scheme allowed updating VCs through the mobile tracking of the vehicles, and use a max-min fair problem for resource management in V2X communication [35].

3.5 Remarks

A summary of our findings from previous works is presented in Table 3.1.

In our study, we have discovered that the majority of earlier works were base station-centric and often failed to deliver stable connections. In addition, there are problems with power consumption, network complexity, dealing with ultra-dense networks, failure in HO, and HO time. A user-centric strategy may be more effective in

Table 3.1: Summary of Previous Works.

Work	HO	V2X	5G	UC	Conn	HM	Approach	Target	Process
[37]	✓		✓		✓	✓	HetNet	Reducing HO	Markov Decision Process
[34]	✓				✓		HetNet	Mobility Mng	Dynamic Fuzzy Q-Learning
[32]	✓		✓		✓	✓	HetNet	HO, Overhead	Probabilistic Analysis
[24]	✓				✓	✓	SDN	Mobility Mng	Multi-access Edge Comp
[25]	✓				✓	✓	SDN	Reducing Handover	UAV, On-demand forwarding
[16]	✓				✓	✓	Fog	Minimize HO time	Feed Forward NN
[29]			✓	✓	✓	✓	VC	VC formation	Probabilistic Estimate
[23]		✓	✓	✓	✓		VC	Limit broadcasting	Probabilistic Estimate
[28]	✓	✓	✓	✓	✓	✓	VC	VC formation, HO	Probabilistic Estimate
[39]	✓		✓	✓	✓	✓	VC	Optimal VC radius	Scheduling Algorithm
[17]	✓		✓	✓	✓		VC	Reducing HO	Local Anchor Cell Control
[22]		✓	✓	✓	✓	✓	VC	Minimize broadcasting	Probabilistic
[3]	✓	✓	✓		✓	✓	ML	Handover Decision	Stochastic Markov model
This	✓	✓	✓	✓	✓	✓	**VC**	**VC & HO mng**	**RL & HML based 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

addressing stable connections, which would address all of the issues. Furthermore, it is found that the user-centric virtual cell did not include HO management. Despite the fact that certain studies have addressed user-centric Virtual Cell, these works are based on a straightforward probabilistic equation-based methodology. They do not guarantee appropriate HO management and VC formation in real-time circumstances. Previous studies didn't adjust HO in 5G networks for instances with dynamic vehicular environments. A learning-based VC formation technique is required since it will improve HO performance and guarantee reliable connection by minimizing HO. In addition, previous studies often relied heavily on numerical analysis, providing mathematical accuracy and predefined assumptions about the network scenario. Moreover, it may not always align with real-world conditions. A study using simulation techniques to model the behavior of the system in a virtual environment that closely resembles real-world conditions, leading to a more comprehensive examination of the system.

Chapter 4

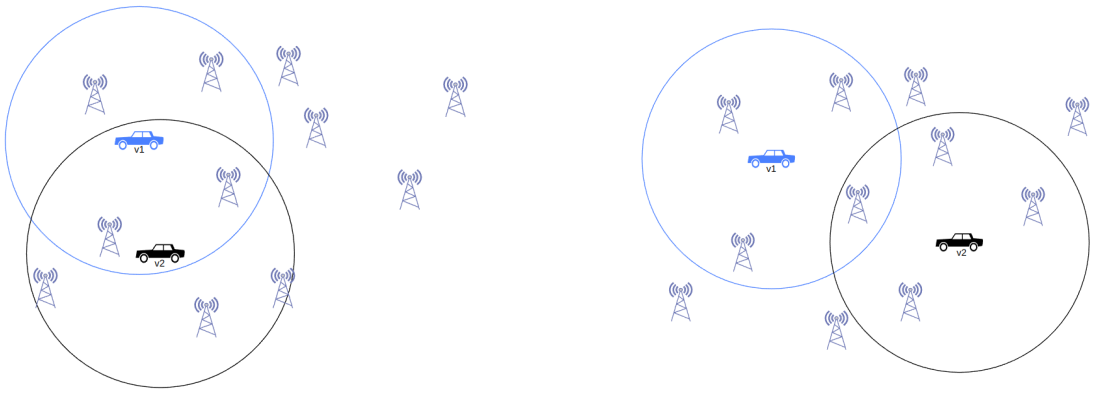
Problem Formulation

Numerous studies that aim to guarantee stable connections have been addressed thus far. They reduce HO by using a variety of technologies for HO management. Only urban or highway settings are taken into account in certain studies. Instead of simulating scenarios, the majority of them do numerical analysis.

Roads are segmented in the real-world traffic scenario, and vehicle movement is dynamic within each section of the road. Additionally, the speed limit differs from road to road. In an urban setting, there is a high volume of traffic that is unpredictable due to time constraints. The congested traffic on roads causes ultra-dense network and unbalanced traffic load among transmission towers. Because of the high mobility of vehicles on the highway, frequent connection losses are experienced when changing serving towers. Due to excessive traffic congestion, accidents, and roadblocks, there are also uncertain traffic situations or changes in the way that vehicles travel. Simple probabilistic calculations, historical statistical data, or predefined models cannot be used to address these dynamic circumstances.

Assume a dynamic vehicular environment where vehicles traverse an urban region following the topology of road segments. There exists a finite number of vehicles in that transportation scenario, denoted as $V = \{v_1, v_2, ..., v_i\}$. The network scenario consists of several 5G cellular towers (gNodeB) denoted as $TCT = \{tct_1, tct_2, ..., tct_j\}$. Towers TCT_j are randomly distributed in the scenario and are connected to each other. Tower has a fixed communication range. TCT has all features of 5G networks, including high bandwidth, low latency, and low coverage range. When vehicle v_i comes to the communication range of cellular towers of TCT at time k, this set of in-range towers, denoted as $CT = \{ct_1, ct_2, ..., ct_j\}$, where $CT \subset TCT$. v_i can exchange beacon messages with cellular towers $ct_j \in CT$ in the range.

Our approach is a user-centric approach. vehicles are responsible for taking deci-



(a) Position-2 of vehicle v2. (b) Position-1 of vehicle v2.

Figure 4.1: Simplistic scenario of Virtual Cell formation of two vehicles.

sions and performing operations. Vehicles need to communicate with cellular towers and depend on cellular towers for the calculation of connectivity parameters. Every vehicle v_i has a high computational On-Board-Unit (OBU) installed and is able to communicate with transmission towers.

In the network cellular communication technology, vehicles switch their serving towers to get connected and services, in another word, perform handover operations. The low-coverage range is the most concerning issue for 5G networks. Installing many transmission towers may be a solution, but it is costly and can occur other issues. These facts lead to frequent handovers, connection drops, ping-pong effects, and unnecessary handovers. Virtual cellular technology can be a potential revolutionary solution [28].

By the definition of cellular network virtualization, the communication range of a vehicle is dependent on the communication range of the connected cellular towers. A vehicle can be connected to multiple cellular towers to form a virtual cell. The cell moves with vehicles. In Figure 4.1, a simplistic scenario of updating the virtual cells for two vehicles is given. We can see that the vehicle of black color is in a position-1 in Figure 4.1b. There are four cellular towers connected to the black vehicle. In Figure 4.1a, the position of the black vehicle changes to position-2 and also its connected towers, which are five. The position and connected cellular towers are in the same position for the blue vehicle in Figure 4.1b and 4.1a. It should be noted that the virtual is not necessary to be a circle, it can be different shapes according to the communication range of a vehicle and connected cellular towers. We draw it as a circle to make it easily understandable. Figure 2.3 can be used as a reference VC size.

Although there have been several studies on problems with connection stability. They adhere to simple procedures or are based on predetermined models. It is ineffective to rely on a predetermined model when taking into account the dynamic real-world circumstance. Additionally, there are a number of factors that affect virtual cells and handover management, including signal parameters, time, speed, and network load. A way to deal with the dynamic, high-mobility nature of vehicular environments is through adaptive learning from real-time scenarios.

To guarantee reliable connections, an appropriate virtual cell (VC) construction is required. Computational complexity can be raised by adding cellular towers or making VCs overly large. On the other hand, VC with a smaller size could struggle to keep connections solid.

We use an adaptive learning technique to solve the connection stability problem by taking into account every scenario. We employ the SARSA RL algorithm because it adapts its behavior relying on the situation and learns from the current policy. In this work, we break down the connection stability problem into two parts: *(1)* virtual cell management and *(2)* handover management. We develop several distinct approaches. In the first approach, we modified a probabilistic approach. Then in the following approach, we rely on adaptive SARSA RL to update the VC dynamically and act to handover. Real-time learning is vital for this method, as is a back-and-forth strategy for sustaining stable connections. We combine time series data-oriented learning with adaptive learning in the third approach. Because cell management occurs less often, virtual cell management is carried out using time series data-oriented learning, while handover management is carried out using adaptive SARSA because it depends on real-time circumstances. As parameters for our methods, we take signal measurement, mobility, and density into consideration. In the approaches we propose, virtual cells are formed in a way that ensures appropriate virtual cell management and prevents connection dropouts. In order to improve user experience and connection stability, handover decision–making involves choosing the best cellular tower.

Chapter 5

Modified Probabilistic Approach

Connection stability relies on different parameters including signal measurement reports, distance, and speed. Parameters impact differently in terms of connection stability. We propose a modified version of the probabilistic approach named FiVH [28], to reduce the number of handovers by determining the appropriate size of the virtual cell (VC) and handover operation. We apply the concept of FiVH to form VC but modify the parameters settings and design of the algorithm. We try additional parameters to the FiVH to make it more effective for the dynamic changes of the environment. Moreover, we adjust the design of the algorithm to be more suitable to the dynamic behavior of different road segments. Our modified proposed version of FiVH named Modified-FiVH (M-FiVH) is discussed in this chapter.

In our proposed M-FiVH approach, we focus to determine VC with appropriate cellular towers and handover operation. An appropriate size of VC helps to balance network load and better stable connectivity by ensuring less number of handovers. Moreover, a proper number of cellular towers in VC helps to maintain stable connections with minimal computation cost.

5.1 Virtual Cell

The virtual cell (VC) adoption is a user-centric approach where vehicle v_i creates and updates its vc_i. We consider the 5G network technology in this work. We assume, 5G network performs like SDN, so there is no need for an additional SDN controller in this work [23]. Every vehicle v_i is traversing on different road segments of a scenario where there are several transmission cellular towers. When v_i comes to the transmission range of a cellular tower $ct_j \in CT$, they exchange beacon messages to share connectivity information. High vehicular mobility and ultra-dense networks

are taken into account in this work.

In M-FiVH, we define the speed thresholds by following 3GPP Technical Report (TR) 38.913 which are adequate by cell type (pico, micro, and macro) as in [28]. We define the threshold of speeds for the selection of VC size by considering real-world traffic scenarios [28]. v_i with a speed $0 - 30km/h$ selects pico-cell; micro-cell is selected when speed is $31 - 120km/h$; and macro-cell with a speed $>= 121km/h$. This threshold is to determine the size of VC, which means discount tolerance while selecting towers.

5.2 Connectivity Metrics

In our M-FiVH, we consider all metrics as in FiVH, such as BS degree, betweenness, and distance. In addition to these matrices, we consider the signal measurement metric RSSI. We have discovered via our study and experiment, that signal measurement affects connection stability [8]. We use the same method as FiVH to calculate metrics. The difference is in how we employ them in the algorithm, which improves connection stability by reducing the number of handovers.

5.2.1 BS degree

We consider the probability of BS degree in this approach. The probability of BS degree $\mathcal{X}_{pBSdeg}$ is the information regarding to the load of a cellular tower. The $\mathcal{X}_{pBSdeg}$ describes the information related to the number of vehicles connected to a particular cellular tower. With this information, we can determine how effectively a cellular tower serves a vehicle. Because a cellular tower with a heavy load leads to network traffic congestion, connection drops. The cellular tower has a high degree means that more vehicles are connected to that tower, and it is the most active cellular tower in the network. The tower with a higher degree has a lower probability to be chosen for VC vc_i of v_i.

The vertex degree of a cellular tower and all of its neighboring cellular towers, and connected vehicles to that cellular tower are considered for the calculation of $\mathcal{X}_{pBSdeg}$. We calculate $\mathcal{X}_{pBSdeg}$ with the same formula in [28], that is 1 (one) minus the division between its vertex degree ($vDeg$) and the sum of the vertex degrees of its neighboring towers($NvDeg$). $vDeg$ is obtained by counting the number of vehicles and towers reachable from a particular cellular tower and $NvDeg$ is the count of neighbouring vehicles and towers reachable. The $\mathcal{X}_{pBSdeg}$ is calculated by following

the Equation in 5.1.

$$\mathcal{X}_{pBSdeg} = 1 - \frac{vDeg}{NvDeg} \tag{5.1}$$

In our user-centric approach, when v_i comes to the communication range of $ct_j \in CT$, it receives the $\mathcal{X}_{pBSdeg}$ with the exchanged beacon messages. The matrices are stored in the OBU of v_i for later computation and decision-making.

5.2.2 Betweenness

Betweenness is the centrality of inter-mediation. The most influential nodes within the network are indicated using betweenness. The strength of a connection and its stability can be measured with betweenness. The higher betweenness of a tower decreases the probability to be chosen as a candidate cellular tower for vc_i.

We calculate the probability of betweenness $\mathcal{X}_{pBetw}$ which helps to measure the importance of a cellular tower. The probability of betweenness $\mathcal{X}_{pBetw}$ is calculated using Equation 5.2 [28]. In equation 5.2, g_{pq} is the shortest path from source (p) to destination (q), and h^upq is the shortest path from p to q passing through u.

$$\mathcal{X}_{pBetw} = \frac{1}{\frac{(N-1)(N-2)}{2}} \sum_{p \neq q \neq h} \frac{h^u_{pq}}{g_{pq}} \tag{5.2}$$

In our user-centric approach, v_i receives h^upq when it comes to the range $ct_j \in CT$. It is stored in the OBU of v_i for later computation. The g_{pq} is calculated on the OBU of v_i. The $\mathcal{X}_{pBetw}$ is calculated for all towers $ct_j \in CT$ that come to the communication range of v_i.

5.2.3 Vehicle Distance to Tower

The position of a vehicle is one of the main factors to choose as a potential cellular tower. The smaller distance makes the higher possibility of a tower being chosen for VC.

v_i traverses on different road segments. It stores the coordinates (x, y) of its traversing path. We assume every towers of TCT share their location coordinates among themselves. When v_i comes to the range of $ct_j \in CT$, it receives the location coordinates of ct_j with the exchanges beacon messages. Vehicle v_i calculates the distance with its coordinates and location coordinates of $ct_j \in CT$.

5.2.4 Received Signal Strength Indicator

Received Signal Strength Indicator (RSSI) is an estimated measure of the power level that a vehicle receives from a tower. The signal strength is increased proportionally to the RSSI.

v_i exchanges beacon messages with $ct_j \in CT$. The beacon messages contain RSSI for every tower in CT. Vehicle-tower pair calculates $\mathcal{X}_{Rssi_{ct_j}}$ using Equation 5.3, where n is signal propagation constant or exponent, d is the distance from sender to receiver and A is received signal strength at a defined distance.

$$\mathcal{X}_{Rssi_{ct_j}} = 10n \log_{10}(d) + A \tag{5.3}$$

The cellular tower ct_j with a higher value of $\mathcal{X}_{Rssi_{ct_j}}$ is selected for the vc_i. This metric has an impact on VC formation for connection stability along with other metrics. Adding this metric to our proposed approach, makes M-FiVH dynamically behave in the vehicular environment. RSSI provides a reliable and simple method to determine the strength of the signal being received by a user node. It can be affected by obstacles, interference, and weather. By considering the reasons behind connection stability depending on signal measurement, adding RSSI to the connectivity metrics improve handover decision-making.

5.3 M-FiVH

Our proposed approach M-FiVH is designed in such a way that it decreases the number of handovers by adding-releasing appropriate cellular tower(s) of the virtual cell. Moreover, it takes the decision on the handover operation. Connectivity metrics, such as BS degree, betweenness, distance of the vehicle to a cellular tower, and RSSI are fatal to VC and HO decisions. We consider the individual probability of these metrics.

Our proposed user-centric M-FiVH selects towers $ct_j \in CT$ from the network with towers that have the best probability of BS degree, betweenness, and best value of the vehicle to cellular tower's distance and RSSI. For minimizing the computation complexity of our proposed approach, we determine a threshold for all connectivity metrics, threshold of the probability of BS degree $\mathcal{X}_{pBSdeg_{th}}$, threshold of the probability of betweenness $\mathcal{X}_{pBetw_{th}}$, threshold of the distance $\mathcal{X}_{Dist_{th}}$, and threshold of the RSSI $\mathcal{X}_{Rssi_{th}}$. These thresholds are calculated with their average values using Equations 5.4, Equations 5.5, Equations 5.6, Equations 5.7 in each time interval $k' - k$.

$$\mathcal{X}_{BSdeg_{th}} = \frac{\mathcal{X}_{pBSdeg_1}, \mathcal{X}_{pBSdeg_2}, ..., \mathcal{X}_{pBSdeg_K}}{k} \tag{5.4}$$

$$\mathcal{X}_{Betw_{th}} = \frac{\mathcal{X}_{Betw_1}, \mathcal{X}_{Betw_2}, ..., \mathcal{X}_{Betw_K}}{k} \tag{5.5}$$

$$\mathcal{X}_{Dist_{th}} = \frac{\mathcal{X}_{Dist_1}, \mathcal{X}_{Dist_2}, ..., \mathcal{X}_{Dist_K}}{k} \tag{5.6}$$

$$\mathcal{X}_{Rssi_{th}} = \frac{\mathcal{X}_{Rssi_{ct_{j_1}}}, \mathcal{X}_{Rssi_{ct_{j_2}}, ..., \mathcal{X}_{Rssi_{ct_{j_k}}}}}{k} \tag{5.7}$$

Vehicle v_i exchanges beacon messages to every tower $ct_j \in CT$, that comes to its communication range. When the connectivity metrics $\mathcal{X}_{pBSdeg}$, $\mathcal{X}_{pBetw}$, $\mathcal{X}_{Dist}$, $\mathcal{X}_{Rssi_{ct_j}}$ of $ct_j \in CT$ satisfy the thresholds, i.e. $\mathcal{X}_{pBSdeg} <= \mathcal{X}_{BSdeg_{th}}$, $\mathcal{X}_{pBetw} <= \mathcal{X}_{Betw_{th}}$, $\mathcal{X}_{Dist} <= \mathcal{X}_{Dist_{th}}$ and $\mathcal{X}_{Rssi_{ct_j}} >= \mathcal{X}_{Rssi_{th}}$, that subset of $ct_j \in CT$ is selected for later computation. Later, it selects the best four cellular towers having the best connectivity metrics. From the selected cellular towers from four connectivity metrics, a maximum of four cellular towers is selected for an initial virtual cell vc'_i, which has the best probability. In other words, for each metric, a cellular tower can be selected and same cellular tower(s) can be selected by different connectivity metrics with a max size of four. After selecting cellular towers for vc'_i, our proposed M-FiVH calculates the distance between the vehicle v_i and the selected four towers $ct_j \in vc'_i$. The maximum distance $\mathcal{Z}_{maxDist}$ among vehicle v_i to cellular towers $ct_j \in vc'_i$ is selected as the radius of vc_i. M-FiVH adds all cellular towers $ct_j \in CT$ within the radius $\mathcal{Z}_{maxDist}$, to the vc_i of v_i. For handover operation, v_i selects its serving tower from the towers of vc_i. v_i calculates a threshold $\overline{\mathcal{Z}_{Rssi_{ct_j}}}$ with the average RSSI of $ct_j \in vc_i$. When $\mathcal{X}_{Rssi_{ct_j}}$ of tower $ct_j \in vc_i$ greater than $\overline{\mathcal{Z}_{Rssi_{ct_j}}}$, ct_j is selected for intra-VC HO. Whenever a cellular tower has max RSSI rather than vc_i, it performs inter-VC HO. The algorithm of the proposed user-centric approach M-FiVH is provided in Algorithm 1.

This algorithm runs for the management of the virtual cell at every time step k. v_i uses connectivity metrics as input and the formation of the virtual cell as output. Four loops on line 1...12 of the algorithm execute for each tower in the communication range. Four towers have been selected with respected operations. The last loop in line 13...14 works four times for calculating the maximum distance among the selected tower(s) from connectivity metrics. All of the towers in between this maximum distance, are included in the virtual cell as in line 15. In line 16, an average value of RSSI $\overline{\mathcal{Z}_{Rssi_{ct_j}}}$ is calculated with the RSSI of all towers in vc_i. The

Algorithm 1: Algorithm M-FiVH

Data: $\mathcal{X}_{BSdeg}$, $\mathcal{X}_{Betw}$, $\mathcal{X}_{Dist}$, $\mathcal{X}_{Rssi_{ct_j}}$, $\mathcal{X}_{speed_i}$
Result: vc_i

1 **for** $ct_j \in CT$ **do**
2 $\quad \mathcal{Z}_{pBSdeg} = min(\mathcal{X}_{pBSdeg})$;
3 $\quad \tau_1 = \mathcal{Z}_{pBSdeg}$;
4 **for** $ct_j \in CT$ **do**
5 $\quad \mathcal{Z}_{pBetw} = min(\mathcal{X}_{pBetw})$;
6 $\quad \tau_2 = \mathcal{Z}_{pBetw}$;
7 **for** $ct_j \in CT$ **do**
8 $\quad \mathcal{Z}_{Dist} = min(\mathcal{X}_{Dist})$;
9 $\quad \tau_3 = \mathcal{Z}_{Dist}$;
10 **for** $ct_j \in CT$ **do**
11 $\quad \mathcal{Z}_{Rssi_{ct_j}} = max(\mathcal{X}_{Rssi_{ct_j}})$;
12 $\quad \tau_4 = \mathcal{Z}_{Rssi_{ct_j}}$;
13 **for** $i = 1; i <= 4; i++$ **do**
14 $\quad \mathcal{Z}_{maxDist} = calculateMaxDistance((v_i, \tau_1), (v_i, \tau_2), (v_i, \tau_3), (v_i, \tau_4))$;
15 $vc_i \leftarrow getTowers(\mathcal{Z}_{maxDist})$;
16 $\overline{\mathcal{Z}_{Rssi_{ct_j}}} \leftarrow avg(\mathcal{X}_{Rssi_{ct_j}} \in vc_i)$;
17 **if** $(\mathcal{X}_{Rssi_{ct_j}} > \overline{\mathcal{Z}_{Rssi_{ct_j}}})$ **then**
18 $\quad performHO$;

handover operation is carried out on lines 17...18 when a cellular tower has a greater RSSI value than $\overline{\mathcal{Z}_{Rssi_{ct_j}}}$ in $ct_j \in vc_i$.

The time complexity for M-FiVH relates to the number ct_j that comes to the range of v_i. M-FiVH calculates $\mathcal{Z}_{pBSdeg}$, $\mathcal{Z}_{pBetw}$, $\mathcal{Z}_{Dist}$ and $\mathcal{Z}_{Rssi_{ct_j}}$ for every tower $ct_j \in CT$. The algorithm runs in each time k. The overall time complexity of M-FiVH is $\mathcal{O}(ct_j \in CT * k)$.

Chapter 6

Adaptive Connectivity-oriented VC Decision-making

We propose a Connectivity-oriented virtual cell decision-making mechanism for ensuring stable connection in 5G networks using SARSA Reinforcement Learning (SRL). Our user-centric approach, Connectivity-oriented SARSA Reinforcement Learning (CO-SRL), runs in the OBU of each vehicle. CO-SRL works in two phases: *(i)* Connectivity Factor (CF) adjustment and *(ii)* Virtual Cell and handover decision-making. The decision-making is adapted through the SRL algorithm. Every vehicle computes and generates a measurement report that includes CFs. The architecture of CO-SRL is given in Figure 6.1

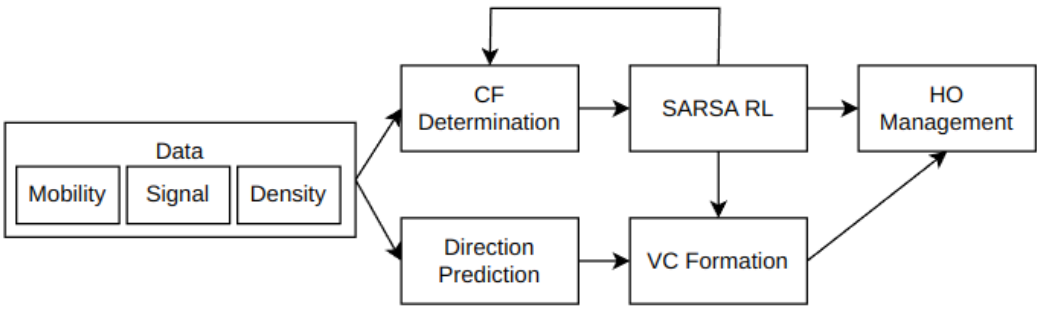


Figure 6.1: Architecture of CO-SRL.

The measurement reports are used as the input of phase *(i)* and for further necessary computation related to CFs. The output of phase *(i)* is a set of cellular towers. Connectivity Factor (CF) adjustment, which is used to form a virtual cell, provides a

set of cellular towers used as the input of phase *(ii)*. In the second phase, the output is the selection and switching of the serving cellular tower among the set of cellular towers selected for a virtual cell using CO-SRL. Other than that, it updates the hysteresis value for CFs, which makes virtual cell selection dynamic with the execution of the adaptive learning approach. We compute a hysteresis for every CF, which describes the dynamic suitable value of CF for the formation of VCs and connections. This behaviour enhances the adaptivity to the dynamic environment of our proposed. Initially, CO-SRL chooses predefined default values for CFs and hystereses. With time, CO-SRL updates them dynamically.

6.1 Connectivity Factors

Virtual cell management and intra-VC and inter-VC handover decision depend on the Connectivity Factors (CFs). CO-SRL considers the Received Signal Strength Indicator (RSSI), Signal to Interference & Noise Ratio (SINR) and Reference Signal Received Power (RSRP), distance, tower load, movement direction, and vehicle speed as CF. When a vehicle comes to the communication range of a cellular tower, it exchanges beacon messages with the tower. The beacon messages contain parameters related to the computation of CFs. We consider all the factors that have an impact on network connectivity because they may affect and impact connectivity differently [8] [25]. Considering several signal measurements in CFs allows dealing with more diversity of environments that have an impact on the stable network connection [8]. We use a dynamic hysteresis value for adjusting signal measurements according to highly mobile vehicles.

6.1.1 Vehicle Direction Prediction

We assume that vehicle mobility is random and dynamic on different road segments, leading to frequent HO. Knowing the direction of a vehicle allows us to identify cellular towers it might be in range in the near future and update its VC with those towers, which helps to reduce the number of hard HO. We assume every tower in TCT shares coordinates with them and v_i. Each vehicle v_i stores the coordinates of its traversed path. With the readings for a certain time period, a vehicle calculates its recent movement direction and estimates the next position it may visit in the near future. We use a simple Linear Regression (LR) model to predict a vehicle's future direction using a list of its traversed coordinates.

In our user-centric approach, every vehicle v_i has its own LR model to predict the tower ct_j^{vdp}, to which v_i is heading. LR is a lightweight model with low computation cost. We select a time interval $k' - k$ to update the LR so that it can be best fitted where its update frequency follows the dynamic mobility of vehicles. We assume that the future location of a vehicle is related to a tower location. So, we need to predict the future coordinate of the vehicle and need to determine which tower is nearer to the predicted coordinates for a vehicle.

Let the initial coordinate of a vehicle v_i be (x_0, y_0). Vehicle v_i traverses a sequence of road segments for a time interval $(k_0, k_1, ..., k)$, building a series of position recordings on a two-dimensional space $P_{(x,y)} = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, where n is the number of readings kept in the vehicle's recording history. Our approach's LR model is described in Equations 6.1, 6.2 and 6.3.

$$y = mx + c \tag{6.1}$$

$$m = \frac{n\sum(x_n y_n) - \sum(x_n)\sum(y_n)}{n\sum(x_n^2) - (\sum(x_n))^2} \tag{6.2}$$

$$c = \frac{\sum(y_n)\sum(x_n)^2 - \sum(x_n)\sum(x_n y_n)}{n\sum(x_n^2) - (\sum(x_n))^2} \tag{6.3}$$

In Equation 6.1, x and y are the coordinates, and m and c are slope and intercept respectively. We obtain slope m and intercept c through Equation 6.2 and 6.3, where x_n and y_n are the n^{th} (x, y) coordinate of vehicle v_i for a time interval of $k' - k$. The calculation of $\sum(x_i y_i)$, $\sum(x_i)$, $\sum(y_i)$, $\sum(x_i)^2$, $\sum(x_n)$, $\sum(y_n)$ support the LR minimum squares.

Thus, each vehicle $v_i \in V$ searches the closest $ct_j^c \in TCT$ based on its direction. According to Equation 6.1, it predicts $y'_{ct_j^c}$ in a time interval of $k' - k$ for every tower in CT and its neighbouring towers, $CT^c \in TCT$. Then, the smallest difference $vc_i \cup \arg\min_x(|y'_{ct_j^c} - y_{v_i}|) : ct_j^c \in CT^c$ is selected as a possible tower ct_j^{vdp} to serve the vehicle in the future based on its mobility, adding ct_j^{vdp} to vc_i.

6.1.2 Speed

We consider speed as a CF. Vehicular speed is independent of each other. Every vehicle store its speed history on the OBU for further processing.

When v_i traverses on different road segments, signal parameters may vary due to the speed changes. A low-speed vehicle can maintain a certain signal measurement more than a high-speed vehicle. Moreover, adaptive learning has an impact on different ranges of speed.

We consider speed to determine the size of the virtual cell [28]. Different sizes of cells are formed for different speed ranges. Cell size impacts on adaptive learning mechanism which is described later in this chapter.

6.1.3 Signal to Interference & Noise Ratio

Signal to Interference & Noise Ratio (SINR) is the ratio of the signal level to the noise level. It is used to determine the signal quality. The SINR value is measured in dB, the higher the value, the better the signal quality. In the dynamic vehicular environment, it is not efficient to use a constant range of signal levels to determine the good quality of signals.

The equally spaced time intervals $k_1, k_2, ..., k_n$ are followed for calculating SINR $sinr_{ct_j}$. In each time k, vehicle v_i receive beacon messages from $ct_j \in CT$. The beacon messages contain the SINR of the vehicle-tower pairs. $sinr_{ct_j}$ of vehicle v_i for each towers ct_j is stored in its computing area.

The SINR for each vehicle is calculated using Equation 6.4, where P is the power of the incoming signal of interest, I is the interference power of the interfering signals and N is noise term.

$$sinr_{ct_j} = \frac{P}{I + N} \tag{6.4}$$

We use a dynamic hysteresis to maintain stable connections in a real-time environment. Our proposed CO-SRL adjusts hysteresis dynamically from phase *(ii)* where adaptive learning is used to learn from real-time scenarios.

Vehicle v_i uses equally spaced time intervals $k_1, k_2, ..., k_n$ to exchange the generated SINR through beacon messages with their respective parameters of CT. The generated measurements of SINR are stored in OBU of v_i. The measurement values of the signals are later used for the calculation hysteresis of CFs.

$$\chi^i_{sinr_k} = \frac{sinr_{ct_j}}{\chi^i_{sinr_{k-1}}} + \beta^i_{sinr} \tag{6.5}$$

A vehicle determines the hysteresis $\chi^i_{sinr_k}$ for SINR using Equation 6.5. Here, $sinr_{ct_j}$ is the SINR of CT, and β^i_{sinr} is computed from the phase *(ii)* of CO-SRL.

The hysteresis $\chi^i_{sinr_k}$ is used for adjusting SINR at each time period k to find which towers are suitable to be in vc_i for better connection stability.

6.1.4 Received Signal Strength Indicator

Received Signal Strength Indicator (RSSI) is an estimated measure of the power level that a vehicle is receiving from a tower. A lower RSSI value means that the signal is weak. We need to find a standard threshold for RSSI to maintain a stable connection.

In our approach, RSSI is calculated at each time k for every vehicle. The beacon messages received by vehicle v_i from towers $ct_j \in CT$ contain RSSI. These values of RSSI are stored in the vehicular computing area of vehicle v_i. Vehicle v_i calculates RSSI $rssi_{ct_j}$ for each tower $ct_j \in CT$. RSSI is calculated using Equation 6.6, where n is signal propagation constant or exponent, d is the distance from sender to receiver and A is received signal strength at a defined distance.

$$rssi_{ct_j} = 10n \log_{10}(d) + A \tag{6.6}$$

CO-SRL calculates hysteresis dynamically from phase *(ii)* where adaptive learning is used to learn from real-time scenarios.

Vehicle v_i exchanges the generated RSSI through beacon messages with their respective parameters of CT using equally spaced time intervals $k_1, k_2, ..., k_n$.

$$\chi^i_{rssi_k} = \frac{rssi_{ct_j}}{\chi^i_{rssi_{k-1}}} + \beta^i_{rssi} \tag{6.7}$$

The measurement values of $rssi_{ct_j}$ are later used for the calculation hysteresis of CFs. The hysteresis of RSSI $\chi^i_{rssi_k}$ is generated using Equation 6.7 at each time period k, where β^i_{rssi} is computed from the phase *(ii)* of CO-SRL.

6.1.5 Reference Signal Received Power

Reference Signal Received Power (RSRP) is the average power of the received signals or the level of the received signal from a transmission tower. The RSRP value is measured in dBm. The RSRP value closer to -100 dBm, is considered as a poor signal, and a signal more than -90 dBm to -80 dBm is considered good signal strength. In our adaptive approach, we do not consider a predefined range of RSRP as good or poor signal strength. Our system updates this signal strength range according to the real-world scenario.

Vehicle v_i uses equally spaced time intervals $k_1, k_2, ..., k_n$ to exchange the generated RSRP through beacon messages. RSRP is calculated using Equation 6.9 in each time k, where $rsrp_{ct_j}$ is calculated from the Equation 6.6 and N is the number of resource blocks.

$$rsrp_{ct_j} = rssi_{ct_j} - 10\log(12N) \tag{6.8}$$

It is not efficient to use a constant range of RSRP signal levels to determine the good quality of signals. CO-SRL generates a dynamic hysteresis value $\chi^i_{rsrp_k}$ for RSRP according to highly mobile vehicles. CO-SRL adjusts $\chi^i_{rsrp_k}$ dynamically from phase *(ii)*.

The generated measurements are stored in OBU of v_i. The measurement values of the signals are later used for the calculation hysteresis of CFs.

$$\chi^i_{rsrp_k} = \frac{rsrp_{ct_j}}{\chi^i_{rsrp_{k-1}}} + \beta^i_{rsrp} \tag{6.9}$$

In Equation 6.9, $\chi^i_{rsrp_k}$ is generated from RSRP $rsrp_{ct_j}$ of time k and hysteresis of $\chi^i_{rsrp_{k-1}}$ of time $k-1$ of CT. The phase *(ii)* of CO-SRL generates β^i_{rssp} to make the measurement adaptive to the environment.

6.1.6 Distance

The distance of a vehicle to cellular towers is an important factor for connection stability. We use hysteresis $\chi^i_{dist_k}$ to update the suitable distance for maintaining connections between vehicle and cellular towers. We assume that the locations of cellular towers are shared with all vehicles in the scenario. Vehicle v_i calculates its distance to $tct_j \in TCT$ using their coordinates at time k.

In our user-centric approach, every vehicle v_i calculates its hysteresis $\chi^i_{dist_k}$ using Equation 6.10, where the minimum distance $dist_(v_i, tct_j)$ between v_i and any tower in TCT is divided by the previous hysteresis $\chi^i_{dist_{k-1}}$, and β^i_{dist} is determined from the adaptive learning in phase *(ii)*.

$$\chi^i_{dist_k} = \frac{\min_{j \in TCT}\left(dist(v_i, tct_j)\right)}{\chi^i_{dist_{k-1}}} + \beta^i_{dist} \tag{6.10}$$

6.1.7 Tower Load

Tower load describes the number of vehicles connected to a cellular tower where heavy load hampers network stability. In our user-centric approach, each v_i receives the load of $ct_j \in CT$ through beacon messages. We also assume all towers in TCT share information related to tower load among themselves.

Vehicle v_i estimates a ratio (R_{ct_j}) of a tower's load using Equation 6.11, where l_{ct_j} is the number of vehicles associated with tower ct_j, and $\sum_{l_{tct_n} \in TCT}(l_{tct_n})$ is the sum

of the number of associated vehicles with towers in TCT in the scenario. This ratio provides a relative load related to the density of a singular cellular tower. The higher the ratio of R_{ct_j}, the higher the density of the cellular tower and the less chance is to select ct_j for vc_i of v_i. The calculated R_{ct_j} is used to determine a hysteresis $\chi^i_{load_k}$ in each time k.

$$R_{ct_j} = \frac{l_{ct_j}}{\sum_{l_{tct_n} \in TCT}(l_{tct_n})} \tag{6.11}$$

Equation 6.12 describes hysteresis $\chi^i_{load_t}$ which is calculated using the load ratio of tower load R_{ct_j} divided by hysteresis of previous time period $\chi^i_{load_{k-1}}$, which is summed with adaptive learning output's β^i_{load}.

$$\chi^i_{load_k} = \frac{R_{ct_j}}{\chi^i_{load_{k-1}}} + \beta^i_{load} \tag{6.12}$$

6.2 SARSA Reinforcement Learning Algorithm

We use SARSA Reinforcement Learning (SRL) which is a model-free RL approach in this work. In our user-centric approach, each vehicle has its own OBU unit. Vehicle communicates with towers of its communication range, and collects and computes information such as SINR, RSSI, RSRP, distance, tower load, speed. These data exchanged between vehicle and towers with beacon messages, and are used as the connectivity factors to behave adaptivity to the real-world traffic scenario.

The State-Action-Reward-State-Action Reinforcement Learning (SRL) algorithm is a model-free, online TD learning technique that updates Q-values using the current policy's actions [38]. The algorithm operates in the following manner. When the agent is in state s_k, it takes action a_k following policy π and receives reward r_{k+1}, leading to the next state s_{k+1}. The agent then takes action a_{k+1} in state s_{k+1} following policy π. SRL operates based on the current policy. SRL is represented as the tuple $(s_k, a_k, r_{k+1}, s_{k+1}, a_{k+1})$. It updates Q-values using the state-action transitions with a learning rate of α and discount factor of γ using Equation 6.13.

$$Q(s_k, a_k) \leftarrow Q(s_k, a_k) + \alpha[r_{k+1} + \gamma Q(s_{k+1}, a_{k+1}) - Q(s_k, a_k)] \tag{6.13}$$

We use the SRL algorithm for adaptive learning in the dynamic vehicular environment. We have figured out some of the advantages to select this algorithm in our work. SRL is considered to be well-suited for network handover management due to its many advantages [30]. SRL can handle online learning, which is crucial for real-time network handover decision-making. SRL updates its policy based on the current

state and the next state, which allows it to consider the immediate consequences of its actions, making it well-suited for scenarios with fast-changing conditions, such as network handovers. It is an on-policy algorithm, which means that it updates its policy based on the actions taken by the current policy, rather than an estimated value function. This makes it more suitable for virtual cell and handover management, where the goal is to optimize a specific policy rather than to find an optimal value function. Moreover, SRL does not need any predefined dataset, training, or testing, which makes it effective for the dynamic vehicular network. Decisions on the high dynamic behaviour of vehicles are hard to achieve efficiently with pre-defined and static models.

6.3 Connectivity-oriented SARSA RL Algorithm

In the phase *(ii)*, our proposed CO-SRL identifies the best-suited cellular tower ct_j^{ϑ} to serve the vehicle v_i among the towers $ct_j \in vc_i$, and it decides intra-VC or inter-VC handover. It also adjusts CF's values so that they are updated dynamically from the adaptive learning.

We define our proposed CO-SRL model components as vehicular connectivity requirements to ensure network stability in dynamic and high-mobility vehicular networks. CO-SLR performs its phase *(ii)* operation each time k.

State

The state s_k is the cellular tower that serves v_i at time k. It is similar to the current serving tower ct_j^{ϑ}. The state s_{k+1} can be any cellular tower from vc_i.

Action

There are several possible actions $a_k \in \{a_{k_1}, a_{k_2}, ..., a_{k_m}\}$ in a state s_k - same number as tower in vc_i. Action $a_{k,p}$ is the possible available migrations from the serving cellular tower to available cellular towers in vc_i at time k. 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} – the serving tower ct_j^{ϑ} changes as the cellular tower of state s_{k+1}.

Reward

The reward r_{k+1} is obtained by taking action a_k of transition s_k to s_{k+1}. The scalar combination of the CFs of ct_j^{ϑ} is used as the reward r_{k+1}.

6.3.1 Policy Gained - Connectivity Factors

We use a known ϵ-greedy policy of RL in our approach, which handles trade-off between exploration and exploitation [30]. At time k, ϵ_k is calculated from the scalar combination of CF averages ($\overline{CT_{rssi}}$, $\overline{CT_{rsrp}}$, $\overline{CT_{sinr}}$, $\overline{CT_{dist}}$, $\overline{CT_{load}}$) of a time interval among the towers in vc_i. An action is initially 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 $(1 - \epsilon_k)$.

6.3.2 Intra-VC Switching Tower and Inter-VC Handover

Initially, when v_i registers in a network, vc_i is assigned with predefined values of CFs, which is suitable for 5G network connectivity. It chooses state, action, and reward according to the first tower that comes to its range. As time progresses, the Q-value of CO-SRL is measured by applying an action in a state and with its corresponding reward.

We consider Reference Signal Received Quality (RSRQ) as Q-value. RSRQ is calculated using RSRP and RSSI. It demonstrates the overall performance of a connection and provides additional information when other decision parameters are not sufficient to take a crucial decision for connectivity management [14]. The higher the RSRQ is, the better the signal. Thus, the higher Q-value indicates a cellular tower is better for connection stability. However, we use the scalar combination of CFs as the reward, so CO-SRL maintains the diversity of factors that have an impact on connectivity.

CO-SRL calculates Q-value for every tower in vc_i; in other words, the number of Q-values is equal to the number of available actions in a state. We define the update of Q-value calculation as in Equation 6.14.

$$Q(s_k, a_k) \leftarrow Q(s_k, a_k)^{\vartheta} + \alpha[CF_{scalar_{k+1}} + \gamma rsrq_{ct_j}^{mbr} - rsrq_{ct_j}^{srv}] \tag{6.14}$$

Assume at time k, the switching of serving cellular tower – intra-VC handover – has occurred. At that time 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 $Q_j^i(s_{k+1}, a_{k+1})$ – 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 $Q_j^i(s_{k+1}, a_{k+1}) = rsrq_{ct_j}^{mbr}$. In Equation 6.14, $rsrq_{ct_j}^{mbr}$ is the RSRQ of a cellular tower in VC and $rsrq_{ct_j}^{srv}$ is the RSRQ of the serving tower in vc_i. These values are updated in every time period k. Note that, the $rsrq_{ct_j}^{srv}$ 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.

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 is equal to 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. Thus, the cellular tower in vc_i gains the maximum Q-value selected as the new serving tower ct_j^{ϑ}, and the value of $Q(s_k, a_k)^{\vartheta}$ is updated accordingly. Since the Q-value is updated based on the RSRQ and CFs of CT, the selected serving tower provides the best connection strength in CT. The β_{rssi}^i, β_{sinr}^i, β_{rsrp}^i, β_{dist}^i, β_{load}^i are also updated for the CFs as the new serving tower ct_j^{ϑ}. This phenomenon ensures better performance as all the parameters related to connectivity are considered while calculating Q-value, and a high Q-value specifies better connectivity performance. Moreover, our proposed CO-SRL updates CFs according to the output of CO-SRL, which ensures CFs dynamically match the environment.

The Algorithm 2 describes our proposed approach CO-SRL. CFs are the input of the algorithm which leads to the serving tower as the output, along with adaptive learning values for phase *(i)*. The algorithm works till there is tower(s) in VC. From line 3...9, the algorithm starts with the calculation of the values for every hysteresis and predicts the future direction of the vehicle. These executions determine VC. Then action, reward, policy, learning rates, and discount factors of SARSA RL are assigned in line 11...14. The Q-value generation is done in line 14. The switching of serving cellular tower or intra-VC HO and update of CFs are done from line 15...18 when Q-value gets max for a cellular tower. Connection drop leads to inter-VC HO and redirects to the initial condition which is followed as mentioned at the end.

We analyze the overall time complexity of CO-SRL. CO-SRL runs each time k, and updates CFs, performs direction prediction of vehicle, and SRL operation. Every vehicle is independent in our proposed user-centric CO-SRL. v_i runs CO-SRL in each time k and the operation needs to be performed for every tower in vc_i. So, the overall complexity of CO-SRL is $\mathcal{O}(ct_j \in vc_i)$. We provide the time complexity of SRL, as in specific, in CO-SRL, we are focusing on SRL. The time complexity of the SRL algorithm depends on the number of states, actions, and updates performed during the learning process [30]. In CO-SRL, the number of states is always 1 as there is always one serving tower, action is the number of towers in vc_i, and the update is performed in each time period k. We can say the time complexity of SRL of CO-SRL is $\mathcal{O}(ct_j \in vc_i * k)$.

Algorithm 2: Algorithm CO-SRL for adaptive learning

Data : $rssi_j$, $sinr_j$, $rsrp_j$, (x, y), $dist$, $load_j$, $speed_i$
Result: ct_j^{ϑ}, β_{rssi}^i, β_{sinr}^i, β_{rsrp}^i, β_{dist}^i, β_{load}^i

InitializeCondition;
while $vc_i \neq \emptyset$ **do**
  calculate $\chi_{rssi_k}^i$, $\chi_{sinr_k}^i$, $\chi_{rsrp_k}^i$, $\chi_{dist_k}^i$, $\chi_{load_k}^i$;
  computeDirectionPrediction();
  if ct_j *satisfies* χ^i **then**
    $vc_i \leftarrow ct_j$;
  if *timeInterval* $k' - k$ **then**
    determine ct_j^{vdp};
    $vc_i \leftarrow vc_i \cup ct_j^{vdp}$;
  $a_k \leftarrow vc_i$;
  $r_{k+1} \leftarrow assignReward()$;
  $\epsilon_k \leftarrow computePolicy()$;
  set α, γ; //depending on speed
  $Q(s_k, a_k) \leftarrow Q(s_k, a_k)^{\vartheta} + \alpha[CF_{scalar_{k+1}} + \gamma rsrq_{ct_j}^{mbr} - rsrq_{ct_j}^{srv}]$;
  if *(*$\arg\max_{ct_j}(Q(s_k, a_k))$ **then**
    execute switching serving tower to new ct_j^{ϑ};
    $Q(s_k, a_k)^{\vartheta} \leftarrow \arg\max_{ct_j}(Q(s_k, a_k))$;
    update β_{rssi}^i, β_{sinr}^i, β_{rsrp}^i, β_{dist}^i, β_{load}^i;
  if *(connectionDrop in* vc_i*)* **then**
    execute inter-VC HO to neighbouring cellular tower;
    break;

6.3.3 Cell Size Selection:

We have considered pico (small), micro (medium), and macro (large) size cells in our approach [28]. Speed is used as the triggering factor for selecting cell size. Because a high-speed vehicle needs more frequent learning than a vehicle with a lower speed, the learning rate α and discount factor γ are adjusted depending on the speed of a vehicle. We consider three categories of speeds: faster ($121km/h$ to $180km/h$), fast ($61km/h$ to $120km/h$), and medium ($0km/h$ to $60km/h$). In our CO-SRL, when a vehicle moves at a faster speed, the macro cell is chosen with a slightly lower value of α and γ. Accordingly, the fast- and medium-speed vehicles choose a micro cell and pico cell with comparatively higher values of α and γ than faster speed as given in Table 8.1. This mechanism is applied in a time interval $k' - k$.

Chapter 7

Factor-distinct Adaptive VC Decision-making

Vehicular mobility and movement depend on both historic data and real-time data about the environment. Vehicular mobility and directions are quite uncertain depending on different road segments, time, weather, and road-side events. Vehicle direction, speed, and value of signal measurement parameters vary due to the dynamic vehicular movement. Sometimes, only adaptive learning or a historical data-based learning model may not be efficient for 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 pre-defined model is not efficient for predicting with an old dataset, and requires high computational performance and time, which is not suitable for vehicles running on the road. By considering all of the challenges, we propose a user-centric approach FD-SRL which consists of both lightweight time-series data-oriented learning and adaptive learning. FD-SRL is an improvement of CO-SRL, 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 parts *(i)* time-series data-oriented Virtual Cell (VC) management and *(ii)* adaptive learning-based handover (HO) management. In part *(i)*, a virtual cell is formed using time series data-oriented Bidirectional LSTM (BLSTM) and HO decision is performed in part *(ii)* using SARSA Reinforcement Learning (SRL). We use time series-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 towers to provide services [2]. For this reason, we are using time-series data-oriented learning for VC management. On the other hand, HO management is mostly dependent

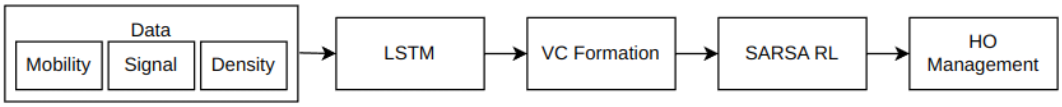


Figure 7.1: Architecture of FD-SRL.

on real-time data and situations, which is efficient in dealing with adaptive learning. To enhance performance in dynamic scenarios by taking into account the dynamic behaviour, we propose a blend of model-based and model-free learning methods. The output of part *(i)* time-series data-oriented VC management, is a set of cellular towers for virtual cell, which is used as the input of part *(ii)* adaptive learning-based handover management. The outcome of part *(ii)* is the selection of a serving tower or intra-VC inter-VC HO. The architecture of FD-SRL is given in Figure 7.1

7.1 Learning Factors

We use Bidirectional LSTM (BLSTM) for Virtual Cell (VC) management. BLSTM uses time series data to predict future value [26] [36]. BLSTM is composed of a cell, 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 design the simple-lightweight structure of BLSTM and the training and testing data to run faster within the timeframe.

Assuming a region of the plane with TCT number of cellular towers. Vehicle v_i receives beacon messages from cellular towers CT when they come in the transmission range. The beacon message consists of connectivity parameters.

Connectivity Parameters

We consider RSSI, SINR, RSRP, distance, tower load, and speed as the connectivity parameters of FD-SRL. These connectivity parameters are calculated in the same way as described in Section 6.1.3 - 6.1.7. In our user-centric FD-SRL approach, we combine RSSI, SINR, RSRP, distance, and tower load to a scalar value ϱ^j. This scalar value ϱ^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 7.3 of this chapter.

$$\varrho^j = (rssi_{ct_j} + sinr_{ct_j} + rsrp_{ct_j} + dist + load_{ct_j})/p \tag{7.1}$$

In Equation 7.1, we take the sum of all of the connectivity parameters and divide them with the total number of connectivity parameters p to calculate the scalar value ϱ^j.

Execution of BLSTM

We design the BLSTM model according to the context of vehicular networks. The input layer of BLSTM contains the input vector of scalar value ϱ^j at time k. The feeding values $\varrho^j = \varrho^j_1, \varrho^j_2, ..., \varrho^j_k$ 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_k is computed based on the input ϱ^j.

A BLSTM process the data in both forward and backward direction with two separate hidden layers. An input sequence $\varrho^j = \varrho^j_1, \varrho^j_2, ..., \varrho^j_k$, computes the hidden vector sequence $h = h_1, h_2, ..., h_k$ and output vector sequence $g = g_1, g_2, ..., g_k$ [36]. 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 of which are the same size as the hidden vector h. This later 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 W terms (denote weight matrices), the b terms denote bias vectors by following the equations 7.2 7.3 and 7.4.

$$\overrightarrow{h} = \mathcal{H}(W_{\varrho\overrightarrow{h}}\varrho_k + W_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{h}_{k+1} + b_{\overrightarrow{h}}) \tag{7.2}$$

$$\overleftarrow{h} = \mathcal{H}(W_{\varrho\overleftarrow{h}}\varrho_k + W_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{h}_{k+1} + b_{\overleftarrow{h}}) \tag{7.3}$$

$$g = W_{\overrightarrow{h}g}\overrightarrow{h}_k + W_{\overleftarrow{h}g}\overleftarrow{h}_k + b_g \tag{7.4}$$

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 input and output squashing functions of the cell. We modify the traditional BLSTM to make it lightweight, and by following the requirement of network connectivity parameters. We use 50 neurons, activation function *ReLU*, optimizer *Adam*, and 100 epochs in the training and test process.

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 do not store data or enrich datasets for a long time period. Because in the high dynamic mobility scenario, a pre-defined model using an old dataset of a long time 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 ϱ^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 time period $k_1, k_2, ..., k'_t$ to perform training and testing of BLSTM. A window size k' is defined for the prediction of ζ^i where the size of the input vector of $\varrho^j_{k'}$.

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 ϱ^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, respectively.

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 ϱ^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.

It is noted that Mean Square Error (MSE) is employed to evaluate the performance of BLSTM. MSE provides a quantitative measure of the model's accuracy in making predictions by measuring the average difference between the actual and predicted values. We use a predefined maximum error rate to identify the performance of the model. 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 is greater than the predefined maximum error rate,

Algorithm 3: Algorithm FD-SRL - Part *(i)*

Data : $rssi_j$, $sinr_j$, $rsrp_j$, (x, y), $dist$, $load_{ct_j}$
Result: vc_i, ζ^i, ct_j^{ϑ}

1 $\varrho^j \leftarrow rssi_j$, $sinr_j$, $rsrp_j$, (x, y), $dist$, $load_{ct_j}$;
2 **for** k_1 **to** k'_{t-1} **do**
3 $\mathcal{D}^i \leftarrow \varrho^j$;
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** $\varrho^j >= \zeta^i$ **then**
12 $vc_i \leftarrow ct_j$;
13 $ct_j^{\vartheta} \leftarrow AlHoMgmt(vc_i)$;

$\mathcal{M}^i$ 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 3 describes the part *(ii)* of VC management. Connectivity parameters are stored and updated in a dataset as train and test data from line 2...7. Then 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.

The time complexity of part *(i)* of our proposed FD-SRL relates to the number of towers in vc_i. We ignore the time complexity of generating BLSTM because we focus on the time the algorithm executes. FD-SRL generates the BLSTM model after each time interval $k'-k$. The connectivity metrics calculation, scalar value calculation, and update of the dataset are performed each time k. We can say, the overall complexity of FD-SRL is $\mathcal{O}(ct_j \in vc_i * k)$.

7.2 Factor-distinct SARSA RL Algorithm

In part *(ii)* of FD-SRL, vehicle v_i takes the decision about the HO management. v_i selects a suitable tower ct_j^{ϑ} for getting service among $ct_j \in vc_i$. FD-SLR decides

the HO decision, i.e. intra-VC or inter-VC handover decision, to maintain stable connections using the SRL algorithm.

We define our proposed FD-SRL model components as vehicular connectivity requirements to ensure network stability in dynamic and high-mobility vehicular networks. FD-SLR performs its part *(ii)* operation each time k. We have modified the CO-SRL. In most cases, FD-SRL is the same as CO-SRL, but we have the design of FD-SRL.

7.2.1 Intra-VC Switching Tower and Inter-VC Handover

The part *(i)* of FD-SRL provides a set of towers for vc_i, which is used as the input of part *(ii)*. Initially, when v_i registers in a network, vc_i is determined with a default value of ζ^i, which is suitable for 5G network connectivity. It chooses the state, action, and reward of the part *(ii)* according to $ct_j \in vc_i$ with max ϱ^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.

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

7.2.2 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.

State

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

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}. So, the serving tower ct_j^{ϑ} changes as the cellular tower of state s_{k+1}.

7.2.3 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 the connectivity parameters of $rssi_{ct_j}$, $sinr_{ct_j}$, $rsrp_{ct_j}$, $dist$, $load_{ct_j}$, $speed_i$ of $ct_j \in vc_i$ is used as the reward r_{k+1}.

We take the sum of $rssi_{ct_j}$, $sinr_{ct_j}$, $rsrp_{ct_j}$, $dist$, $load_{ct_j}$, and $speed_i$, and divide them with their number to calculate the scalar combination $\mathcal{R}_{k+1}$ to calculate reward r_{k+1} using Equation 7.5.

$$\mathcal{R}_{k+1} = (rssi_{ct_j} + sinr_{ct_j} + rsrp_{ct_j} + dist + load_{ct_j} + speed_i)/6 \tag{7.5}$$

7.2.4 Policy Function

In our proposed FD-SRL, we use well known ϵ-greedy policy of RL [30]. 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. 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 $(1 - \epsilon_k)$.

Execution of SRL

FO-SRL calculates Q-value for every tower in vc_i. The number of Q-values is equal to the number of available towers in vc_i. It ensures SRL does not consume a large amount of computation time and resources. We define the update of Q-value calculation as in Equation 7.6.

$$Q(s_k, a_k) \leftarrow Q(s_k, a_k)^{\vartheta} + \alpha[\mathcal{R}_{k+1} + \gamma rsrq_{ct_j}^{mbr} - rsrq_{ct_j}^{srv}] \tag{7.6}$$

Assume at time k, intra-VC handover has occurred, which is the switching of the serving cellular tower. At that time 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 $Q^i_1(s_{k+1}, a_{k+1}) = rsrq^{mbr_1}_{ct_1}, Q^i_2(s_{k+1}, a_{k+1}) = rsrq^{mbr_2}_{ct_2}, ..., Q^i_j(s_{k+1}, a_{k+1}) = rsrq^{mbr}_{ct_j}$.

In Equation 7.6, $rsrq^{mbr}_{ct_j}$ is the RSRQ of a cellular tower in VC and $rsrq^{srv}_{ct_j}$ is the RSRQ of the serving tower in vc_i. These values are updated in every time 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^ϑ_j, is selected as the serving tower.

The part *(ii)* of FD-SRL is discussed in Algorithm 4. The algorithm starts working from the calling of the function $AlHoMgmt(vc_i)$ in Algorithm 4. In line 3, the algorithm assigns towers of VC to the action of SRL. 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.

The overall time complexity of part *(ii)* of FD-SRL is the same as CO-SRL. FD-SRL in each time k, and the operation needs to be performed for every tower in vc_i. Thus, the overall complexity of CO-SRL is $\mathcal{O}(ct_j \in vc_i)$. The time complexity for SRL operation is $\mathcal{O}(ct_j \in vc_i * k)$.

7.3 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 [28]. 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 $60km/h$).

The time interval $k' - k$ in the FD-SRL is calibrated based on the speed of the vehicles. Vehicles traveling at higher speeds have a larger time interval compared to those traveling at lower speeds. This calibration is performed to guarantee that

Algorithm 4: Algorithm FD-SRL - Part *(ii)*

Data : vc_i, $rssi_j$, $sinr_j$, $rsrp_j$, (x, y), $dist$, $load_j$, $speed_i$
Result: ct_j^{ϑ}

```
InitializeCondition;
while vc_i ≠ ∅ do
    a_k ← vc_i;
    R_{k+1} ← (rssi_{ct_j} + sinr_{ct_j} + rsrp_{ct_j} + dist + load_{ct_j} + speed_i)/6;
    ε_k ← computePolicy();
    set α, γ; //depending on speed_i
    Q(s_k, a_k) ← Q(s_k, a_k)^ϑ + α[R_{k+1} + γ rsrq^{mbr}_{ct_j} − rsrq^{srv}_{ct_j}];
    if (arg max_{ct_j}(Q(s_k, a_k)) then
        execute switching serving tower to new ct_j^ϑ;
        Q(s_k, a_k)^ϑ ← arg max_{ct_j}(Q(s_k, a_k));
    if (connectionDrop in vc_i) then
        execute inter-VC HO to neighbouring cellular tower;
        break;
```

sufficient data is collected for training the BLSTM model of FD-SRL.

Moreover, when a vehicle moves at a faster speed, the macro cell is chosen with a slightly lower value of α and γ. Accordingly, the fast- and medium-speed vehicles choose a micro cell and pico cell with comparatively higher values of α and γ than faster speed as given in Table 8.1. This adjustment is for learning according to the dynamic behaviour of a vehicle.

Chapter 8

Performance Analysis

8.1 Simulation Scenario

We have extensively simulated our proposed approaches utilizing VEINS, OMNet++, SUMO, and Simu5G. VEINS is an open source framework for simulating vehicular networks [27]. It is included with a simulation program for vehicular networking. OMNet++ is capable of supporting and running VEINS models [33]. We have used VEINS-5.1.0 for this experiment. The simulation is built on top of VEINS. We use Simu5G for the implementation of 5G network scenarios [19]. It combines the vehicle network and the 5G network. The Simu5G models are executed along with VEINS in OMNet++. We have used Simu5G-1.2.0. Simulation of Urban Mobility (SUMO) is an open-source, highly portable road traffic simulator that can handle extensive networks of roads [15]. SUMO-1.8.0 is the simulation engine we have used in this work. Modules that exchange messages to communicate are the basis of OMNet++ models. Programming in C++ is used to create the models' functional elements. In this simulation, OMNet++-6.0 (pre11) has been used. It uses TCP sockets to talk to SUMO. The beacon messages follow the 5G communication protocol for communicating in the ITS scenario.

8.2 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 roads layout as well as highways allowing variable mobility patterns across the whole region. We have conducted simulations in two different parameter setting with five

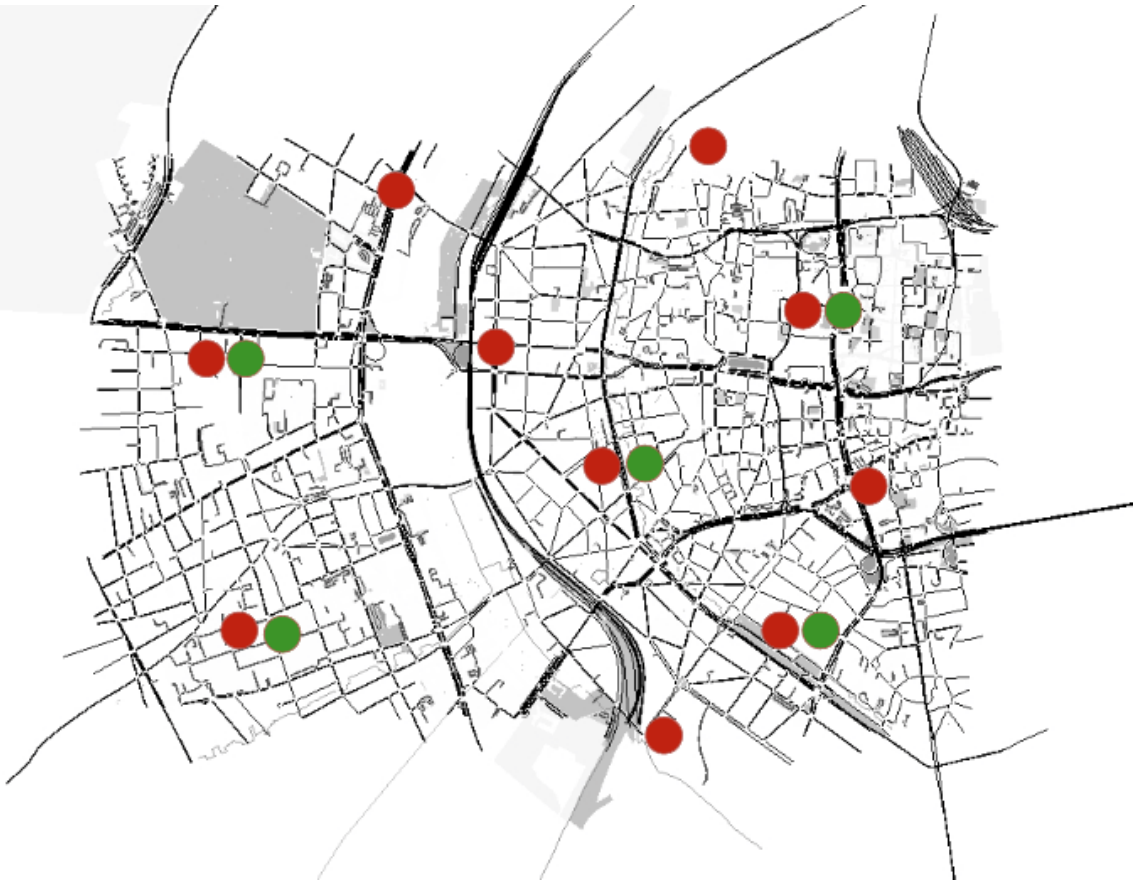


Figure 8.1: Simulation Scenario of Cologne, Germany.

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 8.1 depicts the situation of our 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.

8.3 Simulation Parameters

Our proposed approaches are evaluated using a combination of parameter settings defined in Table 8.1. The results are averages obtained from 30+ runs with different

Table 8.1: Simulation Parameters.

Parameter	**Value Range**
Simulation Area	$5 * 5km^2$
Vehicle Density	$100 - 3100$
Vehicle Speed	$0km/h - 180km/h$
Num of Cellular Tower	5 and 10
Distribution of Cellular Tower	*Random*
Comm. Range of Cellular Tower	$1000km$
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

seeds, with confidence intervals of 95%. 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): $0 - 20km/h$ to $160 - 180km/h$. We run each simulation for $100s$ of simulation time. The different densities of vehicles and speeds impact the HO in terms of connectivity.

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. We have defined the routes of vehicles and scenarios in such a way 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 8.1 represent the approximate location of the simulation with five 5G cellular towers (gNodeB), while the red dots indicate the approximate location of the simulation with ten 5G cellular towers (gNodeB).

8.4 Performance Metrics

In order to improve network stability, our research reduces the overall number of HO and average cumulative HO duration. The computing costs using virtual cell size of various approaches have also been taken into account. The effectiveness of our technique has been tested using a variety of performance parameters on vehicles with various densities and ranges of mobility.

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 [28] [23].

Though the number of intra-VC HO does not directly affect connection stability, a lower number of HOs provides 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.

Number of Inter-VC HO

The number of hard-HO or inter-VC HO represents the total number of HO performed outside VC. This is the HO event that a vehicle faces while traversing on road segments. This type of HO requires the same amount of time as standard HO. [28]. The lower the number, the better the network stability.

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 very short comparatively because every necessary information is already in the OBU of the vehicle. It is calculated by dividing the cumulative HO time by the total number of HO for intra-VC HO.

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 which includes 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 cannot complete the HO operation in a smaller time because it lacks sufficient information for inter-VC HO.

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

Percentage of Intra-VC

We calculate the percentage of intra-VC HO for different speeds. The percentages are obtained from the total number of intra-VC HO divided by the total 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.

Percentage of Inter-VC HO

For different speeds, we determine the percentage of inter-VC HO. The percentages are calculated by dividing the total number of vehicles that reached a specified speed range by the total number of inter-VC HO. This measure explains how inter-VC HO recurrence varies throughout speed ranges. Performance improves with a decreased inter-VC HO proportion.

The smaller proportion of intra-VC HO than inter-VC HO proofs approaches performs well in terms of maintaining stable connection by occurring HO inside VC. In order to deliver better results in terms of speed ranges, the percentages of intra-VC and inter-VC HO should retain stability.

Size of VC

We consider the size of VC as a performance metric. It gives a picture of the computational cost of networks. By dividing the average number of cellular towers in VC for a number of randomly chosen vehicles by the total number of vehicles, the size of VC is determined.

Increasing the number of cellular towers in a virtual cell will increase the number of connections that need to be managed by the network, which will, in turn, increase the computation required to maintain the network. This means that a larger virtual cell size can lead to more computation in the network that requires more computational resources to manage.

8.5 Compared Approaches

We have thoroughly examined numerous approaches in this thesis. The probabilistic approach, known as FiVH from the work in [28], 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 [28]
- M-FiVH: A modified probabilistic approach of FiVH [28]
- CO-SRL: Connectivity-oriented adaptive learning approach using SRL
- FD-SRL: Time series data-oriented adaptive learning approach using BLSTM and SRL

8.6 Results

We have conducted the performance analysis in two distinct parts. The first part involves a comparison of CO-SRL, M-FiVH, and FiVH using five 5G cellular towers (gNodeB). The second part involves a comparison of CO-SRL and FD-SRL using ten 5G cellular towers (gNodeB).

8.6.1 Connectivity-oriented SARSA RL

In the beginning, the performance of CO-SRL is compared with a previous work FiVH [28] and M-FiVH. The results are the average values along with 95% confidence intervals, obtained from conducting over 30 experiments with different seeds. We evaluate our work in different vehicle densities (100..3100).

Figure 8.2 represents the total number of intra-VC HO, which increases with the density of vehicles. The number of intra-VC HO for CO-SRL is 10...400 while M-FiVH and FiVH are 55...1300 and 76...1600, respectively. The graph increases almost linearly over the increasing number of densities. The slope is lower for CO-SRL than others, which shows that CO-SRL outperforms M-FiVH and FiVH in terms of reducing the number of intra-VC HO.

Figure 8.3 shows the total number of inter-VC HO, which increases linearly with the vehicle density, as expected. For CO-SRL, the increase remains almost flat for different densities, demonstrating that our designed CO-SRL handles stable connections from space to dense scenarios. The number of inter-VC HO for CO-SRL is 5...70 while 40...450 and 60...800 for M-FiVH and FiVH, respectively.

Figure 8.4 and 8.5 describes the average cumulative time required for intra-VC and inter-VC HO. Intra-VC HO requires at most $0.05ms$, $0.23ms$ and $0.30ms$ for 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 CO-SRL. The average cumulative time of inter-VC HO for high-density

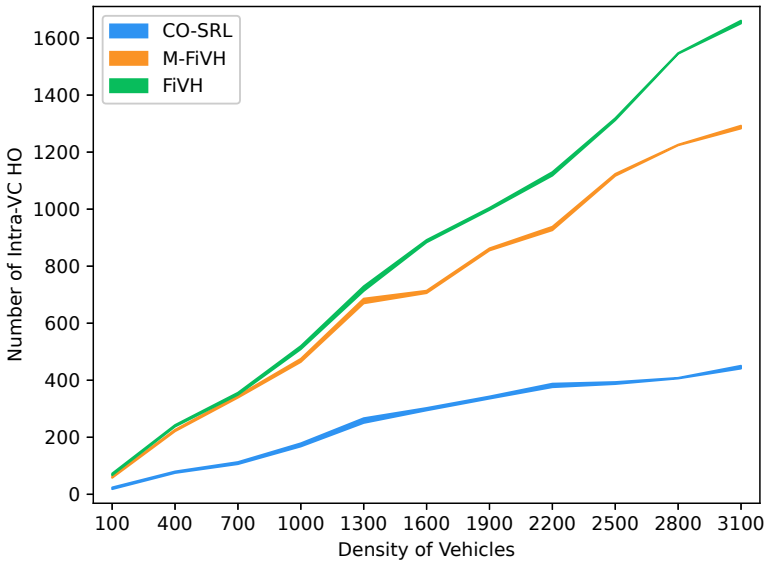


Figure 8.2: Number of intra-VC HO over the different densities of vehicles.

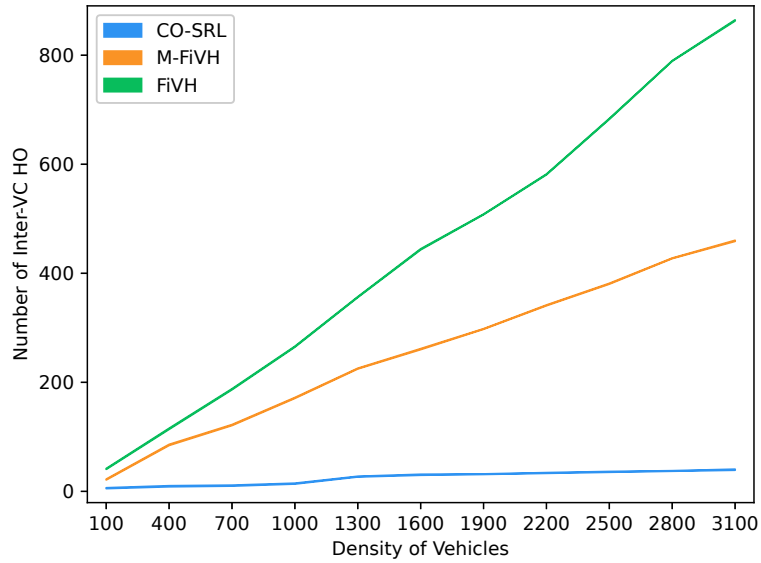


Figure 8.3: Number of inter-VC HO over the different densities of vehicles.

3100 vehicles is $7ms$, $45ms$ and $60ms$ for CO-SRL, M-FiVH and FiVH, respectively. CO-SRL requires less time than M-FiVH and FiVH for inter-VC HO. Moreover, Figure 8.4 and 8.5 also show that intra-VC HO requires less amount of time than inter-VC HO.

It should be noted that the average cumulative time for intra-VC and inter-VC HO increase with densities. We have identified some reasons behind this. With the increasing densities, there are more potential handover candidates. Higher densities

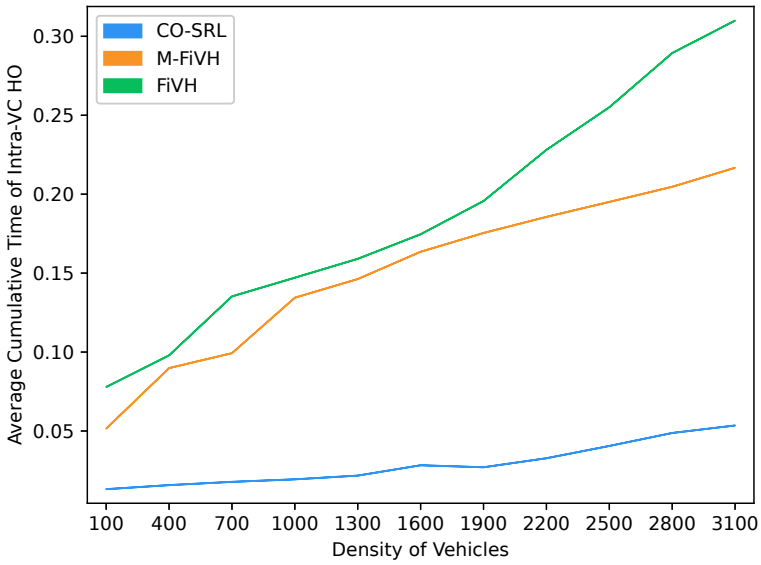


Figure 8.4: Average Cumulative intra-VC HO Time vs vehicle density.

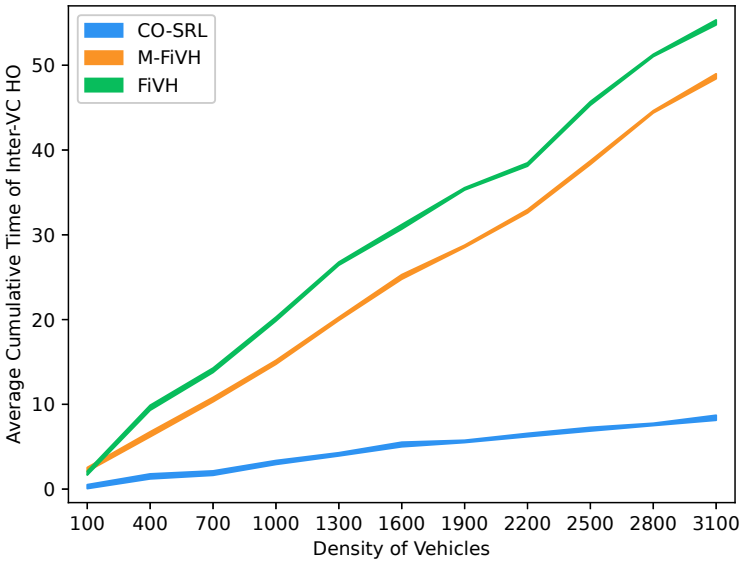


Figure 8.5: Average Cumulative inter-VC HO Time vs vehicle density.

thus lead to growth in network congestion, which directly increases latency, extending delay in completing the handover process. Additionally, more nodes can lead to increased interference, which can also slow down the handover process. This time requires more for inter-VC HO than intra-VC HO because some additional time requires for inter-VC HO as the HO occurs outside VC.

Moreover, M-FiVH showed better performance than FiVH. We can understand from Figure 8.2, 8.3, 8.4 and 8.5 that adding signal measurement RSSI as a parameter

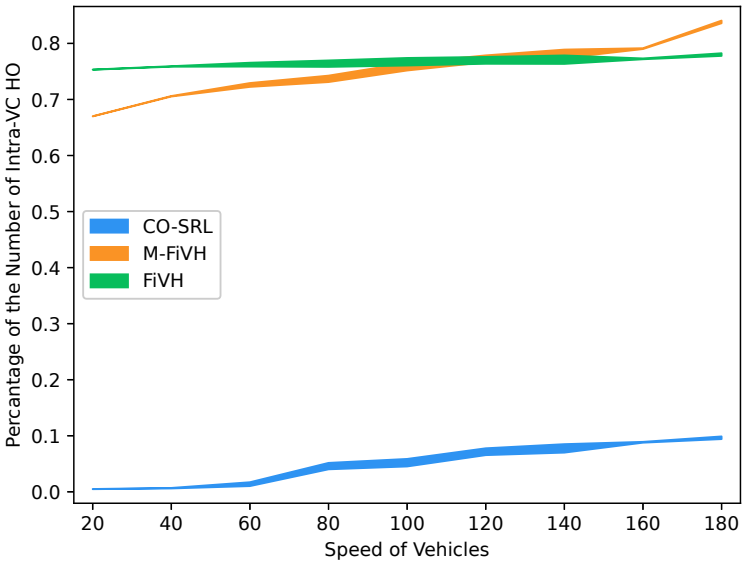


Figure 8.6: Percentages of intra-VC HO vs speed.

to FiVH improved the VC and HO management performance. Adding real-time adaption, CO-SRL, allowed the best HO decision-making compared to M-FiVH and FiVH.

Speed impacts HO decision-making, so we conducted an analysis observing different speed ranges in intervals of $20km/h$ (min-max km/h): $0 - 20km/h$ to $160 - 180km/h$. Figure 8.6 presents the percentage of the number of intra-VC HO for different speed ranges. CO-SRL has a slightly increasing pattern with respect to speed. 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 it still has a huge difference from 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 than M-FiVH.

CO-SRL remains below 0.1 for every speed range while M-FiVH and FiVH have $0.65-0.85$. The percentage of the number of inter-VC HO is shown in Figure 8.7. CO-SRL remains close to 0.1, and M-FiVH and FiVH have linearly increasing percentages of the number of intra-VC HO, showing that CO-SRL outperforms M-FiVH and FiVH, where high mobility does not affect the number of HOs.

The VC size serves as a measurement of the computational costs of the employed

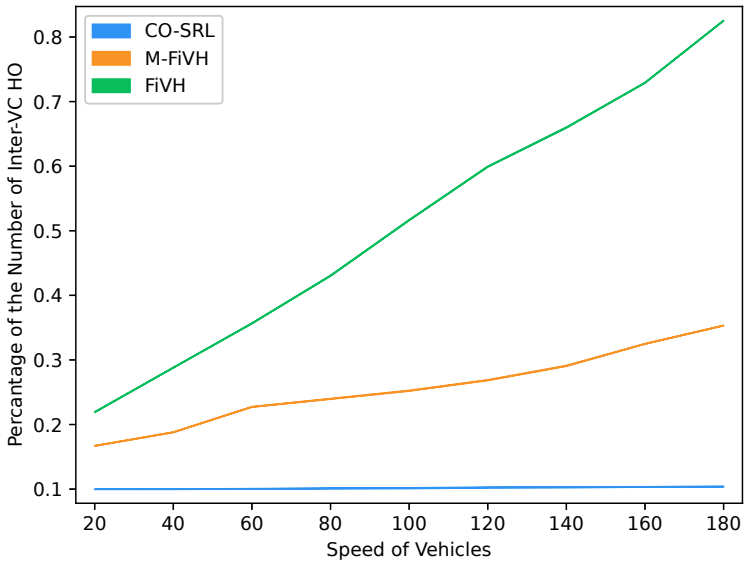


Figure 8.7: Percentages of inter-VC HO vs speed.

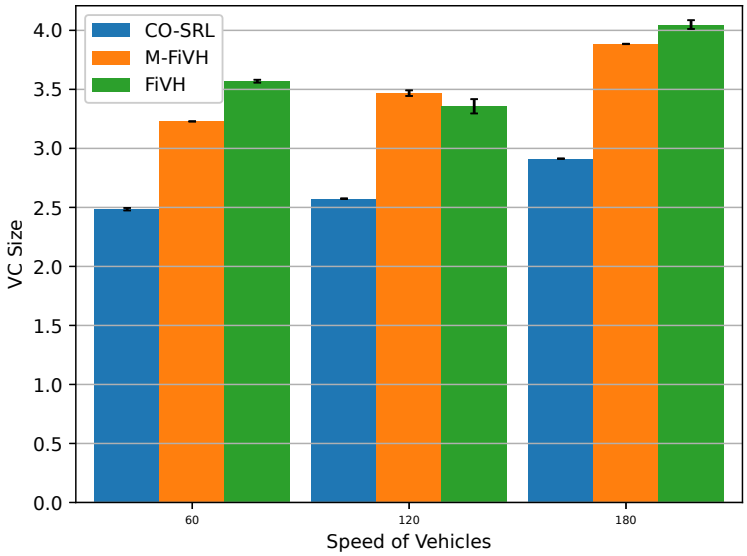


Figure 8.8: The size of VC over speed.

approaches. A hundred vehicles were monitored at three different speeds in order to determine the VC size. In Figure 8.8, it can be found that CO-SRL has a smaller VC size of 2.5...2.7 while M-FiVH and FiVH have a larger size of 3.1...3.8 and 3.3...4.1. This is because CO-SRL maintains a lower computational cost than M-FiVH and FiVH since it installs a little bit fewer towers.

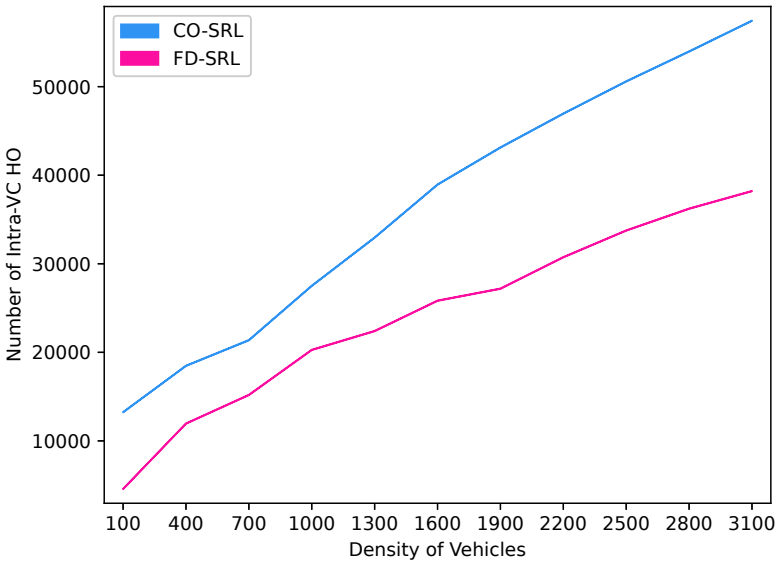


Figure 8.9: Number of intra-VC HO over the different densities of vehicles.

8.6.2 Factor-distinct SARSA RL

We have evaluated our proposed CO-SRL with our other contribution, FD-SRL, in this study. We consider a new, more complex scenario in which more HO may occur. The vehicle densities in which we assess our work range from (100..3100). The findings are averages with 95% confidence intervals that were acquired from more than 30 runs using various seeds.

The total number of intra-VC HO, which rises with vehicle density, is shown in Figure 8.9. For FD-SRL, there are 2000...36000 intra-VC HO, compared to 12000...57000 for CO-SRL. As the number of densities increases, the graph climbs approximately linearly. 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.

The total number of inter-VC HO is shown in Figure 8.10. The increase for FD-SRL and CO-SRL is linear over the increase of vehicle density. Despite growing linearly, they continue to exist in smaller quantities. There are situations when performing HO is necessary for improved user experience. While the number of inter-VC HO for CO-SRL is 1800...12000, for FD-SRL is 200...8300 HOs.

Figure 8.11 and 8.12 provide a summary of the average cumulative time needed for intra-VC and inter-VC HO. Intra-VC HO fluctuates between 0.160...0.200ms for FD-SRL. The average cumulative time increases slightly 0.275...0.350ms for CO-SRL with the increasing number of densities. The average cumulative time of inter-VC HO is 0.50...0.55ms for CO-SRL, while FD-SRL requires less time, 0.05...0.40ms.

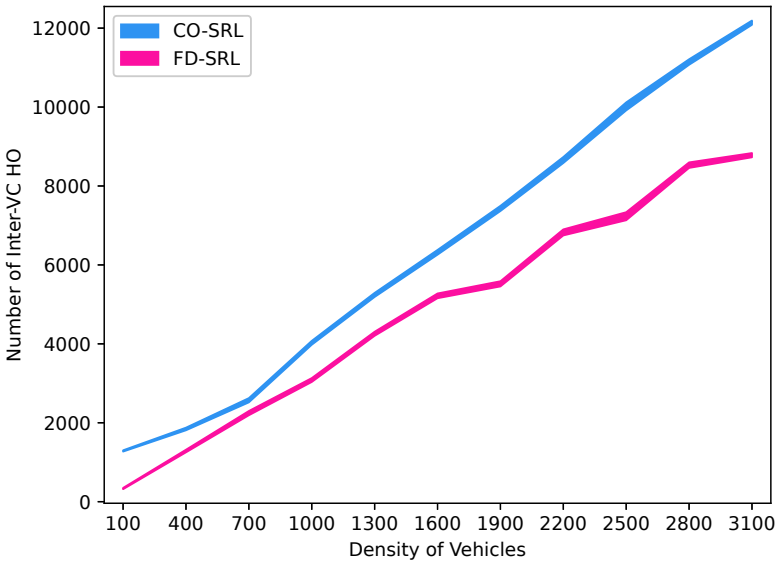


Figure 8.10: Number of inter-VC HO over the different densities of vehicles.

Moreover, Figure 8.11 and 8.12 also show that intra-VC HO requires less amount of time than inter-VC HO. The average cumulative time for CO-SRL varies from our previous analysis. The cause is due to altered simulation parameter settings and environmental variations. Although it differs from the previous analysis, the change is the same.

It should be noted that the average cumulative time for intra-VC and inter-VC HO increase with densities. We have identified some reasons behind this. With the increasing densities, there are more potential handover candidates. Higher densities thus lead to growth in network congestion, which directly increases latency, extending delay in completing the handover process. Additionally, more nodes can lead to increased interference, which can also slow down the handover process. This time requires more for inter-VC HO than intra-VC HO because some additional time requires for inter-VC HO as the HO occurs outside VC.

For various speed ranges, the percentage of the number of intra-VC HO is shown in Figure 8.13. In terms of speed, FD-SRL and CO-SRL exhibit an increasing pattern. It is expected because high-mobility vehicles must carry out more intra-VC HO in order to keep connections steady. The FD-SRL varies slightly more since the situations in which it learns might change. In spite of this, FD-SRL outperforms CO-SRL.

The percentage of the number of inter-VC HO is shown in Figure 8.14. FD-SRL remains close to $0.28...0.32ms$, and CO-SRL increases with fluctuations between $0.36...0.45ms$ percentages of the number of intra-VC HO. It proves that FD-SRL

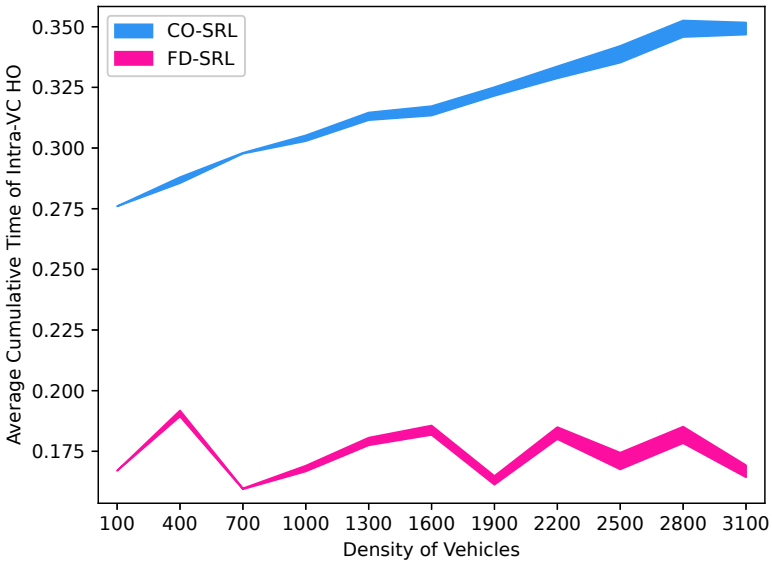


Figure 8.11: Average Cumulative intra-VC HO Time vs vehicle density.

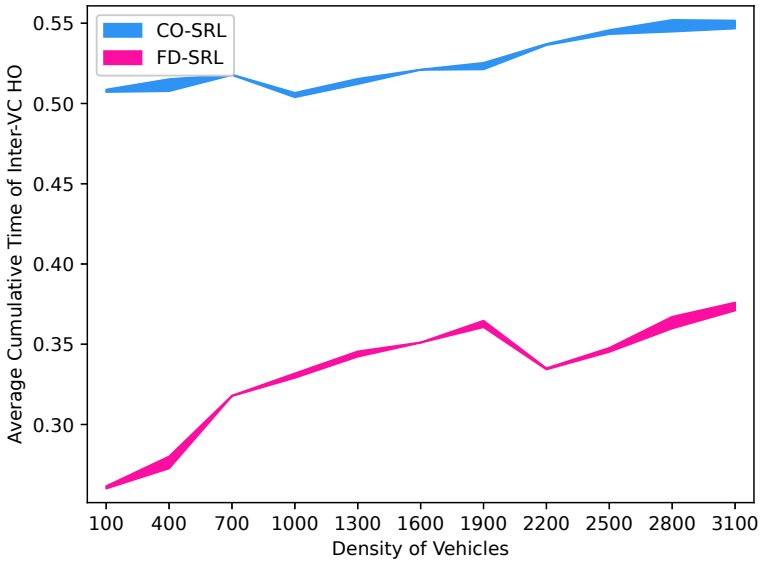


Figure 8.12: Average Cumulative inter-VC HO Time vs vehicle density.

performs better than CO-SRL, where high mobility does not affect the number of HOs.

We can see that though FD-SRL performs better than CO-SRL, there are more fluctuations in FD-SRL than CO-SRL. The LSTM model in FD-SRL is designed to learn long-term dependencies in sequences of data while the SRL algorithm in CO-SRL is designed to learn a policy for making decisions. Moreover, LSTMs are sensitive to the amount of data, with the variation of captured data the LSTM model

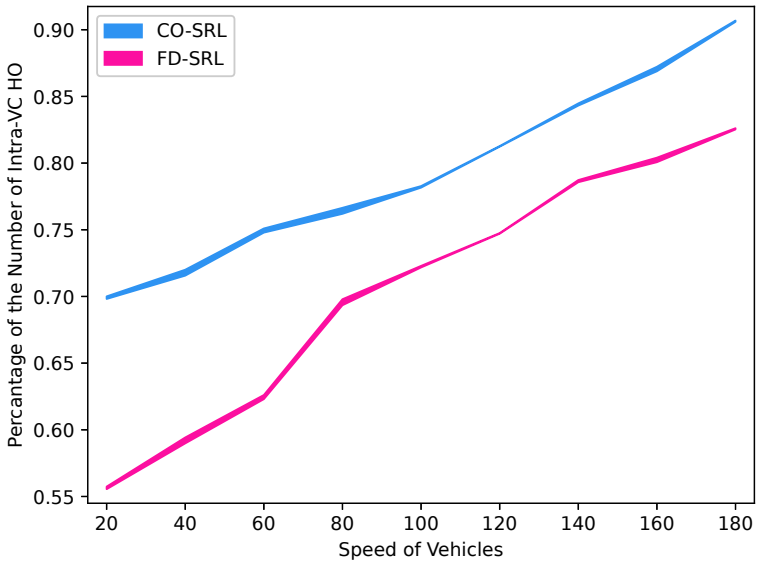


Figure 8.13: Percentages of intra-VC HO vs speed.

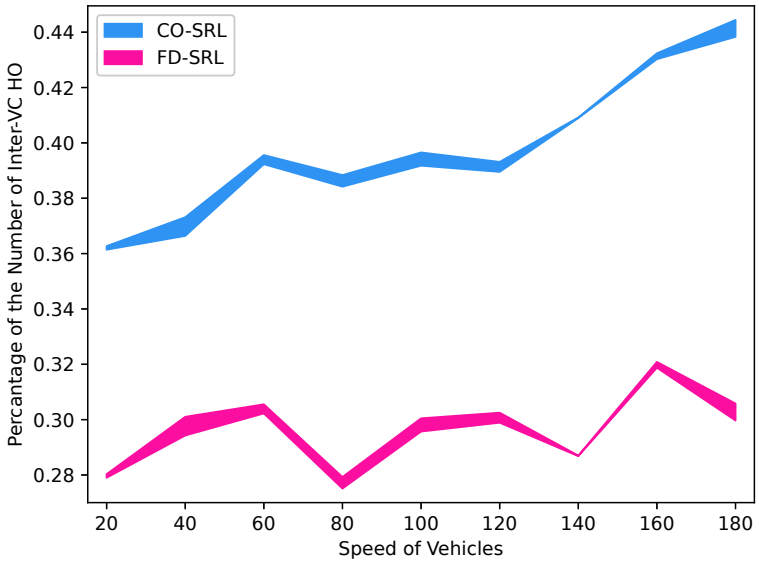


Figure 8.14: Percentages of inter-VC HO vs speed.

can perform differently and fluctuate more than SRL.

The computational cost of the implemented techniques is measured by the VC size. To determine the VC size, a hundred vehicles were observed at three different ranges of speed. Figure 8.15 shows that CO-SRL is discovered to have a smaller VC size of 3.2...4 than FD-SRL of 4.2...5.1. This is because 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

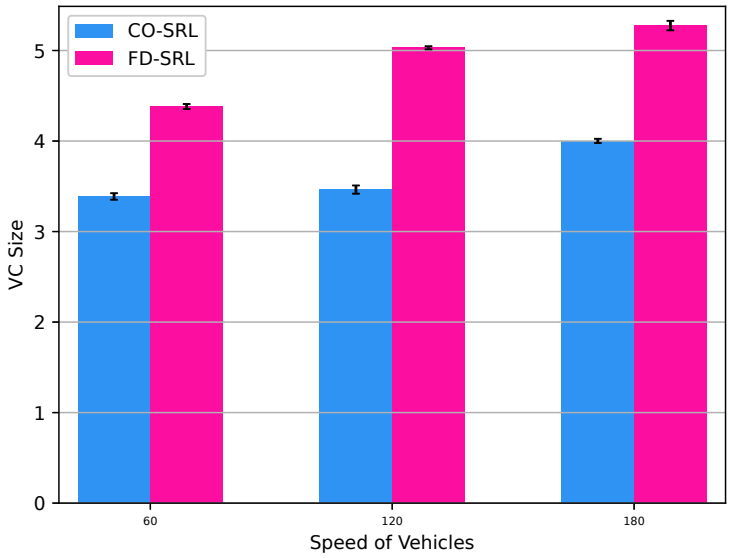


Figure 8.15: The size of VC over speed.

requires more computation cost than CO-SRL.

8.7 Remarks

Our proposed approaches, M-FiVH, CO-SRL, and FD-SRL, outperform FiVH, according to our analysis of the analysis of findings. With various tactics, the performance gradually becomes better. It is noteworthy that FD-SRL and CO-SRL produce the greatest outcomes. In terms of the number of handovers, average cumulative handover time, and percentages of handovers, FD-SRL outperforms CO-SRL, while CO-SRL has a lower computational cost than FD-SRL. Our analysis demonstrates that connection stability is maintained for high-mobility and ultra-density vehicular networks in both highway and urban scenarios.

Chapter 9

Conclusion

9.1 Summary

This thesis addressed the vehicular network stability problem in 5G vehicular networks for ultra-dense networks and high mobility of vehicles. As our several contributions to connection stability, we devised a number of strategies. We started our contribution by modifying a previous work FiVH. We went through several studies and experiments in terms of parameter settings and algorithm design. This version of M-FiVH performed well by adding signal parameters.

Our second contribution is the introduction of CO-SRL, an adaptive SRL strategy for lowering the number of HO and average cumulative HO duration. To predict vehicle direction, we used an LR model, and to update CFs, we used a dynamic adaptive learning technique based on SRL. A stable connection is ensured together with the virtual cell and handover management using SRL.

By presenting a time-series data-oriented BLSTM learning with adaptive SRL, we made another substantial contribution to this thesis. Time series data have been applied for managing virtual cells. For handover management, we kept SRL. In this section, we explored how the promise of connection stability is enhanced when time-series data-oriented learning is combined with adaptive learning.

In a series of simulations, we examined several scenarios and offered insights. We conducted simulations with different parameter settings. We considered extensive scenarios for the analysis of our proposed approaches for the connection stability problem.

The results of the conducted studies have shown that both CO-SRL and FD-SRL ensured stable connectivity of networks while reducing HO overhead, outperforming FiVH and M-FiVH approaches. In terms of reducing HO and average cumulative HO

time, FD-SRL outperforms CO-SRL but increases computation cost. Our proposed approaches enhance connection stability in each of the proposed strategies, M-FiVH, CO-SRL, and FD-SRL.

Limitations

Even though the proposed approaches have produced encouraging outcomes for connection stability, there are still a number of areas that we have identified as limitations that need to be explored.

- We kept the Linear Regression (LR) model for predicting direction simple. We did not consider model performance and analyze the behaviour of the algorithm with wrong predictions.
- We did not examine alternative learning rates and discount factors for the SRL algorithm. Varying these values may affect the results and lead to improved performance.
- We did not evaluate the BLSTM model with various design configurations and parameter settings. Furthermore, the generation of the BLSTM model is contingent on a specific time interval. We did not consider the real-time parallel generation of the BLSTM model alongside the simulation.
- We used Mean Squared Error (MSE) as a performance metric for the BLSTM model. However, MSE alone does not provide a complete and accurate evaluation of the model.

9.2 Future Discussion

Our proposed approaches demonstrate an improvement in the connection stability issue. We have identified some limitations of our work and offer potential avenues for overcoming them. Additionally, we suggest directions that can further enhance performance.

Overall

- We will study our approach in multiple scenarios with different maps of several different routes of vehicles.

- We will analyze the simulation by placing 5G cellular towers in different places. Moreover, we will explore more extensive scenarios with different numbers of cellular towers.
- We will also study other time series data-oriented-based learning and adaptive learning strategies. An analysis of computational cost will be taken into action in our future studies.

CO-SRL

- We will investigate different learning rates and discount factors.
- The update of Connectivity Factors (CFs) will be examined with direct adaptive learning.
- We will add a model evaluation for LR for error calculation and ensure better performance.

FD-SRL

- The BLSTM model will be examined with different parameter settings, including different epochs, number of neurons, and batch size.
- The BLSTM model will be generated in real-time at each time step, alongside the simulation. We want to examine the result with such a design for improving performance.
- Loss calculation will be taken into action more precisely in our future work. We will evaluate our trained model with various error calculation methods to ensure model performance.

Bibliography

[1] Vivek Agarwal, Chandra Sharma, Rajaneesh Shetty, Anil Jangam, and Rajiv Asati. A journey towards a converged 5g architecture & beyond. In *Proceedings of the IEEE 5G World Forum (5GWF)*, pages 18–23, 2021.

[2] Noura Aljeri and Azzedine Boukerche. An efficient handover trigger scheme for vehicular networks using recurrent neural networks. In *Proceedings of the ACM International Symposium on QoS and Security for Wireless and Mobile Networks*, pages 85–91, 2019.

[3] Noura Aljeri and Azzedine Boukerche. A two-tier machine learning-based handover management scheme for intelligent vehicular networks. *Elsevier Ad Hoc Networks*, 94:101930, 2019.

[4] Noura Aljeri and Azzedine Boukerche. Mobility management in 5g-enabled vehicular networks: Models, protocols, and classification. *ACM Computing Surveys*, 53(5):1–35, 2020.

[5] Ansif Arooj, Muhammad Shoaib Farooq, Aftab Akram, Razi Iqbal, Ashutosh Sharma, and Gaurav Dhiman. Big data processing and analysis in internet of vehicles: architecture, taxonomy, and open research challenges. *Springer Archives of Computational Methods in Engineering*, 29:793–829, 2022.

[6] Aaron Ayub, Thiago S. Gomides, and Robson E. De Grande. Edge-based V2X efficient traffic emergency responding protocol. In Mirela Notare and Peng Sun, editors, *Proceedings of the ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, pages 57–64, 2021.

[7] Azzedine Boukerche and Robson E. De Grande. Vehicular cloud computing: Architectures, applications, and mobility. *Elsevier Computer Networks*, 135:171–189, 2018.

[8] Murtaza Cicioğlu. Performance analysis of handover management in 5g small cells. *Elsevier Computer Standards & Interfaces*, 75:103502, 2021.

[9] Felipe Cunha, Guilherme Maia, Heitor S. Ramos, Bruno Perreira, Clayson Celes, André Campolina, Paulo Rettore, Daniel Guidoni, Fernanda Sumika, Leandro Villas, Raquel Mini, and Antonio Loureiro. *Emerging Wireless Communication and Network Technologies: Principle, Paradigm and Performance*, chapter Vehicular Networks to Intelligent Transportation Systems, pages 297–315. Springer, 2018.

[10] Amina Gharsallah, Faouzi Zarai, and Mahmoud Neji. Exploring software-defined network (sdn) for seamless handovers in future vehicular networks: Mobility management approach. *International Journal of Smart Security Technologies (IJSST)*, 9(1):1–16, 2022.

[11] Thiago S. Gomides, Robson E. De Grande, Allan Mariano de Souza, Fernanda S. H. Souza, Leandro A. Villas, and Daniel L. Guidoni. An adaptive and distributed traffic management system using vehicular ad-hoc networks. *Elsevier Computer Communications*, 159:317–330, 2020.

[12] Thiago S. Gomides, Robson E. De Grande, Rodolfo I. Meneguette, Fernanda S.H de Souza, and Daniel L. Guidoni. Predictive congestion control based on collaborative information sharing for vehicular ad hoc networks. *Elsevier Computer Networks*, 211:108955, 2022.

[13] Rodolfo I. Meneguette, Robson E. De Grande, and Antonio A. F. Loureiro. Intelligent transportation systems. In *Intelligent Transport System in Smart Cities: Aspects and Challenges of Vehicular Networks and Cloud*, pages 1–21. Springer International Publishing, Cham, 1 edition, 2018.

[14] Gaganpreet Kaur, Raman Kumar Goyal, and Rajesh Mehta. An efficient handover mechanism for 5g networks using hybridization of lstm and svm. *Springer Multimedia Tools and Applications*, 81:37057–37085, 2022.

[15] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *Proceedings of the IEEE international conference on intelligent transportation systems (ITSC)*, pages 2575–2582, 2018.

[16] Zaigham Mahmood. *Connected Vehicles in the Internet of Things: Concepts, Technologies and Frameworks for the IoV*, chapter Connected Vehicles in the IoV: Concepts, Technologies and Architectures, pages 3–18. Springer International Publishing, Cham, 2020.

[17] Na Meng, Hongtao Zhang, and Bo Lin. User-centric mobility management based on virtual cell in ultra-dense networks. In *Proceedings of the IEEE/CIC International Conference on Communications in China (ICCC)*, pages 1–6, 2016.

[18] Nadia Mouawad, Rola Naja, and Samir Tohme. User-centric vs network-centric vertical handover algorithms in 5g vehicular networks. In *Proceedings of the Springer International Conference on Internet of Vehicles*, pages 46–59, 2018.

[19] Giovanni Nardini, Dario Sabella, Giovanni Stea, Purvi Thakkar, and Antonio Virdis. Simu5g–an omnet++ library for end-to-end performance evaluation of 5g networks. *IEEE Access*, 8:181176–181191, 2020.

[20] Yong Niu, Yong Li, Depeng Jin, Li Su, and Athanasios V Vasilakos. A survey of millimeter wave communications (mmwave) for 5g: opportunities and challenges. *Springer Wireless networks*, 21:2657–2676, 2015.

[21] Haider Rizvi and Junaid Akram. Handover management in 5g software defined network based v2x communication. In *Proceedings of the IEEE International Conference on Open Source Systems and Technologies (ICOSST)*, pages 22–26, 2018.

[22] Taylan Şahin, Markus Klügel, Chan Zhou, and Wolfgang Kellerer. Multi-user-centric virtual cell operation for v2x communications in 5g networks. In *Proceedings of the IEEE Conference on Standards for Communications and Networking (CSCN)*, pages 84–90, 2017.

[23] Taylan Sahin, Markus Klugel, Chan Zhou, and Wolfgang Kellerer. Virtual cells for 5g v2x communications. *IEEE Communications Standards Magazine*, 2(1):22–28, 2018.

[24] Syed Danial Ali Shah, Mark A Gregory, Shuo Li, Ramon Fontes, and Ling Hou. Sdn-based service mobility management in mec-enabled 5g and beyond vehicular networks. *IEEE Internet of Things Journal*, 9(15):13425–13442, 2022.

[25] Vishal Sharma, Fei Song, Ilsun You, and Han-Chieh Chao. Efficient management and fast handovers in software defined wireless networks using uavs. *IEEE Network*, 31(6):78–85, 2017.

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

[27] Christoph Sommer, David Eckhoff, Alexander Brummer, Dominik S Buse, Florian Hagenauer, Stefan Joerer, and Michele Segata. Veins: The open source vehicular network simulation framework. In *Springer Recent Advances in Network Simulation*, pages 215–252. Springer, 2019.

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

[29] Carlos Renato Storck, Guilherme G. de O. Silva, and Fátima Duarte-Figueiredo. A vehicle-centric probabilistic approach to virtual cell management in ultra-dense 5g networks. In *Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*, pages 1–7, 2020.

[30] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[31] Livinus Tuyisenge, Marwane Ayaida, Samir Tohme, and Lissan-Eddine 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*, pages 1–64. IGI Global, 2020.

[32] Mehmet Fatih Tuysuz and Murat Ucan. Energy-aware network/interface selection and handover application for android-based mobile devices. *Elsevier Computer Networks*, 113:17–28, 2017.

[33] András Varga and Rudolf Hornig. An overview of the omnet++ simulation environment. In *Proceedings of the International Conference on Simulation Tools and Techniques for Communications (ICST), Networks and Systems*, page 10, 2010.

[34] Jin Wu, Jing Liu, Zhangpeng Huang, and Shuqiang Zheng. Dynamic fuzzy q-learning for handover parameters optimization in 5g multi-tier networks. In *Proceedings of the IEEE International Conference on Wireless Communications & Signal Processing (WCSP)*, pages 1–5, 2015.

[35] Hailin Xiao, Xue Zhang, Anthony Theodore Chronopoulos, Zhongshan Zhang, Hailong Liu, and Shan Ouyang. Resource management for multi-user-centric v2x communication in dynamic virtual-cell-based ultra-dense networks. *IEEE Transaction on Communications*, 68(10):6346–6358, 2020.

[36] Zhou Yu, Vikram Ramanarayanan, David Suendermann-Oeft, Xinhao Wang, Klaus Zechner, Lei Chen, Jidong Tao, Aliaksei Ivanou, and Yao Qian. Using bidirectional lstm recurrent neural networks to learn high-level abstractions of sequential features for automated scoring of non-native spontaneous speech. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 338–345, 2015.

[37] Shizhe Zang, Wei Bao, Phee Lep Yeoh, Branka Vucetic, and Yonghui Li. Managing vertical handovers in millimeter wave heterogeneous networks. *IEEE Transactions on Communications*, 67(2):1629–1644, 2018.

[38] Shubo Zhang, Tianyang Wu, Maolin Pan, Chaomeng Zhang, and Yang Yu. A-sarsa: A predictive container auto-scaling algorithm based on reinforcement learning. In *Proceedings of the IEEE International Conference on Web Services*, pages 489–497, 2020.

[39] Yingxiao Zhang and Ying Jun Zhang. User-centric virtual cell design for cloud radio access networks. In *Proceedings of the IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 249–253, 2014.

[40] Ying Zhou and Yuexia Zhang. User-centric data communication service strategy for 5g vehicular networks. *IET Communications*, 16(10):1041–1056, 2022.