

Container Based Nomadic Vehicular Cloud Using Cell Transmission Model

Devakirubai Navulkumar^{1,*} and Menakadevi Thangavelu²

¹Department of Computer Science and Engineering, Jayam College of Engineering and Technology, Dharmapuri, 636813, India

²Department of Electronics and Communication Engineering, Adhiyaman College of Engineering, Hosur, 635109, India

*Corresponding Author: Devakirubai Navulkumar. Email: flytondk@gmail.com

Received: 14 November 2021; Accepted: 07 January 2022

Abstract: Nomadic Vehicular Cloud (NVC) is envisaged in this work. The predominant aspects of NVC is, it moves along with the vehicle that initiates it and functions only with the resources of moving vehicles on the heavy traffic road without relying on any of the static infrastructure and NVC decides the initiation time of container migration using cell transmission model (CTM). Containers are used in the place of Virtual Machines (VM), as containers' features are very apt to NVC's dynamic environment. The specifications of 5G NR V2X PC5 interface are applied to NVC, for the feature of not relying on the network coverage. Nowadays, the peak traffic on the road and the bottlenecks due to it are inevitable, which are seen here as the benefits for VC in terms of resource availability and residual in-network time. The speed range of high-end vehicles poses the issue of dis-connectivity among VC participants, that results the container migration failure. As the entire VC participants are on the move, to maintain proximity of the containers hosted by them, estimating their movements plays a vital role. To infer the vehicle movements on the road stretch and initiate the container migration prior enough to avoid the migration failure due to vehicles dynamicity, this paper proposes to apply the CTM to the container based and 5G NR V2X enabled NVC. The simulation results show that there is a significant increase in the success rate of vehicular cloud in terms of successful container migrations.

Keywords: Vehicular cloud; container migration; cell transmission model; 5G NR V2X PC5 interface

1 Introduction

Static [1–3] and dynamic [4,5] VC are classified according to the movement of the vehicles participating in VC. Static VC is so named because the vehicles involved are 'static' in one location (like a parking lot). In practice, even static vehicles may leave their location at any moment, necessitating the same resource volatility management as dynamic VC [1]. The sole distinction is that dynamic VC has a very high level of resource volatility, while static VC has a very low level of resource volatility. As a result, this work considers both static and dynamic VC in terms of VC movement. Static VCs operate in a specific place, such as around a base station, with moving or parked cars, or both, while dynamic VCs operate with moving vehicles on the road and move in parallel with the VC initiator. It's worth mentioning that, while



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

studies suggest VC based on vehicle movement, there are only a few research that address VC moving along with a vehicle. The objective of this work is to create a 5G NR V2X PC5 enabled VC that only uses vehicle resources and it move along with a vehicle that initiates the VC.

Establishing and sustaining Vehicular Cloud (VC) [4–7] is the demanding step of VC implementation. Owing to the rapid mobility design of the vehicles, the connection between the VC members is very fluctuating and in turn, the efficiency of VC is reduced. This work focus to build a reliable NVC with the maximum stabilised topology. To accomplish this, the proposed VC is built only with the resources of moving vehicles on the highway without using any additional infrastructures and an efficient mechanism is constructed to determine the container (task) migration initiation time within VC. The motivational features of 5G NR V2X [8] are the reason for choosing NR V2X for NVC implementation. The new transmission types, unicast and groupcast help to communicate within NVC and finally the advanced security features justify the use of NR V2X for NVC. 5G New Radio Vehicle-to-Everything (NR V2X) Mode 2(a) is used in this work because of its enhanced features than the Cellular V2X (C-V2X) mode 4 [9], as it is specially designed for V2V communications using PC5 sidelink interface without relying on the cellular infrastructure support.

Cooperative Awareness Message (CAM) [10] and Decentralized Environmental Notification Message (DENM) [10] are the two message forms that floats between vehicles using direct communication mode. In the proposed work, to avoid transacting additional messages, CAM and DENM are used to create NVC. VM and its migration are the baseline of vehicular cloud. Each service, like Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and software as a service (SaaS) has a different focus, but all have VM migration as an underlying operation [11]. This work uses Containers in the place of VMs. Containers are more light weight and use fewer resources than VMs [12]. Containers have too short life cycle than VMs and more dynamic in use. It takes less time to implement, operate, and monitor. Containers are much more mobile than VMs because they can launch and run applications within seconds. Similar to the reason for VM migration, container migration is necessary to face the dynamic nature of VC. Container migration has a significant role in VC and also it is the major challenge for VC. The main contribution of this work is to address the unsuccessful container migrations due to dual mobility of VC participants by applying CTM [13] to the proposed NVC. Therefore, it is intended to mitigate the failure rate of NVC due to container migration failure to increase the reliability of NVC.

Distance and coverage area are the two factors that leads a vehicle to discontinue its participation from the vehicular cloud. When the distance between the VC_I and VC_M increases, the PC5 interface connection begins to fade out and the undergoing process gets interrupted. To preserve the proximity between VC_I and VC_M , it is important to determine when and where to migrate the container in a VC_M whose distance from the VC_I is expected to be above the limit. This decision on container migration is facilitated by applying CTM. CTM is used to simulate the actual traffic state of a road. This model estimates the position and number of vehicles in each section of the road. As the traffic conditions are simulated the motions of VC_I and VC_M s can be predicted. So prior decision making about the containers of the exiting VC_M can be attained.

The rest of the paper is organized as follows. Section 2 discusses about the concepts of vehicular cloud, sidelink interface, container migration and CTM. Nomadic Vehicular Cloud is introduced in Section 3. The performance of NVC is evaluated using simulation in Section 4 and concluded in Section 5.

2 Related Work

Vehicular Cloud is intriguing due to the unused resources in contemporary automobiles. Mobility of vehicles involving in VC, communication model applied to VC, the choice of VM or containers, vehicle residency time within VC are the major factors that decides the performance of VC. The primary aspect of VC, as well as its primary obstacle, is the mobility of vehicles. As a result, comparatively, few articles

on VCs focused on the parked automobiles [1–3] and few others experimented into datacenters made from automobiles parked at a large parking lots. The distinguishing between static and dynamic VC is made primarily on resource volatility caused by vehicle speed rather than network connectivity [5].

Dynamic VC [5], which works with moving cars on highways and access points that have been pre-installed. Access points are linked to the VC controller. The extension of [5] includes job migration, whereas each job is encapsulated as container images [4]. References [6,14] proposes VC with wireless base stations as data hosts. Wireless base stations [6] act as the initiator of VC and VC is static in the area around the wireless base station. Vehicles entering the region participate in the VC with their own choice and interest. The ongoing workload of the vehicles departing from the region has been migrated to other vehicles within the region. Conventional edge clouds [14] offload computing processes to VC, proposed Cooperative task scheduling scheme for VC and developed a vehicular mobility model to increase the vehicular residency time of the vehicles involving in VC. Connected vehicular cloud computing (CVCC) [15] technology is a hybrid VC that exploited the resources in the internet cloud, RSUs and vehicles. The internet cloud and RSUs serve as the main computation service provider for vehicles. Out sourcing concept is introduced in CVCC, the vehicle that receives the task can outsource sub-task to other vehicles.

Reference [16] explored with ways to improve VC's reliability and availability in terms of vehicle residency time. To extend the vehicle residency duration, they developed redundancy-based work assignment methodologies. Their method extends the duration of a vehicle's stay, which has the unintended consequence of increasing resource use. Vehicle residency time is treated as the cell residency time [14] of vehicles since they are in the VC as long as they are inside the coverage of the base station and simulated using the Hyper-Erlang distribution [17].

The communication model applied to the VC have an impact on the performance of the VC. In Reference [4,5,16], DSRC (Dedicated Short Range Communication) is applied to VC. When compared to the 5G NR V2X PC5 interface, DSRC has a lower data transfer rate, shorter communication range [18] and lower Packet Reception Ratio [19]. Reference [20] discuss about the PC5 QoS model for NR sidelink, group cast and unicast V2X communication connection establishment and the physical layer design of the NR side link as per 3GPP Release 16. The evolution of V2X standards [21–25] towards IEEE 802.11bd paves way to double the IEEE 802.11p communication range, up to 1000 m. The 5G NR V2X's goal is to support extended sensor applications that require high bandwidth to communicate huge quantity of data [20].

The efficient migration of VM or containers plays a key role in VC, as in the traditional cloud. The initiation time of the container migration process is one aspect that determines the performance of the migration process. Reference [6] handled this problem in the term 'first hop' failure. The traffic density in the chosen scenario is an another important factor of VC. Higher traffic density and logically partitioning the road helps the migration process to be successful [13,22]. The length of the road segment is measured on the basis of the length of the vehicle travelling for a given time [13]. Tab. 1 compares the related work. CTM [13] is used in this work to decide the time for container migration on the basis of their concept of road segmentation. Basically, by measuring the traffic density and traffic flow at specific locations at different stages of time, CTM predicts the traffic attitude on the identified road stretch. A key requirement for VC to avoid dis-connectivity among its participants is to estimate the spatio-temporal data of vehicles. By applying CTM to VC, the movements of vehicles can be calculated and the start time of the container migration can be calculated using it. To the best of our understanding, this is the first effort to apply CTM concept in VC to determine the time to initiate the container migration.

Table 1: Comparison of similar works

Reference	Scenario: Highway	VC mobility: Dynamic	Task portability: Containers	Task migration: Container migration	Resources: Only moving vehicles	VC server: Vehicle	VC client: Vehicle	Communication model: NR V2X
2	✓	✗	✓	✓	✗	✓	✓	✗
14	✗	✗	✗	✗	✗	✗	✓	✗
15	✗	✗	✗	✗	✗	✗	✓	✗
16	✗	✗	✗	✗	✗	✗	✓	✗
NVC	✓	✓	✓	✓	✓	✓	✓	✓

3 Proposed Model

A container based Nomadic Vehicular Cloud (NVC) using CTM is proposed here. The moving vehicles on the road are NVC participants. In contrast with [2,6], NVC is initiated by one of the moving vehicles on the road. The NVC region moves along with the vehicle that initiated the NVC. Vehicles moving around the initiator are selected as VC members.

3.1 System Model

3.1.1 Establishing Nomadic Vehicular Cloud

NVC uses 5G V2X PC5 interface to build the vehicular cloud. NVC communicates in the range of 1000 m [23]. VC_I , searches for the potential moving vehicles that can act as a VC_M . VC_I , through CAM beacons, broadcasts the VC member request to the single hop vehicles. Vehicles that receive the CAM beacons, if it has the requested resources with the required resource capacity and time, triggers DENM transmission that consists of the information about the resources and its capacity. VC_I forms a resource profile registry from the collected DENMs. VC_I selects the VC_M s from the resource profile registry to create the Vehicular Cloud. By using periodic CAMs and event-triggered DENMs, the resource profile registry is periodically updated. When one of the following events happen in the environment, the resource profile registry is modified using CAMs even after the VC is established: (i) a new vehicle enters into the range, (ii) one of the VC_M s leaves the range and (iii) when any of the VC_M complete its task while it is within the range.

3.1.2 NVC in Cell Transmission Model

Heavy traffic environment of the road is considered and exploited for this work. The peak traffic provides the opportunity to utilize the resources of many vehicles, but the mobility of vehicles tends to raise the problem of dis connectivity between NVC participants. This issue is handled with the aid of CTM. Prior estimation of vehicle movement and initiating the container migration process at the earliest are the aim of applying CTM to NVC. The pre informed container migration initiation, increases the success rate of container migration across NVC. The container migration time remains same [24], as it is assumed that the containers' size are fixed and the bandwidth are same between VC_M s. In the proposed work, each VCM receives containers in different size and hence the container migration time differs. The rate of container migration requirement to maintain the VC topology, found very low with the combination of establishing NVC with CTM in a heavy traffic road.

3.1.3 Container Migration in NVC

Container migration is one of the key factor that decides the reliability and availability of Vehicular Clouds. When the VC_M s drive at the same speed without accelerating or decelerating, at a given point of time, the containers are either migrating its workload from VC_I to VC_M or perform its task at VC_M or

migrate the result back to the VC_I from VC_M . While containers performing its task at VC_M , due to the speed variance of vehicles, VC_M s may leave out of the coverage of VC_I . To avoid this, before leaving the coverage of VC_I , VC_M s need to migrate its containers to another VC_M s within the coverage of VC_I . To make decision on when and where to migrate the container, in the CTM applied NVC environment, a time and space dependent procedure is implemented. The intention of applying CTM to NVC is to complete the container migration process before the VC_M leaves the coverage area of VC_I .

There can be a failure in either of the above steps while container migration. If the container failed to migrate back to the VC_I , the failed state would persist indefinitely and it is called as a non-recoverable state. For a period of time, a recoverable condition will continue in the broken state until the unfinished container is reassigned to the other possible vehicle and the incomplete job is done. The non-recoverable state is a condition at t and have a constant failure rate, if either the container with incomplete task cannot be assigned to any other vehicle or if the container migrates back to VC_I gets interrupted.

3.2 Problem Formulation

CTM [13] is used to simulate the time based actual traffic conditions for a defined stretch of road. The established VC in the 1 Km range is framed into CTM. The notations used are summarized as following, S as Segment set, ns as Number of segments, nrs as Number of segments in Range, S_d as Segment distance, S_σ as σ^{th} Segment ($\sigma \in [1..ns]$), d as Range distance (in km), spd as Vehicle Speed (kmph), B as Bottleneck Density (veh/km), Δ^R as Max. No. of vehicles that can flow into the range distance (veh/hr), NF as Normal Flow (%), BF as Bottleneck Flow (%), NFC as Normal Flow Count, BFC as Bottleneck Flow Count, t_s as Time step (in s), θ^N as Maximum number of vehicles that can flow into segment during normal flow, θ^C as Maximum number of vehicles that can flow into segment during bottleneck flow, $\delta_\sigma(t)$ as Number of vehicles in S_σ at time t , $\Delta_\sigma(t)$ as Maximum number of vehicles in S_σ at time t , $\theta_\sigma(t)$ as Maximum number of vehicles that can flow into S_σ at interval t , α as Number of vehicles in present segment at previous time stamp ($\delta_\sigma(t-1)$), β as Number of vehicles in previous segment at previous time stamp ($\delta_{\sigma-1}(t-1)$), γ as Number of vehicles in next segment at current time stamp ($\delta_{\sigma+1}(t)$), cs as Container Size. NVC is framed in the segments $S_{\sigma-1}$, S_σ and $S_{\sigma+1}$. Initially VC is built with VC_I in the segment S_σ and VC_M s in $S_{\sigma-1}$, S_σ and $S_{\sigma+1}$. Vehicles involved in this model can be denoted as σV_v , where $\sigma \in [1..(ns)]$, $v \in [1..(\delta_\sigma(t))]$, $V \in \{\sigma - 1V_v, \sigma - 1V_{v+1}, \dots, \sigma - 1V_{v+\Delta_{\sigma-1}(t)}, \sigma V_v, \sigma V_{v+1}, \dots, \sigma V_{v+\Delta_\sigma(t)}, \sigma + 1V_v, \sigma + 1V_{v+1}, \dots, \sigma + 1V_{v+\Delta_{\sigma+1}(t)}, \dots, \sigma + (ns-1)V_v, \sigma + (ns-1)V_{v+1}, \dots, \sigma + (ns-1)V_{v+\Delta_{\sigma+(ns-1)}(t)}\}$.

Among all the vehicles flowing into CTM, vehicles agreed to act as a VC_M are labelled with M and VC_I is labelled with I . The defined stretch of road where NVC is to be established, is parted into segments, S_σ . It is assumed that the vehicle speed and segment distance are constants [13]. The vehicle flow is also assumed from left to right on the road. The distance of the segment, S_d , is based on the time step, t_s . S_d is defined by the distance covered by a vehicle in one time step and the number of segments in the range can be calculated based on it.

$$S_d = spd \cdot \frac{t_s}{3600} \quad (1)$$

$$nrs = \frac{d}{S_d} \quad (2)$$

Since the vehicles are on the move, while establishing and maintaining VC, the position of range segments on the road differ for every time step. Though the range distance is 1 km, physically this 1 km distance varies on the road. For example, if 15 segments are assumed, the segments in the range are 1, 2 & 3 for the first time step, the segment in the range becomes 2, 3 & 4 for the second time step, etc. Hence, for the proposed work, simulation is carried out for 5 times the number of segments. The maximum number of vehicles in a segment at a time step is evaluated with the bottleneck density and S_d .

$$\Delta_{\sigma}(t) = S_d \cdot B \quad (3)$$

To demonstrate the real traffic condition, the road stretch is considered with Bottleneck Situations (BS) in varying time duration. BS induces on-the-road congestion. With the high congestion level, vehicles' speed is decreased. This forced the VC vehicles to stay with in the Range of Segments (ROS) for an extended period of time. The movement of vehicles into segments therefore depends on the density of vehicles that varies during the flow of bottleneck traffic, BF, and ordinary traffic flow, NF. θ^N and θ^C can be approximated with the assumed BF and NF. Δ^R , the maximum number of vehicles that can flow in the given road stretch per hour is assumed (3000 veh/hr) based on the road conditions. With Δ^R , $\theta_{\sigma}(t)$ can be calculated.

$$\theta^N = \frac{\Delta^R \cdot \left(\frac{NF}{100}\right)}{\left(\frac{3600 \text{ s}}{t_s}\right)} \quad (4)$$

$$\theta^C = \frac{\Delta^R \cdot \left(\frac{BF}{100}\right)}{\left(\frac{3600 \text{ s}}{t_s}\right)} \quad (5)$$

$$\theta_{\sigma}(t) = \frac{\Delta^R}{\left(\frac{3600 \text{ s}}{t_s}\right)} \quad (6)$$

In a normal traffic flow, it is presumed that all vehicles in a segment proceed to the next segment for each time step, t_s , $S_{\sigma+1}(t+1) = S_{\sigma}(t)$. Now, to measure the traffic flow at a time interval, it is important to determine the number of vehicles in each segment at each phase of the process. The traffic progression equation, thus, can be written as

$$\delta_{\sigma}(t) = \alpha + \beta - \gamma \quad (7)$$

$$\alpha = \delta_{\sigma}(t-1) \quad (8)$$

$$\beta = \min(\delta_{\sigma-1}(t-1), \theta_{\sigma}(t-1), \Delta_{\sigma}(t-1) - \delta_{\sigma}(t-1)) \quad (9)$$

$$\gamma = \min(\delta_{\sigma}(t-1), \theta_{\sigma+1}(t-1), \Delta_{\sigma+1}(t-1) - \delta_{\sigma+1}(t-1)) \quad (10)$$

A traffic entry table [13] is generated with the values of the equations Eqs. (1)–(6) as constants and the value of Eq. (7) as the occupancy of each segment at t . The entries in the traffic flow entry table are as seen in Tab. 3. Here, $S_{\sigma}(t)$ is considered as congested segment, $S_{\sigma+1}(t)$ is assumed as Bottleneck segment, α is the number of vehicles in $S_{\sigma}(t-1)$, β is the number of vehicles that are able to enter $S_{\sigma}(t)$ from $S_{\sigma-1}(t-1)$ and γ is the number of vehicles that are able to enter $S_{\sigma+1}(t)$ from $S_{\sigma}(t-1)$. Using these α , β and γ values, $\delta_{\sigma}(t)$, the number of vehicles in $S_{\sigma}(t)$ can be estimated. Each and every segment has its α , β and γ and values. These values in Tab. 2 belongs to $S_{\sigma}(t)$.

Table 2: Traffic flow entry table

	$\sigma-1$	σ	$\sigma+1$
$t-1$	β	α	
t		δ_{σ}	γ

When a vehicle from $S_\sigma(t-1)$ attempts to enter a bottleneck segment, $S_{\sigma+1}(t)$, it can move in to, only if it falls within the γ number of vehicles of $S_\sigma(t)$ or else, even during the next time step vehicle will remain in the same segment, $S_\sigma(t)$. When a vehicle attempts to reach the congested segment, $S_\sigma(t)$, from $S_{\sigma-1}(t-1)$ it will be queued back after $(\alpha - \gamma)$ number of vehicles of $S_\sigma(t)$. When the BS lasts for x t_s , then the vehicle entered $S_\sigma(t)$ from $S_{\sigma-1}(t-1)$ may remain in the same segment for $(x-1)$ t_s . It is assumed that vehicle flow is from left to right on the road. Throughout the vehicle movement along the road, VC_M s position is required to be around VC_I to maintain the VC connectivity.

By applying CTM to NVC, VC_I is identified as $\sigma V_v I$ and VC_M s can be identified as $\sigma - 1 V_v M, \sigma V_v M, \sigma + 1 V_v M$. Algorithm 1, presents the pseudocode that frames the NVC environment in CTM. In practical, vehicles speed can wave up to the road speed limit. Therefore vehicles can keep moving to next cells or it can slow down in the same cell for a specific period. This speed variation can lead the NVC participants to get disconnected from the NVC. This instance raises the requirement to focus on maintaining proximity between VC_I and VC_M . The position of the containers in the VC_M 's within VC can be tuned based on the VC_I 's position.

Algorithm 1: Framing NVC into cell transmission model

```

1: tsHr  $\leftarrow t_s/3600$ 
2:  $S_d \leftarrow spd \times tsHr$ 
3:  $\Delta_\sigma(t) \leftarrow S_d \times B$ 
4:  $\theta^N \leftarrow (NFC/100) \times tsHr$ 
5:  $\theta^C \leftarrow (BFC/100) \times tsHr$ 
6:  $\theta_\sigma(t) \leftarrow \Delta^R \times tsHr$ 
7:  $\alpha \leftarrow \delta_\sigma(t-1)$ 
8:  $\beta \leftarrow \min(\delta_{\sigma-1}(t-1), \theta_\sigma(t-1), \Delta_\sigma(t-1) - \alpha)$ 
9:  $\gamma \leftarrow \min(\alpha, \theta_{\sigma+1}(t-1), \Delta_{\sigma+1}(t-1) - \delta_{\sigma+1}(t-1))$ 
10: if  $(\theta_{\sigma-1}(t) = \theta^C$  and  $\delta_{\sigma-1}(t-1) \leq \theta^N)$ 
11:  $\delta_\sigma(t) \leftarrow \delta_{\sigma-1}(t-1)$ 
12: else if  $(S_\sigma(t) = \theta^C)$ 
13:  $\delta_\sigma(t) \leftarrow \alpha + \delta_{\sigma-1}(t-1) - \gamma$ 
14: Else
15:  $\delta_\sigma(t) \leftarrow \alpha + \beta - \gamma$ 
16: Endif

```

Vehicle speed spd is assumed as constant, so vehicles flow in same speed until a bottleneck situation (BS) occurs on the way of VC. Therefore, vehicles' acceleration can be decreased or sustained based on the occurrence of BS. VC_I checks for BS within its range at each time step, in $S_{\sigma+2}(t)$, $S_{\sigma+1}(t)$ and $S_\sigma(t)$ to classify the vehicle movement and categorize the vehicles based on it. If BS is detected in a segment, since not every vehicle in the previous segment can enter the BS segment, the vehicles are categorized as 'Movenext' and 'Slowdown' in the previous segment. This process is presented in Algorithm 2.

Algorithm 2: Identify BS and VC_I position category

```

1:  $S_{\sigma-1}(t)\_Movenext = \sigma - 1V_1(t)$  to  $\sigma - 1V_\gamma$  of  $\sigma-1(t+1)(t)$ 
2: if ( $\delta_{\sigma-1} > \gamma$  of  $S_{\sigma-1}(t+1)$ )
3:    $S_{\sigma-1}(t)\_Slowdown = \sigma - 1V_{(\gamma+1)}$  of  $\sigma-1(t+1)(t)$  to  $\sigma - 1V_{\delta_{\sigma-1}(t)}(t)$ 
4: endif
5:  $S_\sigma(t)\_Movenext = \sigma V_1(t)$  to  $\sigma V_\gamma$  of  $\sigma(t+1)(t)$ 
6: if ( $\delta_\sigma > \gamma$  of  $S_\sigma(t+1)$ )
7:    $S_\sigma(t)\_Slowdown = \sigma V_{(\gamma+1)}$  of  $\sigma(t+1)(t)$  to  $\sigma - 1V_{\delta_\sigma(t)}(t)$ 
8: Endif
9:  $x = v$  of  $\sigma V_v I$ 
10: if ( $x$  in  $S_\sigma(t)\_Movenext$ )
11:    $VC_PosCat = \text{"Movenext"}$ 
12: else if ( $x$  in  $S_\sigma(t)\_Slowdown$ )
13:    $VC_PosCat = \text{"Slowdown"}$ 
14: Endif
15:  $S_{\sigma+1}(t)\_Movenext = \sigma + 1V_1(t)$  to  $\sigma + 1V_\gamma$  of  $\sigma+1(t+1)(t)$ 
16: if ( $\delta_{\sigma+1} > \gamma$  of  $S_{\sigma+1}(t+1)$ )
17:    $S_{\sigma+1}(t)\_Slowdown = \sigma + 1V_{(\gamma+1)}$  of  $\sigma+1(t+1)(t)$  to  $\sigma + 1V_{\delta_{\sigma+1}(t)}(t)$ 
18: Endif

```

Migration decision is based on the position of VC_I in $S_\sigma(t)$. As discussed in Algorithm 1, if VC_I in $S_\sigma(t)$ is within the γ of $S_\sigma(t+1)$ positions, ($S_\sigma(t)_Movenext$ category) and if VC_{MS} in $S_{\sigma-1}(t)$ is not within the β of $S_\sigma(t+1)$ positions, ($S_{\sigma-1}(t)_Slowdown$ category) then containers in that range need to be migrated to the VC_{MS} in the range of first β of $S_\sigma(t+1)$, positions in $S_{\sigma-1}(t)$. The search for the destination is executed in the order of $S_{\sigma-1}(t)_Movenext$, $S_\sigma(t)_Slowdown$, and $S_\sigma(t)_Movenext$. If no destination VC_M has been identified in all three category of vehicles, declare the migration as unsuccessful. This process is presented in Algorithm 3.

As coded in Algorithm 4, migration is initiated with two parameters, containers to be migrated and the destination vehicles' range. With the assumption that no other overhead parameters are taken into account in this work, each container to be migrated is moved to the chosen destination, after finding the destination for migration. Predicting the percentage of vehicles in a particular segment S at time t helps to estimate the movement of VC participants with in CTM. This prediction is done using conditional probability mass function. Let T and X be the discrete random variables and [T,X] be a discrete random vector. $X=x$ is being informed, where x is the realization of X. The conditional probability mass function of T given $X=x$ is a function. $P_{T|X \rightarrow x}: R_{TX} \rightarrow \Omega$, $R_T \rightarrow \Omega_T$ where R_{TX} and R_T is the support vector and, Ω and Ω_T are the sample spaces. Here, $\Omega = \{i, j\}$, $i \in [1..t]$, $j \in [t_i, i+2]$, $t_i = \begin{cases} x-2, & \text{if } (x-i) > 0 \\ 1, & \text{if } (x-i) \leq 0 \end{cases}$ and $i, \text{ if } (x-i) > 2$

$\Omega_T = \{T_{sup1}, T_{sup2}, T_{sup3}, \dots, T_{sup n}\}$.

Algorithm 3: Container Migration and Management

```

1: Public  $Mig\_RC$ ;  $Mig\_SC$ ;  $C\_Mig\_NS$ 
2:  $C\_Mig\_NS$  : Not Successfully migrated containers
3:  $Mig\_RC$  : Migration required containers' count
4:  $Mig\_SC$  : Successfully Migrated containers' count
5:  $VC_I PosCat$  :  $VC_I$  's Position Category
6:  $Smoni$  : Segment to be monitored for Container Migration
7:  $Sncm$  : Segment number for Container Migration
8: Begin
9:   call Identify_Bottleneck_Situations()
10:  Switch ( $VC_I PosCat$ )
11:  Case "Movenext"
12:     $Sncm = -1$ 
13:     $Smoni = \text{"Slowdown"}$ 
14:  Case " Slowdown "
15:     $Sncm = 1$ 
16:     $Smoni = \text{" Movenext "}$ 
17:  End Switch
18:  if ( $S_{\sigma+Sncm}(t)\_Smoni \neq 0$ )
19:     $Mig\_RC = Mig\_RC + |\text{Containers in the } VC_{MS} \text{ of } (S_{\sigma+Sncm}(t)\_Smoni)|$ 
20:     $C\_Mig\_NS = \{\text{Containers to be migrated in the } VC_{MS} \text{ of } (S_{\sigma+Sncm}(t)\_Smoni)\}$ 
21:    initiate  $i$  to 0.
22:    Do
23:      increment  $i$  by 1.
24:      Switch ( $i$ )
25:      Case 1
26:         $dest = S_{\sigma+Sncm}(t)\_VC_I PosCat$ 
27:      Case 2
28:         $dest = S_{\sigma}(t)\_Smoni$ 
29:      Case 3
30:         $dest = S_{\sigma}(t)\_ VC_I PosCat$ 
31:      Case 4
32:        return "Un Successfull Container Migration"
33:      End Switch
34:      Initiate_Migration ( $C\_Mig\_NS$ ,  $VC_{MS}$  of  $dest$ )
35:      While ( $Mig\_SC < Mig\_RC$ )
36:      Endif
37: End

```

To derive the probability mass function of T lets compute the marginal probability mass function of S by iteratively adding the joint probability mass over the support of R_T . Let the joint probability mass function T be $P_{TX}(t, x) = \begin{cases} p, & \forall R_{TX} \\ 0, & otherwise \end{cases}$. The marginal probability mass function of S evaluated at $X=x$ is $P_X(x) = \sum_{(T,X) \rightarrow R_{TX}: X=x} P_{TX}(t, x)$. The conditional probability mass function of T given $X=x$ is

$$P_{T|X \rightarrow x}(x) = \begin{cases} \frac{P_{TX}(T_{sup1}, x)}{P_X(x)}, & \text{if } t = T_{sup1} \\ \frac{P_{TX}(T_{sup2}, x)}{P_X(x)}, & \text{if } t = T_{sup2} \dots \\ \frac{P_{TX}(T_{supn}, x)}{P_X(x)}, & \text{if } t = T_{supn} \end{cases}$$

4 Simulation

Simulation using MATLAB was performed to test the efficiency of the proposed model, with the metrics to be observed.

4.1 Simulation Settings

The model built on MATLAB includes many modules. The road mobility model is built in SUMO. NVC is established with the formal request-reply model using V2X's basic safety messages. The established NVC is framed in to CTM. A pre-informed migration decision algorithm is used to achieve the NVC task successfully. The simulation settings with the data are given in [Tab. 3](#).

4.2 Simulation Metrics

The rate of successful container migration and the rate of container migration requirement are the two metrics used to evaluate the proposed methodology. Only the dis connectivity of NVC participant due to high mobility is considered and assumed that the other metrics like enough resource space and the workload of the destination are well satisfied. The distance and time of VC_M and VC_I are used to gauge the success of container migration.

$$\text{Rate of Successful Container Migration} = \frac{Mig_SC}{Mig_RC} \quad (11)$$

$$\text{Rate of Un successful Container Migration} = \frac{C_Mig_NS}{Mig_RC} \quad (12)$$

$$\text{Rate of Container Migration Requirement} = \frac{|C_Mig_R|}{Total\ Containers\ in\ VC} \quad (13)$$

Algorithm 4: Initiate Migration

```

1: Private  $C\_Mig\_R, V\_Mig\_D, C\_Mig\_S$ 
2: Public  $C\_Mig\_NS, Mig\_SC$ 
3:  $C\_Mig\_R$  : To be migrated containers
4:  $V\_Mig\_D$  : Destination vehicles for container migration
5:  $C\_Mig\_S$  : Successfully migrated containers
6: if ( $|Mig\_R| > 0$ )
7:   if ( $VC_M$  found in  $Mig\_D$ )
8:     Migrate containers from  $Mig\_R$  to  $VC_M$ s of  $Mig\_D$ 
9:      $C\_Mig\_R = \{Containers\ in\ Mig\_R\}$ 
10:     $V\_Mig\_D = \{VC_Ms\ in\ Mig\_D\}$ 
11:     $C\_Mig\_S = \{Successful\ Container\ Migrations\ within\ V\_Mig\_D\}$ 
12:     $Mig\_SC = Mig\_SC + |C\_Mig\_S|$ 
13:     $C\_Mig\_NS = \{Un\ Successful\ Container\ Migrations\ within\ C\_Mig\_R\}$ 
14:  Endif
15: Endif

```

Table 3: Simulation settings

Parameter	Value	Parameter	Value
d	1 km	NFC	2400 veh/hr
S_d	0.3333 km	BFC	600 veh/hr
Spd	50 kmph	t_s	24 s
ns	15	$\Delta_\sigma(t)$	60
nrs	3	$\theta_\sigma(t)$	20
B	180 veh/km	θ^N	16
Δ^R	3000 veh/hr	θ^C	4
NF	80%	cs	10 MB
BF	20%		

4.3 Simulation Scenarios

The proposed model was initially simulated without any bottlenecks when testing the metrics. The vehicle movement flows smoothly without any need for container migration, as the vehicle speed is constant. The bottleneck situations on the VC trajectory were later added. For the following scenarios, calculations will be carried out:

4.3.1 Scenario I-Bottleneck Situations (BS) in the Various Part of Range of Segments (ROS) at $t + 1$ with Three Distinct Traffic Flows at t

The various traffic flow trends considered in this scenario are, Free flow in all the segments of ROS (F), Congestion only in $S_\sigma(C)$, Congestion in all the segments of ROS (A). As it is assumed that the flow of traffic

on the road is from left to right, $S_{\sigma+1}$ is viewed as the initial part of the ROS. The part of ROS are categorized as Initial : $S_{\sigma+1}$ (I), Mid: S_{σ} (M), Last : $S_{\sigma-1}$ (L).

When there exists a BS at the initial part of ROS, $S_{\sigma+1}(t+1)$, then each VC_M in segment $S_{\sigma+1}(t)$ is supposed to move to either $S_{\sigma+2}(t+1)$ or to $S_{\sigma+1}(t+1)$. Due to BS at $S_{\sigma+2}(t+1)$, only the assumed bottleneck flow percentage of vehicles can pass into $S_{\sigma+2}(t+1)$, and the remaining will be slowdown in $S_{\sigma+1}(t+1)$. While BS exists at the last part of ROS, $S_{\sigma}(t+1)$, each VC_M in the segment, $S_{\sigma-1}(t)$, are expected to move into either $S_{\sigma}(t+1)$ or $S_{\sigma-1}(t+1)$. Due to BS at $S_{\sigma}(t+1)$, only the assumed bottleneck flow percentage of vehicles will move to $S_{\sigma}(t+1)$, and the remaining will be slowdown in $S_{\sigma-1}(t+1)$. When BS happens in the mid part of ROS, $S_{\sigma+1}(t+1)$, the VC_I and VC_M s in $S_{\sigma}(t)$ may be moving to either $S_{\sigma+1}(t+1)$ or $S_{\sigma}(t+1)$. In this scenario, based on the expected VC_I 's position, the following cases of container migration are expected to happen.

- I. The destination selection for the migration of each container in VC_M s of $S_{\sigma+1}(t)$ _Movenext category is performed among the VC_M s of $S_{\sigma+1}(t)$ and $S_{\sigma}(t)$, in the following hierarchy: (i) VC_M s in the $S_{\sigma+1}(t)$ _Slowdown category. (ii) VC_M s in the $S_{\sigma}(t)$ _Movenext category. (iii) VC_M s in the $S_{\sigma}(t)$ _Slowdown category.
- II. The destination selection for the migration of each container in VC_M s of $S_{\sigma-1}(t)$ _Slowdown category is performed among the VC_M s of $S_{\sigma-1}(t)$ and $S_{\sigma}(t)$, in the following hierarchy: (i) VC_M s in the $S_{\sigma-1}(t)$ _Movenext category. (ii) VC_M s in the $S_{\sigma}(t)$ _Slowdown category. (iii) VC_M s in the $S_{\sigma}(t)$ _Movenext category.

4.3.2 Scenario II-Bottleneck Situations (BS) in Different Proximity

The different proximities are categorized as Initial segments-Earlier t_s , Mid segments-Mid t_s , Last segments-Last t_s . The category of proximity is further parted into segments as First part : $S_{\sigma-1}$, Mid part : S_{σ} , Last part : $S_{\sigma+1}$. The ROS at a specific t_s is the NVC position on the trajectory of NVC. To simulate the impact of BS proximity in the trajectory of NVC, BS is assumed on the path of NVC. The total number of segments are partitioned into First, second and third blocks of segments for earlier, mid and late t_s and again each block is parted into first, mid and last part of the segment blocks. The total number of segments are partitioned into 3 blocks. $ns_p = ns/3$. Each block can be separated as $Seg_Block_i = [1 + (ns_p \times (i - 1))]to[ns_p \times i]$ where $i \in [1, 2, 3]$. In each Seg_Block_i , the first part of the segment is n , mid part is $n+1$, $n+2$ and last part is $n+3$, $n+4$ and the BS time steps for each Seg_Block_i is assumed to be $n+1$, $n+4$, where n is $(ns_p \times (i - 1))$.

4.3.3 Scenario III-the Occurrence Time of Bottleneck Situations

The BS occurrence time is categorized as Single t_s , Double t_s , triple and above t_s . To simulate the impact of BS duration on the trajectory of VC, BS are assumed on the path of VC for single t_s , double t_s and triple and above t_s in the same environment of scenario II.

4.4 Simulation Results

Simulation was conducted by applying the settings and metrics at various phases as proposed in the previous sections. Each phase was simulated for 10 times and the results collected were averaged. The following subsections discusses about each phase of simulation with the given settings and metrics.:

4.4.1 Rate of Successful Container Migration (SCM) Based on the Impact of Bottleneck Situations (BS), in Different Part of ROS

Based on the simulations conducted, it is summarized that, regardless of the occurrence of BS in different part of ROS, container migration is not required in the following cases:

- i) When there is a free traffic flow at t_s and BS at the initial part of ROS at $(t+1)$

- ii) When there is a free flow at t_s and VC_I positions at $S_\sigma(t)$ _Movenext with BS in the mid part of ROS at $(t + 1)$,
- iii) When there is congestion only in S_σ at t_s with VC_I positions at $S_\sigma(t)$ _Movenext and BS at the initial and mid part of ROS at $(t + 1)$.

Among the following container migration requiring cases, case (i) and (ii) are due to the occurrence of BS, case (iii) is partially due to the occurrence of BS and other cases of migrations are required regardless of the occurrence of BS.

- i) When there is a free traffic flow at t_s and BS at the last part of ROS at $(t + 1)$,
- ii) When there is a free flow at t_s and VC_I positions at $S_\sigma(t)$ _Movenext with BS in the mid part of ROS at $(t + 1)$,
- iii) When there is congestion only in S_σ at t_s and BS at last part of ROS at $(t + 1)$,
- iv) When there is congestion only in S_σ at t_s and VC_I positions at S_σ _Slowdown with BS in initial and mid part of ROS at $(t + 1)$,
- v) When there is congestion in all segments of ROS at t_s and BS in anywhere in initial, mid and last part of ROS at $(t + 1)$.

For the above cases of required container migrations, item (i) requires |containers in $S_{\sigma-1}(t)$ _Slowdown| number of container migrations, item (ii) and (iv) requires |containers in $S_{\sigma+1}(t)$ _Movenext| and item (iii) and (v) requires |containers in $S_{\sigma+1}(t)$ _Movenext| or |containers in $S_{\sigma-1}(t)$ _Slowdown|. For the migration requirements of containers in $S_{\sigma+1}(t)$ _Movenext, the destinations are VC_{Ms} of $S_{\sigma+1}(t)$ _Slowdown and $S_\sigma(t)$. For the migration requirements of containers in $S_{\sigma-1}(t)$ _Slowdown, the destinations are VC_{Ms} of $S_{\sigma-1}(t)$ _Movenext and $S_\sigma(t)$.

Fig. 1 illustrates the destination availability ratio of each case. The BS raises the container migration requirement in the following cases with average availability of container migration destinations: BS at mid part and last part of ROS with free traffic flow and congestion in all parts of ROS. Based on Fig. 1, the availability of container migration destination is classified in Tab. 4. Fig. 2 shows the rate of successful container migrations based on the impact of BS in various parts of ROS and different patterns of traffic flows. Whenever the container migration requirement is raised due to the occurrence of BS, the availability ratio is average, which in turn impacts the success rate of container migration within ROS. Comparing with the work of [6], where with the unrestricted resource usage and high congestion level, 97% of successful VM migration is achieved. In this work, with the unrestricted resource usage and high congestion in the entire ROS, the simulation results 98.8% of successful container migration.

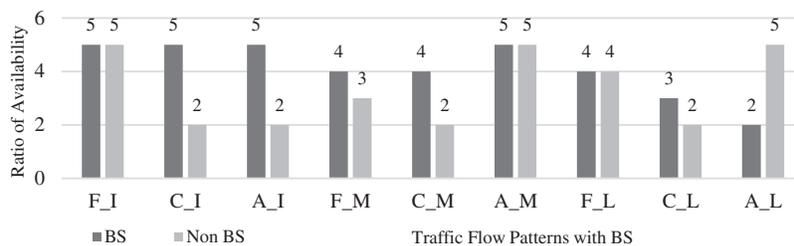
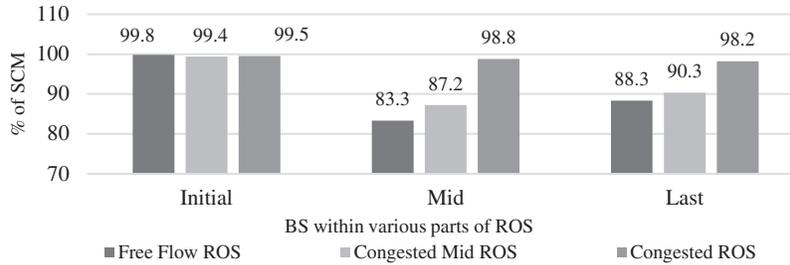


Figure 1: Ratio of destination availability for container migration

Table 4: Container migration destination availability based on the BS location within ROS

Traffic pattern	Initial ROS	Mid ROS	Last ROS
Free flow (F)	High	Average	Average
Congestion at $S_{\sigma}(t)$ (C)	High	Average	Average
Congestion at all parts of ROS (A)	High	High	Low

**Figure 2:** Ratio of SCM based on the impact of BS in various parts of ROS and different patterns of traffic flows

4.4.2 Rate of Successful Container Migration (SCM) Based on the Impact of BS Proximity on the Trajectory of VC

In the first part of the segments in each Seg_Block_i , the BS has no effect on the SCM rate as it is always in non-ROS. The BS has a high SCM rating in the last part of segments in the first segment block, (Seg_Block_1). In this instance, the VC has a wide number of destinations for container migration as the BS occurred in the earlier time steps. Container migration is needed for the congested segment before the BS segment at each t_s . When BS occurs in a segment x that falls in to the ROS for t number of t_s , then there raises container migration requirement for the number of $|\text{containers in } S_{x \cdot (t_s + i)_Movenext}|$, where $i=0$ to $t-1$. VC participants float with in the congested and BS segment till the duration of BS. After BS is cleared, the segment where BS occurred allows the maximum flow. So the VC trajectory have a high number of vehicles until another BS happens on its way.

The BS in the mid part of segments in each Seg_Block_i , results VC to partially move into BS segment. VC's ROS differ based on the VC_1 's position within its segment. If VC_1 falls in to the $Movenext$ category, while encountering BS segment, VC travels on the $S_{\sigma-1}$, S_{σ} and $S_{\sigma+1}$ with σ have the value of t_s for further time steps. If VC_1 falls in to the Slowdown category, while encountering BS segment, VC participants float within the congested segment before BS segment and in the BS segment until the occurrence of BS and continue to travel on $S_{\sigma-1}$, S_{σ} and $S_{\sigma+1}$ with σ have the value of t_s .

There are $|\text{containers in } S_{x \cdot (t_s + i)_Movenext}|$ number of container migration requirements, where $i=0$ to $t-1$, when BS occurs in a segment x that falls in to the ROS for t number of t_s . The BS in mid part of segments at earlier t_s in the first block of segments, have low rates of SCM, since the VC travels with a low number of vehicles within ROS. As showed in Fig. 3, the rate of SCM gradually increases in the second and third block of segments due to increase in the number of vehicles within the ROS as the BS happens during the later time steps. On the average, the impact of BS proximity on the trajectory of VC results 98.2% of SCM.

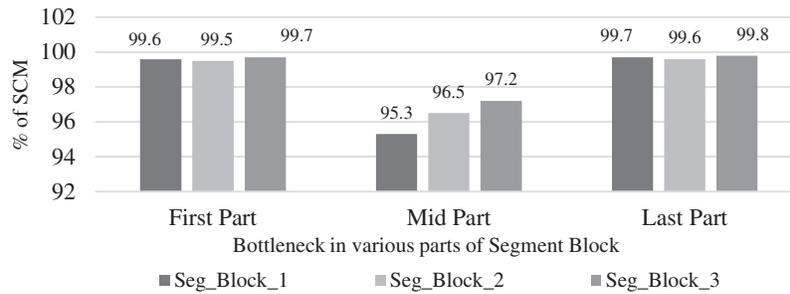


Figure 3: Ratio of SCM based on the effect of BS proximity on the trajectory of VC

4.4.3 Rate of Successful Container Migration (SCM) Based on the Impact of the Length of the BS on the VC Trajectory

If BS lost for a single t_s in the first part of all the segment blocks, as in the other scenarios, completely have no impact. BS for single t_s in the mid part of segment blocks, results comparatively high negative impact due to the less number of destinations for migration. This result is based on the VC_1 position and the container migration requirements raised due to it. In the last part of segment blocks, BS for triple and above t_s has a positive impact as this case has a high number of destinations for migration. These facts are displayed in Fig. 4. While the same trend of the SCM rate follows in both double t_s and single t_s within the different parts of segment blocks, but as in Figs. 5 and 6, due to the reduction in the incidence period of BS, there is a steady decrease in the total SCM rate.

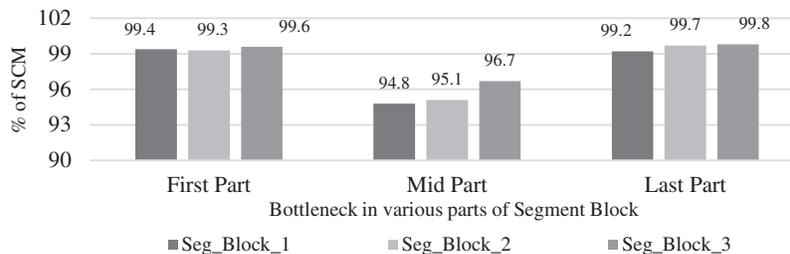


Figure 4: Ratio of SCM based on the impact of BS for triple and above time steps in various block of segments

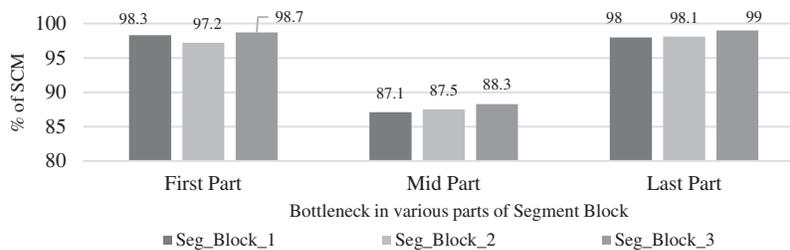


Figure 5: Ratio of SCM based on the impact of BS for double time steps in various block of segments

It can be summarized that (i) Except when the bottleneck occurs at the end of the ROS, the inevitable bottleneck situations increase the availability of the container migration destinations. (ii) On the VC trajectory, when the bottleneck occurs in the near future, the residual in-network time of VC participants is increased and the requirements for container migration are reduced. (iii) When migrating containers to the subsequent vehicles from the migrating source, the rate of effective container movement is

high relative to vehicles travelling ahead of the migrating source. In Tab. 5 all the simulation outcomes are compared.

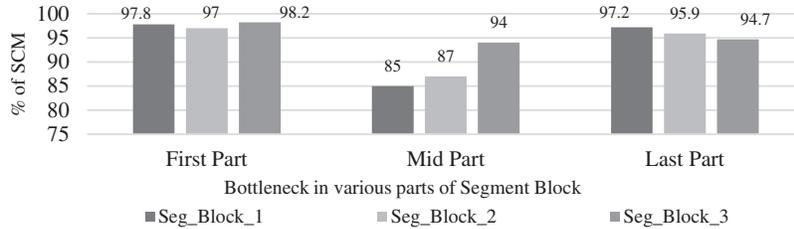


Figure 6: Ratio of SCM based on the impact of BS for single time steps in various block of segments

Table 5: Average rate of SCM comparison for various BS scenario

BS scenario	Traffic pattern			Average
	Free flow ROS	Congested mid ROS	Congested ROS	
BS within ROS	90.47	92.3	98.8	93.9
	Seg. block 1	Seg. block 2	Seg. block 3	
	BS proximity	98.2	98.5	
BS duration-single t_s	93.3	93.3	95.6	94
Double t_s	94.5	94.3	95.3	94.7
Triple t_s	97.8	98	98.7	98.2

The MDWLAM [6] algorithm at high density of congestion with unrestricted resources results 97% of successful VM migrations, comparatively, this work results 98.5% of successful container migrations in the BS proximity scenario and 98.2% in the triple t_s BS duration scenario.

5 Conclusion

This paper has proposed Nomadic Vehicular Cloud with two preeminent features: (i) only moving vehicles on the heavy traffic road are the NVC participants and (ii) Applying CTM as aid to decide about the initiation time of container migration. NVC is simulated with containers and 5G NR V2X PC5 interface. The usage of PC5 interface helps NVC to function independently without relying on the cellular network coverage. In terms of its portability and short start-up time, the lightweight containers are very apt for the functioning of NVC. From the simulations it is noted that with the estimation of the vehicle trajectory movement and the number of vehicles in a road with bottleneck situation, there is a significant increase in the rate of successful container migrations. In turn it also increases the success rate of NVC. For future work, the same proposal can be tested with more two-way lanes on the road, can apply additional 5G NR V2X features, such as mm Wave for NVC connectivity, can incorporate the effects of path loss models in NVC and can analyse the effect of dual mobility in NVC.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] P. Ghazizadeh, R. Florin, A. G. Zadeh and S. Olariu, "Reasoning about mean time to failure in vehicular clouds," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 3, pp. 751–761, March 2016.
- [2] R. Florin, S. Abolghasemi, A. Ghazizadeh and S. Olariu. "Big data in the parking lot," in *Big Data Management and Processing*, 1st edition., New York: Taylor and Francis, pp. 21.1–21.8, 2017.
- [3] L. Decker, "Supporting big data at the vehicular edge," *Ph.D. dissertation*. Old Dominion University, Virginia, US, 2018.
- [4] A. Ghazizadeh, "Towards dynamic vehicular clouds," *Ph.D. dissertation*. Old Dominion University, Virginia, US, 2020.
- [5] A. Ghazizadeh, P. Ghazizadeh and S. Olariu, "Job completion time in dynamic vehicular cloud under multiple access points," in *Int. Conf. on Cloud Computing*, Cham, Springer, vol. 12403, 2020.
- [6] T. K. Refaat, B. Kantarci and H. T. Mouftah, "Virtual machine migration and management for vehicular clouds," *Vehicular Communications*, vol. 11, no. 6, pp. 47–56, 2016.
- [7] A. Bazzi, G. Cecchini, M. Menarini, B. M. Masini and A. Zanella, "Survey and perspectives of vehicular Wi-Fi versus sidelink cellular-V2X in the 5G era," *Future Internet*, vol. 11, no. 6, pp. 122, 2019.
- [8] H. Bagheri, M. Noor-A-Rahim, Z. Liu, H. Lee, D. Pesch *et al.*, "5G NR-V2X: Toward connected and cooperative autonomous driving," *IEEE Communications Standards Magazine*, vol. 5, no. 1, pp. 48–54, 2021.
- [9] R. Molina-Masegosa, J. Gozalvez and M. Sepulcre, "Configuration of the C-V2X mode 4 sidelink PC5 interface for vehicular communication," in *14th Int. Conf. on Mobile Ad-Hoc and Sensor Networks (MSN)*, Shenyang, China, pp. 43–48, 2018.
- [10] M. Rondinone and A. Correa, "Definition of V2X message sets," *TransAID Consortium*, 2018.
- [11] M. Tao, J. Li, J. Zhang, X. Hong and C. Qu, "Vehicular data cloud platform with 5G support: Architecture, services, and challenges," in *IEEE Int. Conf. on Computational Science and Engineering and IEEE Int. Conf. on Embedded and Ubiquitous Computing*, Cham, pp. 32–37, 2017.
- [12] Z. Li, M. Kihl, Q. Lu and J. A. Andersson. "Performance overhead comparison between hypervisor and container based virtualization," in *IEEE 31st Int. Conf. on Advanced Information Networking and Applications*, Taipei, Taiwan, IEEE, pp. 955–962, 2017.
- [13] C. F. Daganzo, "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp. 269–287, 1994.
- [14] F. Sun, F. Hou and N. Cheng, "Cooperative task scheduling for computation offloading in vehicular cloud," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 11, pp. 11049–11061, 2018.
- [15] J. Shao and G. Wei, "Secure outsourced computation in connected vehicular cloud computing," *IEEE Network*, vol. 32, no. 3, pp. 36–41, 2018.
- [16] R. Florin, A. Ghazizadeh, P. Ghazizadeh, S. Olariu and D. C. Marinescu, "Enhancing reliability and availability through redundancy in vehicular clouds," *IEEE Transactions on Cloud Computing*, vol. 9, no. 3, pp. 1061–1074, 2021.
- [17] M. H. C. Garcia, A. Molina-Galan and M. Boban, "A tutorial on 5G NR V2X communications," *IEEE Communications Surveys Tutorials*, vol. 22, no. 3, pp. 1972–2026, 2021.
- [18] J. Lianghai, A. Weinand, B. Han and H. D. Schotten, "Multi-RATs support to improve V2X communication," in *2018 IEEE Wireless Communications and Networking Conf. (WCNC)*, Barcelona, Spain, pp. 1–6, 2018.
- [19] K. Ganesan, J. Lohr, P. B. Mallick, A. Kunz and R. Kuchibhotla, "NR sidelink design overview for advanced V2X service," in *IEEE Internet of Things Magazine*, vol. 3, no. 1, pp. 26–30, 2020.
- [20] A. Bazzi, B. M. Masini, A. Zanella and I. Thibault, "On the performance of IEEE 802.11p and LTE-V2V for the cooperative awareness of connected vehicles," in *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10419–10432, 2017.
- [21] R. Regin and T. Menakadevi. "Dynamic clustering mechanism to avoid congestion control in vehicular ad hoc networks based on node density," *Wireless Personal Communications*, vol. 107, no. 4, pp. 1911–1931, 2019.
- [22] K. Ganesan, P. B. Mallick, J. Lohr, D. Karampatsis and A. Kunz, "5G V2X architecture and radio aspects," in *2019 IEEE Conf. on Standards for Communications and Networking (CSCN)*, Granada, Spain, pp. 1–6, 2019.

- [23] S. Maheshwari, S. Choudhury, I. Seskar and D. Raychaudhuri, "Traffic-aware dynamic container migration for real-time support in mobile edge clouds," in *2018 IEEE Int. Conf. on Advanced Networks and Telecommunications Systems (ANTS)*, Indore, India, pp. 1–6, 2018.
- [24] K. S. Gautam and T. Senthil Kumar, "Video analytics-based intelligent surveillance system for smart buildings," in *Springer Soft Computing*, vol. 23, no. 8, pp. 2813–2837, 2019.
- [25] A. Vishnu Priya and A. k. srivastava, "Hybrid optimal energy management for clustering in wireless sensor network," in *Computers and Electrical Engineering*, vol. 86, pp. 379–387, 2020.