UNIVERSIDADE FEDERAL DO PARANÁ

LEONARDO BARBOSA DA SILVA

BOUNDING BOX STRATEGIES FOR VULNERABLE ROAD USER CLUSTERS IN V2X NETWORKS

CURITIBA

2024

LEONARDO BARBOSA DA SILVA

BOUNDING BOX STRATEGIES FOR VULNERABLE ROAD USER CLUSTERS IN V2X NETWORKS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica, no Setor de Tecnologia, na Universidade Federal do Paraná, como requisito parcial à obtenção do título de Mestre em Engenharia Elétrica.

Supervisor: Prof. Dr. Evelio Martín García Fernández
Co-supervisor: Prof. Dr. Christian Facchi

CURITIBA

2024

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Silva, Leonardo Barbosa da
Bounding box strategies for vulnerable road user clusters in V2X networks / Leonardo Barbosa da Silva. – Curitiba, 2024.
1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica.

Orientador: Evelio Martín García Fernández
Coorientador: Christian Facchi

1. Estradas - Medidas de segurança. 2. Comunicações rodoviárias. 3. Telecomunicações – Tráfego. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Engenharia Elétrica. III. García Fernández, Evelio Martín. IV. Facchi, Christian. V. Título.

Bibliotecário: Elias Barbosa da Silva CRB-9/1894

MINISTÉRIO DA EDUCAÇÃO
SETOR DE TECNOLOGIA
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO ENGENHARIA ELÉTRICA - 40001016043P4

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação ENGENHARIA ELÉTRICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **LEONARDO BARBOSA DA SILVA** intitulada: **Bounding Box Strategies for Vulnerable Road User Clusters in V2X Networks**, sob orientação do Prof. Dr. EVELIO MARTÍN GARCÍA FERNÁNDEZ, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.
A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 22 de Agosto de 2024.

Assinatura Eletrônica
30/08/2024 12:18:12.0
EVELIO MARTÍN GARCÍA FERNÁNDEZ
Presidente da Banca Examinadora

Assinatura Eletrônica
14/10/2024 21:02:11.0
ANELISE MUNARETTO FONSECA
Avaliador Externo (UNIVERSIDADE TECNOLóGICA FEDERAL DO PARANá/CURITIBA)

Assinatura Eletrônica
26/09/2024 15:51:50.0
CARLOS MARCELO PEDROSO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

To my family and friends, who I love and miss dearly.

ACKNOWLEDGEMENTS

I want to express my sincerest gratitude to Prof. Dr. Evelio Martín García Fernández for his full support not only through plenty of advice and scientific input during the development of this Master's Thesis but also for helping me along the whole process of the double degree program.

Thank you also to Prof. Dr. Christian Facchi, who warmly welcomed me into his research group in Ingolstadt, offering me plenty of teachings and making me a better researcher and engineer.

I am also thankful to M.Sc. Silas Correia Lobo and my other colleagues in the Car2X Laboratory from Technische Hochschule Ingolstadt for all the collaborations in research, projects, and publications. Thank you also for your support and friendship these past two and a half years.

Thank you to the Universidade Federal of Paraná and the Technische Hochschule Ingolstadt for providing the tools and knowledge needed to conduct my Master's research. Both institutions foster an environment of innovation and collaboration that was fundamental for me to perform my studies at the highest level.

My gratitude also goes to all my friends for being present whenever I needed it. Without your encouragement, I would have never gotten as far as I did. Knowing that I can count on your help wherever I am is reassuring.

Finally, I am immensely grateful to my parents Emilia Minae and Julio Cezar, my sister Juliana Yassue, and my uncle Julio Kazuaki, for all their love, incentive, patience, and advice despite the distance. Thank you for pushing me to always work hard for what I believe in and to stay curious.

RESUMO

No campo da comunicação Vehicle-to-Everything, tipos distintos de usuários de vias são levados em consideração quando novos casos de uso e padrões de mensagem são desenvolvidos. Entre os numerosos possíveis participantes de tráfego, o termo *Vulnerable Road User* (VRU, ou Usuário Vulnerável de Estrada) é atribuído aos nós de comunicação sob maior risco no evento de um acidente, incluindo, mas não limitando-se a pedestres, ciclistas e trabalhadores de rodovia. Com a proteção destes agentes em mente, a *VRU Awareness Message* (VAM, ou Mensagem de Conscientização de VRUs) é introduzida como uma abordagem que permite a um VRU participação ativa em sua segurança em sistemas de transporte. Através dessas mensagens, VRUs podem transmitir sua posição e dados dinâmicos, como velocidade e orientação, informando outros usuários de sua presença e garantindo que sejam percebidos, reduzindo a probabilidade de colisões. A comunicação nessas aplicações é realizada diretamente, com os dispositivos enviando e processando os dados uns dos outros, seguindo uma estrutura de rede decentralizada. Porém, incluir pedestres como nós de comunicação pode pode levar ao congestionamento do canal em áreas superlotadas. Como tal, VAMs oferecem o recurso de clustering (agrupamento) no qual VRUs com cinemáticas e posições similares podem se agrupar de maneira que somente um dispositivo transmite mensagens que representam todo o cluster. O *VRU Basic Service* (Serviço Básico de VRU) proposto nas especificações ETSI 103 300 do European Telecommunications Standards Institute (Instituto Europeu de Normas de Telecomunicação) descreve o cluster como uma caixa delimitadora que deve encobrir todos os seus membros utilizando uma forma geométrica de maneira que os veículos na vizinhança possam evitar colidir com os VRUs contidos. Este trabalho contribui com o esforço de padronização ao introduzir uma estrutura de dados, o Mapa de Cluster, para o clustering no VRU Basic Service. Além disso, este trabalho é o primeiro a sugerir estratégias para formar todos os tipos de formas de caixas delimitadoras previstas no padrão. Todas estas contribuições são então implementadas como novos módulos para o framework de simulação Artery V2X. Resultados de simulação mostram que apesar dos tipos de geometrias oferecerem resultados simulares em alguns aspectos, eles possuem comportamentos opostos de maneira geral. Formas circulares são adequadas para canais mais ocupados, enquanto caixas delimitadoras poligonais são recomendadas para aplicações nas quais fidelidade de forma da nuvem de pontos do cluster é preferível.

Palavras-chaves: Usuários Vulneráveis de Vias. Agrupamento de VRUs. Caixas Delimitadores de Cluster. Comunicação V2X. Mensagem de Conscientização de VRUs. Serviço Básico de VRU.

ABSTRACT

In the field of Vehicle-to-Everything communication, different road user types are taken into consideration when developing new use cases and message standards. Among the numerous possible traffic members, the term *Vulnerable Road User* (VRU) is assigned to the nodes at most danger in the event of an accident, including but not limited to pedestrians, cyclists, and road workers. With the protection of said agents in mind, the *VRU Awareness Message* (VAM) has been introduced as an approach that allows a VRU active participation in its safety on traffic systems. Through those messages, VRUs can broadcast their position and dynamic data, e.g., speed and heading, informing other users of their presence and ensuring they were perceived, reducing the likelihood of collisions. The communication in these applications is done directly, with the devices sending and processing the data from each other, following a decentralized network structure. However, including pedestrians as transmitting nodes can lead to channel congestion in crowded areas. As such, VAMs offer a clustering feature in which VRUs with similar kinematics and positions can group themselves so that only one device transmits messages representing the whole cluster. The VRU Basic Service proposed in the 103 300 specifications by the European Telecommunications Standards Institute describes the cluster as a bounding box that must cover all its members using a geometric shape so that other vehicles in the vicinity can avoid colliding with the contained VRUs. This work contributes to the standardization effort by introducing a data structure, the Cluster Map, for the clustering in the VRU Basic Service. Furthermore, this work is the first to suggest strategies for forming all the standardized bounding box shape types. All those contributions are then implemented as new Artery V2X simulation framework modules. Simulation results show that even though each of the geometry types offers similar results in some regards, they have opposing behaviors overall. Circular shapes are suited for busier communication channels, while polygon bounding boxes are advised in applications where the shape-fidelity of the cluster point cloud is preferred.

Key-words: Vulnerable Road Users. VRU Clustering. Cluster Bounding Boxes. V2X Communication. Vulnerable Road User Awareness Message. VRU Basic Service.

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ACRONYMS

3GPP	Third Generation Partnership
AABB	Axis Aligned Bounding Box
API	Application Programming Interface
ASN.1	Abstract Syntax Notation One
BSM	Basic Safety Message
BTP	Basic Transport Protocol
BV	Bounding Volume
C-ITS	Cooperative Intelligent Transport Systems
C-V2X	Cellular V2X
C2C-CC	Car 2 Car Communication Consortium
CAM	Cooperative Awareness Message
CARE	Community database on Accidents on the Roads in Europe
CBR	Channel Busy Ratio
CIC	Cluster Information Container
CM	Cluster Map
COC	Cluster Operation Container
CPM	Collective Perception Message
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DCC	Decentralized Congestion Control
DENM	Decentralized Environmental Notification Message
DSRC	Dedicated Short Range Communication
ERSO	European Road Safety Observatory
ETSI	European Telecommunications Standards Institute

GEMV2	Geometry-based Efficiency Propagation Model for Vehicle-to-Vehicle
GN	GeoNetworking
ITS	Intelligent Transport Systems
ITS-S	Intelligent Transport Systems Station
InTAS	Ingolstadt Traffic Scenario for SUMO
LDM	Local Dynamic Map
LTE	Long-Term Evolution
LiDAR	Light Detection and Ranging
MAC	Media Access Control
NED	Network Description
OFDM	Orthogonal Frequency Division Multiplexing
OMNeT++	Objective Modular Network Testbed in C++
PHY	Physical Layer
QoS	Quality of Service
RB	Resource Blocks
RFID	Radio Frequency Identification
RHS	Road Hazard Signalling
RSU	Roadside Unit
SC-FDMA	Single Carrier Frequency Division Multiplexing Access
SUMO	Simulation of Urban MObility
TraCI	Traffic Control Interface
UC-B2	Use Case B2
V2X	Vehicle-to-everything
VAM	Vulnerable Road User Awareness Message
VBS	Vulnerable Road User Basic Service

VRU	Vulnerable Road User
WAVE	Wireless Access in Vehicular Environments
WLAN	Wireless Local Area Networks
WSMP	Wave Short Message Protocol
WSN	Wireless Sensor Networks
XML	Extensible Markup Language
iCM	Initial Cluster Map
mCM	Maintenance Cluster Map

LIST OF SYMBOLS

d$_{Bf}$	Buffer distance
T$_{CBR}$	CBR time standardized by the ITS-G5 stack
CBR	Channel Busy Ratio
D	Cluster density
d$_{max}$	Distance between the furthest VRU and the AABB center
v$_{VRU}$	Highest VRU velocity
log	Logarithmic function
θ	Rotation angle of the point cloud
v$_{diff}$	Speed difference
T$_{Busy}$	Time a single channel is busy with transmissions
$\mathcal{O}(\,)$	Upper bound of space time complexity of an algorithm
t$_{VAM}$	VAM assembly time
VRU$_{low}$	y-lowest VRU in a point cloud

CONTENTS

1 INTRODUCTION

According to a report from the United Nations - Department of Economic and Social Affairs (2018), the current world population is going through an urbanization process. Every year, the concentration of people living in urban areas grows in comparison to rural regions, and by the year 2030, projections suggest that over 60% of the world's population will live in urban environments (UNITED NATIONS - DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS, 2018). For Europe and, more specifically, Germany, the prediction for 2030 tends even more toward urbanization, with respective percentages of 77.47% and 78.93%. By 2050 the same study expects the world population to be 68.36% urban, while Europe and Germany will have an urbanization rate of over 80%, indicating that this trend will continue in the following decades.

As the populations in urban environments grow, more people will need to circulate these areas as part of their daily routines, including activities like commuting and running errands. According to a 2022 report by the European Commission (ARMOOGUM et al., 2022), a European makes around 2.7 trips per day with an average total distance traveled of 34.8 km and a duration of 67 minutes.

Based on data published in 2019 by the German Federal Ministry for Digital and Transport (FOLLMER; GRUSCHWITZ, 2019), Germany follows a similar pattern to the European statistics. As can be observed in TABLE 1, between 2002 and 2017, there is an increase in the daily average distance and time spent traveling per individual, while the amount of trips has remained similar. The report associates this behavior with a trend of conurbation, which is the enlargement of urban areas due to the fusing of towns that surround each other, leading to the interconnection of these regions in many aspects, such as the traffic systems.

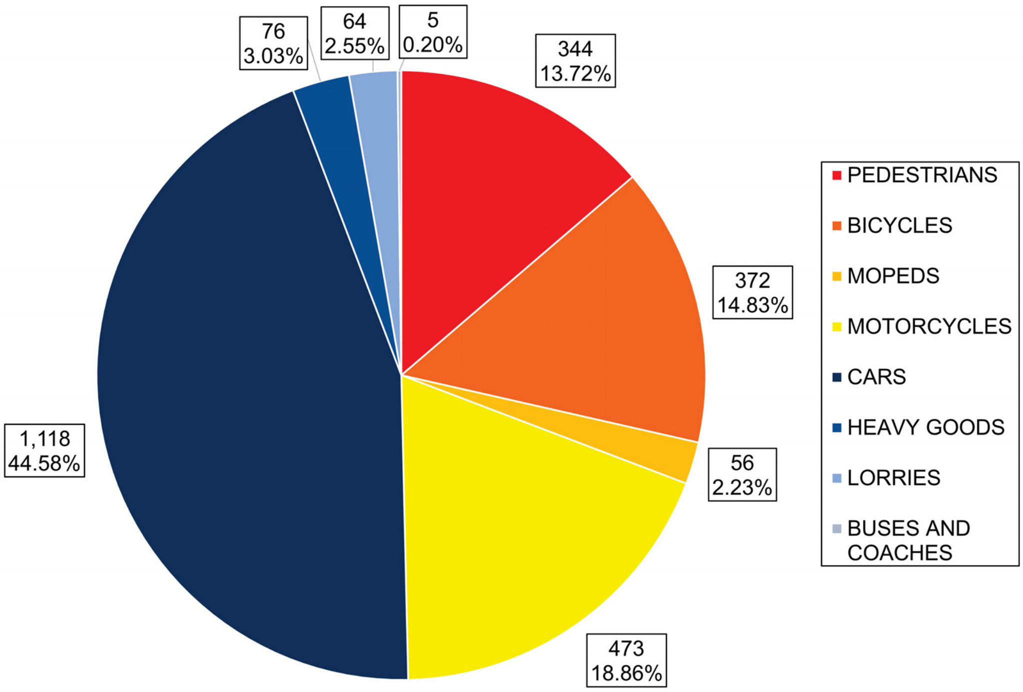
TABLE 1 – Daily trips per person in Germany (FOLLMER; GRUSCHWITZ, 2019)

	2002	**2008**	**2017**
Daily trips per person	3.3	3.4	3.1
Traveling time (min/person)	76	82	85
Daily distance traveled (km/person)	33	38	39

Those numbers indicate that people nowadays spend more time on the move than in previous years. As such, it is reasonable to presume that, more than ever, traffic needs to be strategically managed to minimize the likelihood of accidents. Considering that some groups of transportation are more at risk in the event of a collision than others, especially cyclists, motorcyclists, moped riders, and pedestrians, this is even more relevant. These entities can be categorized as *Vulnerable Road Users* (VRUs)

(ETSI, 2022) and should be treated with additional emphasis since they represent a significant amount of the fatal victims of traffic accidents as shown on FIGURE 1.

FIGURE 1 – Traffic accident fatalities in Germany (2021), elaborated with data from (DECAE, 2023)



Based on data from the *Community database on Accidents on the Roads in Europe* (CARE), managed by the *European Road Safety Observatory* (ERSO), 1,245 of the road accident fatalities reported in Germany in the year 2021 were VRUs, roughly representing 49.64% of the 2,508 total registered occurrences. An expressive share of the VRUs involved in those accidents were pedestrians, with 344 reports (13.72%). A similar trend is perceptible when the numbers from the whole European Union are considered, with 19,484 total deaths, 48.01% of which were VRUs and, more specifically, 18.83% were pedestrians. Meaning that nearly one out of every five people killed in a traffic accident in the European Union is a pedestrian (DECAE, 2023). That is especially worrying because, according to reports from ERSO, 98% of the deaths in collisions involving pedestrians are the pedestrians themselves (SLOOTMANS, 2021). Suggesting that even within the VRUs, they are especially vulnerable, and authorities should make additional efforts to ensure their protection.

1.1 VEHICLE-TO-EVERYTHING COMMUNICATION

One way to improve the prevention of traffic accidents involving pedestrians in the current context of smart cities is through the use of wireless communication. In the vehicular research field, this type of communication is called *Vehicle-to-Everything* (V2X). Among many applications, they can be used to exchange messages between the different actors that can be found in those mobility scenarios. This can be used for example to alert a vehicle approaching a crossing that there are pedestrians in their blind spots (XIOG et al., 2018), enhancing the awareness of said car to potential risks and by doing that, raises the traffic security in this particular area. Plenty of maneuvers and traffic management use cases can be assisted by V2X services involving different levels of automation, such as lane merging, emergency braking, safe overtaking, intersection management, optimal traffic light control, etc (C2C-CC, 2023). To ensure interoperability in the real world, many standards are in development to ensure that every message and service implemented follows similar guidelines and offers a minimum performance threshold.

The *European Telecommunications Institute* (ETSI) proposed one of said specifications through the definition of the *Intelligent Transport Systems* (ITS) concept, leading to the ITS-G5 standard (ETSI, 2010), named after the allocated 5.9 GHz frequency band. ITS-G5 is commonly referred to as a WiFi-based V2X approach because it uses the Physical and Medium Access Control layers from IEEE 802.11p (FESTAG, 2015). In this paradigm, devices can operate cooperatively in the same network to offer a more advanced set of ITS applications, forming *Cooperative Intelligent Transport Systems* (C-ITS) that benefit from those enhanced ITS services (C2C-CC, 2024). The *CAR 2 CAR Communication Consortium* (C2C-CC) categorizes the services proposed by ETSI as Day 1, Day 2, and Day 3+ releases based on how far in the future each one of them is expected to be deployed. Day 1 focuses on vehicles broadcasting their position and kinematics, while Day 2 deals with perception sharing among other road users. Day 3+ focuses on vehicular maneuver coordination and VRUs broadcasting data, thus assuming an active role in their safety (C2C-CC, 2019).

To cover the Day 3+ VRU active advertisement, ETSI published the 103 300 series of reports and specifications that lay the groundwork for a *VRU Awareness Message* (VAM) (ETSI, 2021a,b,c). VAMs are decentralized awareness messages sent periodically, advertising a VRU's presence to other stations in its range. This message is the basis for the *VRU Basic Service* (VBS) that proposes a functional application of the VAM, describing operations, message generation rules, and transmission trigger conditions. The generation time for these messages depends on channel occupation through the use of the *Decentralized Congestion Control* (DCC) mechanism (ETSI, 2018). Some use cases of the VBS are collision avoidance and intersection management (C2C-CC, 2023).

In a region densely populated with VRUs, plenty of individual VAMs can be exchanged, consuming spectrum resources and requiring plenty of processing from each *ITS Station* (ITS-S) in the local network, leading to higher overhead. As a solution, the VBS offers the *clustering* functionality, through which VRUs with similar positions and kinematics are grouped and represented by a single transmitting node called *cluster leader*. To advertise the set of members, the leader must include in its VAM an additional container, describing a bounding box that accommodates all the cluster participants. These objects can assume three different geometrical shape types: *circle*, *rectangle*, or *polygon* (ETSI, 2021c). Each of these is described in a VAM with different sets of parameters, and thus it can be assumed that the message size and its generation time can be affected by which shape is selected.

Despite ETSI 103 300 (ETSI, 2021c) defining plenty of functionalities, both for standalone operation and clustered VRUs, it does not offer strategies on how to produce the cluster bounding boxes, nor does it discuss the impact of selecting each of the available shape types on the performance of the service. The lack of explicit shape formation methodologies results in an implementation gap for the VBS, as in the current state, it is not possible to have a fully functional service with clustering functionalities. Furthermore, to the best of the author's knowledge, there is no other research that focuses on solving or investigating those gaps. As such, this thesis contributes to the state of the art by proposing strategies to improve the current VBS standard towards a more complete clustering functionality by defining strategies to build all the standardized VRU Cluster Bounding Box shapes using computational geometry techniques. Additionally, the concept of a *Cluster Map* is introduced as a supporting data object facilitating the formation and maintenance of cluster shapes.

Due to the scale and infrastructure necessary to do real-world testing of the performance of a V2X service, they require plenty of time, resources, and planning. Thus, a common approach is to first qualify the study of the behavior of the proposed messages and services through the use of computational simulations. This step is important to validate and improve a C-ITS concept because through them it is possible to do multiple iterations of various test cases, with the benefit of easily switching between a set of different constraints and parameters (C2C-CC, 2015). For this study, as the VBS is an ITS-G5 service, the simulation framework selected is Artery[1] (RIEBL; GÜNTHER, et al., 2015). This decision is based on the fact that this tool is open-source and freely available in a GitHub public repository, allowing access to the source code and permitting the author to develop and implement the VBS, the cluster management functionality, the Cluster Map data object, and the proposed bounding box shaping strategies.

To evaluate the results of the simulations, a series of metrics are selected to compare how each shape type affects the service. First is the average cluster area and number of participants. Then, with these parameters, the cluster density (i.e., the number of VRUs per bounding box area) is proposed, indicating how well the shape types represent a set of cluster members without occupying excessive space. Next is

[1] `http://artery.v2x-research.eu/`

the average VAM size in the Facilities layer, used to compare if the increment to the message is affected by each shape. The amount of simultaneous active clusters and the number of sent VAMs are observed, with a special focus on the cluster operation messages. Finally, the average Channel-Busy-Ratio is measured to evaluate if clustering in fact managed to reduce a potential channel overload. Multiple simulation runs are executed for each shape type and these parameters are confronted to evaluate the merits of every geometry, recommending situations in which their use is ideal.

1.2 RESEARCH GOAL

This research was conducted using the already existing extension to Artery developed by the Car2X Laboratory to support the standalone VBS, which the author has further improved to support the clustering features of the VBS and the shaping techniques proposed in this thesis. Simulations using a realistic traffic scenario have been executed to evaluate the impacts of clustering on a V2X network crowded with pedestrians equipped with communicating devices containing the VRU Basic Service. Various parameters were tested to observe their impact on the performance of the service and verify possible improvement suggestions. This led to the research goal:

Determine Cluster Bounding Box formation strategies for a VRU Basic Service, comparing the benefits of each shape type and suggesting possible applications in which each is preferable.

To fulfill this research goal the following objectives were considered:

Obj. 1: Implement the clustering containers for VAMs in Artery, utilizing the already existing objects in the library Vanetza;

Obj. 2: Develop cluster management functionality to handle VRU clustering operations;

Obj. 3: Define strategies to form the different Bounding Box shape types;

Obj. 4: Implement the functionalities of the VBS;

Obj. 5: Define metrics to compare the performance of the VBS using the different shape types;

Obj. 6: Model simulation scenario in Artery and run it with various seeds;

Obj. 7: Evaluate simulation results and identify use cases in which each shape is advised.

1.3 CONTRIBUTIONS OF THIS STUDY

This study led to a series of contributions to the state-of-the-art of the research on VRU Clustering. First, a paper titled *"To Cluster or not to Cluster: A VRU Clustering Based on V2X Communication"* (LOBO; BARBOSA DA SILVA; FACCHI, 2024) was co-authored by the student, in which the objectives **Obj. 1** and **Obj. 2** are tackled, laying the foundation for the next steps.

A paper titled *"What Is the Right Bounding Box of a VRU Cluster in V2X Communication? How to Form a Good Shape?"* (BARBOSA DA SILVA et al., 2024) was published in the Procedings of the *10th International Conference on Vehicle Technology and Intelligent Transport Systems* (VEHITS 2024)[2] under the DOI number 10.5220/0012699100003702. This paper was one of the nominees of the "Best Student Paper Award" of the event, confirming the relevance and uniqueness of this approach. It contains the development of the research done to fulfill objectives **Obj. 3** to **Obj. 7**, making it the central contribution of this Thesis.

Another paper, a post-publication on Springer's "Communications in Computer and Information Science" proceedings, has also been authored as an extended version of the VEHITS paper. Titled *"Boosting VRU Awareness: Bounding Box Strategies for V2X Clustering"*, this work is in the publishing phase, with a scheduled release in February of 2025. It uses the strategies and code proposed in (BARBOSA DA SILVA et al., 2024) to investigate the same use-case scenario, however with a longer simulation duration and evaluating more performance metrics, thus providing an extended analysis of the impact of clustering on the VBS when compared to the results in the previous paper. The simulation framework described in Chapter 4 and the results presented in Chapter 5 are derived from the research developed in this post-publication.

Finally, the code developed by the author of this Thesis can be listed as an additional contribution, since it was developed as a new set of modules to the Artery framework, enabling the simulation of VRU Clustering on the context of the VBS using this tool. Since this implementation of the cluster functionality contains original strategies developed by the Car2X Laboratory, the open-source release on GitHub is planned to a future date, after these approaches have been thoroughly documented through further scientific publications.

1.4 STRUCTURE OF THIS MASTER'S THESIS

The remainder of this document has the following structure: Chapter 2 offers an overview of the V2X communication technology, its protocols and the service that is used as basis for VRU Clustering. Chapter 3 presents a literature review, covering

[2] `https://vehits.scitevents.org/?y=2024`

the standards for the VRU Basic Service and its known implementation gaps, while also discussing how the concept of clustering is present in other fields. Chapter 4 describes the development of this thesis, presenting the methodology and the metrics selected to compare the Cluster Bounding Box strategies. Furthermore, it presents the framework used to simulate a V2X network to evaluate the clustering algorithms. Chapter 5 discusses the simulation results, comparing how each cluster shape performs in the same traffic scenario. Chapter 6 concludes this Master's Thesis and discusses possible future research directions.

2 BASICS OF V2X-COMMUNICATION

There are several V2X standards in active development, operating in the 5.9 GHz band (C-V2X, DSRC, and ITS-G5), at the forefront of the state-of-the-art. Each of them handles the steps of the communication stack differently, in many cases possessing their own protocols to deal with each layer. Thus, even though the present study focuses on a V2X service pertaining to the ITS-G5 standard, knowing the fundamentals of each specification is necessary to portray the current state of the vehicular communication field.

2.1 C-V2X

C-V2X is a cellular-network protocol, with a physical layer based on *Single Carrier Frequency Division Multiplexing Access* (SC-FDMA), that operates on a channel of 10 or 20 MHz divided into sub-frames of 1 ms, while in frequency it is divided in *Resource Blocks* (RB) of 180 kHz. A group of RBs on the same sub-frame is called a sub-channel and it contains a set of data and control information (MANNONI et al., 2019).

In LTE Release 14, Modes 3 and 4 were introduced, and with them, some interfaces that enabled the communication of vehicles, mainly on Vehicle-to-Vehicle and Vehicle-to-Infrastructure scenarios. In Mode 3 there is a base station responsible for the management of sub-channels, while in Mode 4 there is no such thing and the vehicles must select sub-channels by themselves using a common set of parameters, for example by having the same amount of sub-channels per sub-frame. In Mode 4 nodes communicate directly with each other in a decentralized structure (MOLINA-MASEGOSA; GOZALVEZ; SEPULCRE, 2020).

2.2 DSRC

Since DSRC is derived from a WLAN protocol, the Physical Layer is very similar to WiFi, with *Orthogonal Frequency Division Multiplexing* (OFDM) as the transmission scheme. It, however, operates at half-clock, with a sampling rate of 10 MHz. This results in a doubling of the timing parameters when compared to the WiFi standard, giving the protocol more robustness against fast-varying channels (MANNONI et al., 2019). Those potential rapid changes are associated with the high-speed movement capabilities of the vehicular nodes, leading to inter-carrier interference due to the Doppler Effect (FESTAG, 2015).

Another relevant addition from the DSRC is the *Wave Short Message Protocol* (WSMP), part of the WAVE set of standards, that contains both network and transport capabilities, providing an alternative for the IP and UDP/TCP protocols of a traditional WLAN. The WSMP is aimed at the use of lightweight messaging for single-hop networks, resulting in lower latency communication in Vehicle-to-Vehicle and Vehicle-to-Infrastructure applications. In the Facilities Layer, the main feature is the *Basic Safety Message* (BSM), which is sent periodically at a maximum rate of 10 Hz and contains dynamic information of the vehicle (FESTAG, 2015). This kind of message can then be used to raise the awareness of the other nearby connected nodes, thus composing a V2X service.

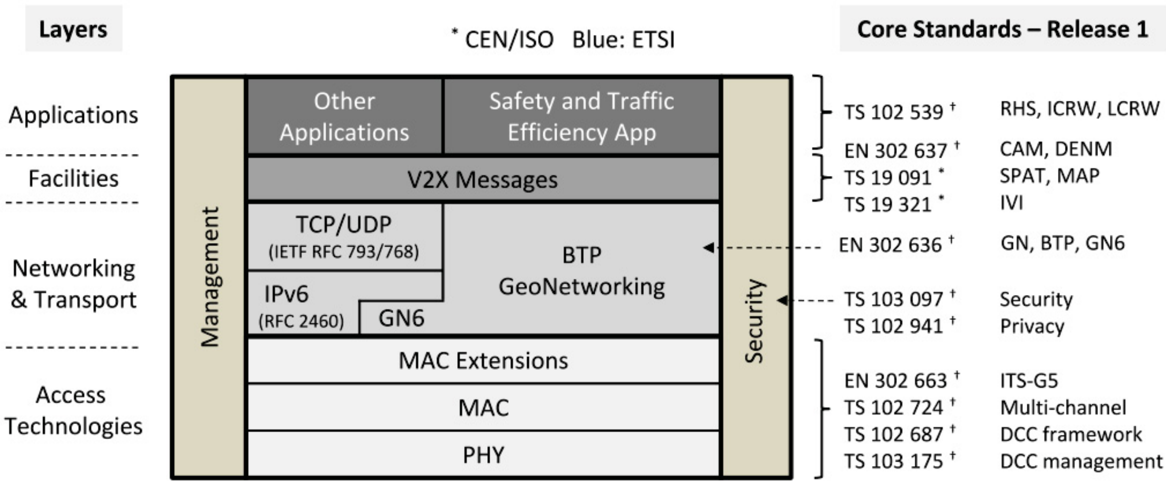
2.3 ITS-G5

Parallel to the development of the DSRC, the *European Telecommunications Institute* published several technical reports and standards that define the *Intelligent Transport Systems* and how they should behave. Those systems can be either vehicles or other road users such as pedestrians, cyclists, first responders, and even the infrastructure nearby, only being required to have compatible communication capabilities for vehicular applications, mainly through IP-based communications and Ad Hoc direct communication (ETSI, 2022). This resulted in the creation of an alternative WiFi-based V2X protocol that would later be called ITS-G5 in a reference to the ITS concept applied to the 5.9 GHz band.

Even though it was developed as an alternative to the North American standard, the ITS-G5 contains similarities to the DSRC. For example, the use of 802.11p as the base for the Access Layers (ETSI, 2019a) and the support for IPv6 and UDP/TCP. A particularity of the standard is the fact that it is subdivided into four different frequency band groups (from A to D) bands depending on its application. Each group has a different allocated frequency interval, with the most relevant for this study being the ITS-G5A, which is dedicated to road safety applications and is allocated in the 5.875 to 5.905 GHz range (ETSI, 2013a). The other groups are: ITS-G5B, which operates between 5.855 and 5.875 GHz and is used for ITS applications not related to safety; ITS-G5C occupying the interval 5.470 to 5.725 GHz, sharing the band with Radio Local Area Network services; ITS-G5D found in the range 5.905 to 5.925 GHz and is envisioned for future ITS applications. An overview of the stack is presented by FIGURE 2.

As seen in FIGURE 2, some of the particularities of this standard is the use of the IEEE 802.11p on the Access Layers, composed of the *Physical* (PHY) and *Media Access Control* (MAC), implementing OFDM with 10 MHz frequency channels (half-clock operation) at the PHY Layer to account for the inter-carrier interference of highly mobile ITS-Stations (FESTAG, 2015).

FIGURE 2 – ITS-G5 protocol stack (FESTAG, 2015)



At the MAC Layer, it implements *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA), which is a protocol aimed at wireless devices to avoid collisions due to hidden nodes. With this technique, an ITS-Station will first sense the channel to evaluate if it is idle (i.e., if it is not occupied by another transmission) before attempting to send a message. This "sensing" is done by detecting if the power from received signals is higher than a set threshold called *Clear Channel Assessment*, which would indicate the channel as busy (MOLINA-MASEGOSA; GOZALVEZ; SEPULCRE, 2020). In that case, the station will wait a randomized backoff period before trying to transmit again (ETSI, 2019a), and by doing so, it avoids overloading an already occupied medium with more messages.

On the Network Layer of the stack, the main feature is GeoNetworking, a routing protocol for multi-hop communication (ETSI, 2014). The main feature is the sending of packets based on the geographical position of the transmitter and receiver nodes, regardless of the range of a single hop. It is assumed that every device knows its position and has knowledge of the nearby network topology. Furthermore, it considers that every packet contains a set of coordinates as part of the identification of its destination. Thus, the forwarding of the data by a node is based on the packet address and its view of the network, with an autonomous route decision that has low communication overhead. This strategy is perfect for vehicular networks, which, due to their dynamic nature, must have the lowest delay possible for the received message to be usable. Of course, this time constraint is dependent on each application, for example, a general awareness notification and a pre-crash message will have different tolerances regarding how old a piece of information can be while still being useful. However, as a reference value for its *Road Hazard Signalling* (RHS) application, ETSI affirms that for critical road safety services, the maximum end-to-end delay must be 300 ms to avoid wrong decision-making related to outdated information (ETSI, 2013c). The delay mentioned in this specification depends on the performance of both the ITS-Station that is sending the message and also of the receiving device, while also being related to the characteristics of the wireless network (e.g., channel load and presence of obstacles).

Another important aspect of the ITS-G5 architecture is the Facilities Layer, a middleware where the application-specific V2X messages are generated. This layer supports the ITS services from the Application Layer, while also enabling data exchange to other layers of the protocol stack (Network, Transport, Management, and Security), and also providing interoperability to other systems and networks through gateways. Facilities are divided into Common or Domain facilities depending on whether the offered functions are to be used as a basic core service (e.g., position and time management) for all devices or whether they support more specific applications in some ITS Stations (ETSI, 2013b).

An interesting and unique aspect of the ITS-G5 stack is the *Decentralized Congestion Control* (DCC) mechanism. This feature is responsible for managing a device's message transmission schedule, altering how often it can send a message based on how congested the wireless channel is. This decision is based on the *Channel Busy Ratio* (CBR) metric, given by a ratio between 0 and 1 that indicates for how long the channel was occupied during a certain time interval. In this context, a channel is considered "busy" when occupied by message transmissions. By default, the DCC is engaged when the CBR is above 65%, minimizing the likelihood of congestion by managing the communication priority of the devices and avoiding that a small group of devices can occupy all the available resources. This approach is necessary on V2X, since they are decentralized networks (i.e., without a common network controller that grants permissions for each device to transmit) and with an always-changing number of connected nodes in a given area. This functionality operates in diverse layers, such as MAC, Facilities, and Management, since it requires the knowledge of the applications running on the device (ETSI, 2018).

2.4 C-ITS SERVICES

ITS services themselves can already offer positive impacts with applications such as safe-distance alerts and lane-keeping support. However, vehicular communication for ITS can do much more when these different devices exchange messages with each other. This type of interaction is called *Cooperative Intelligent Transport Systems* (C-ITS) and it can be used in order to coordinate actions as part of a collective service, which benefits from data with higher availability, quality, and reliability (ETSI, 2022).

The C-ITS also made viable the creation of more complex services, such as the already standardized Cooperative Awareness Basic Service (ETSI, 2019c) and Decentralized Environmental Notification Basic Service (ETSI, 2019d). These are based, respectively, on the *Cooperative Awareness Message* (CAM) and *Decentralized Environmental Notification Message* (DENM). Briefly speaking, the CAM is used to exchange periodical data on positioning, speed, and heading in order to create a collective awareness of the actors present in a given area. The DENM is an event-triggered message, that is activated in situations such as a vehicle suddenly braking, notifying the other nearby nodes of a potential risk of collision or irregularity on the road.

Based on the C-ITS deployment roadmap developed by the *CAR 2 CAR Communication Consortium* (C2C-CC), the Basic Services based on CAMs and DENMs compose the Day 1 release (C2C-CC, 2019). This phase, also called "Awareness Driving", involves the communication between vehicles and road infrastructure to broadcast simple status and environmental notifications in a short range. This can be used to issue, for example, warnings about a possible collision risk at an intersection or to alert nearby cars to an approaching emergency vehicle.

Day 2 deployment, or "Sensing driving", enhances the V2X's functionalities by allowing the sharing of data acquired from on-board sensors through the *Collective Perception Message* (CPM) and its Collective Perception Service (ETSI, 2019b). The service aims to improve awareness by notifying other road users about detected vehicles or objects on the road that may otherwise not be perceived. For example, in situations where the line of sight is obstructed or the road user does not have a communication device to broadcast their status (C2C-CC, 2019).

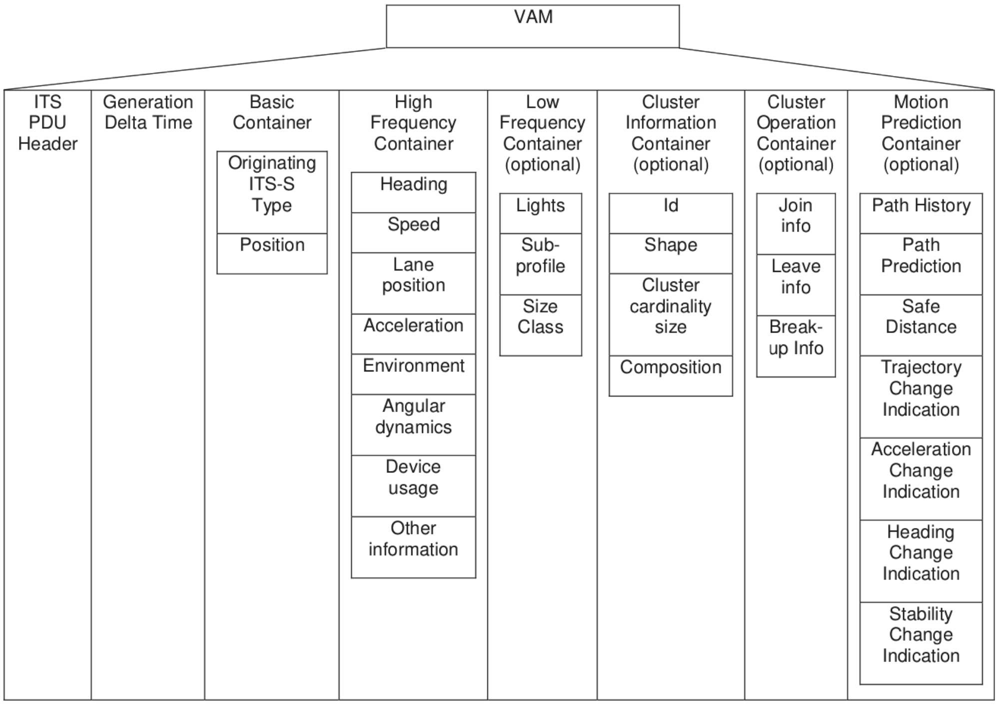
The Day 3+ phase is labeled "Cooperative Driving" and focuses on intention sharing by vehicles with automated driving capabilities. Messages exchanged on this deployment also aim to enable the implementation of maneuver coordination services, such as lane merging, with the hopes of reducing conflicts within the involved nodes. Furthermore, by knowing the intentions of each road user, traffic light controllers can have a more dynamic phase duration and thus provide a more seamless traffic flow based on expected lane occupation (C2C-CC, 2019).

Another Day 3+ service that is in the process of becoming an ETSI standard, and that is the focus of this study, is the *VRU Basic Service* (VBS). As the name suggests, it aims to provide a set of functionalities to enhance the safety of VRUs through the use of wireless awareness messages, making them active traffic participants instead of passive ones (C2C-CC, 2019). This is a very relevant addition since this behavior can help increase the awareness of VRUs in a transport system, whose necessity is justified by the number of pedestrian fatalities discussed in Chapter 1.

2.5 VULNERABLE ROAD USER AWARENESS MESSAGE

The base of the Vulnerable Road User Basic Service is the ETSI 103 300, which is a series of Technical Reports and Specifications aimed at pedestrian safety applications (ETSI, 2021a,b,c). In those reports, the general specifications of the *Vulnerable Road User Awareness Message* (VAM) are described. The VAMs are envisioned as awareness messages sent by pedestrians, cyclists, and other VRUs to notify other nodes of their presence, location, speed, heading, etc. The general structure of these messages can be observed in FIGURE 3.

FIGURE 3 – Structure of a VAM (ETSI, 2021c)



The first four columns of FIGURE 3 indicate the mandatory components of the VAM:

- **ITS PDU Header:** contains general data elements of the VAM, such as station ID and protocol version;
- **Generation Delta Time:** stores the time in milliseconds elapsed since the ITS epoch 2004-01-01T00:00:00:000Z, relating to the time for the generation of the message itself;
- **Basic:** contains an identifier to the type of ITS station (e.g., pedestrians, cyclists, motorcycles, etc) that originated the packet. Furthermore, it also contains the reference position, confidence, and altitude;

- **High Frequency:** VRU dynamic data that changes rapidly, such as speed, heading, longitudinal acceleration, lane position, etc.

The next four containers are optional and related to data that will only be included in some VAMs based on elapsed time or specific functionalities being enabled. The *Low Frequency Container* aggregates the slow-changing or static information of the VRU, and because of their nature, they are not transmitted as frequently as the *High Frequency Container* (ETSI, 2021c).

The *Cluster Information and Operation Containers* are responsible for the definition of the physical characteristics and the event handling of a cluster, respectively. Clustering is a grouping technique through which a set of similar devices that are close to each other are transformed into a single entity. One of the VRUs from the group is then defined as the cluster leader and will start to transmit packets that represent all the members. Clustering is implemented to reduce the wireless channel occupation with near-identical data. Since this is the main topic of the Thesis, those containers will be discussed in more detail in section 2.6.

The last container is the *Motion Prediction Container*, which includes a history of the motion of the VRU, including path and acceleration patterns. This is recorded to produce possible trajectories that can then be utilized to predict possible risks of collision. Another possible usage is to more efficiently evaluate which VRU nodes can be grouped in a cluster based on the similarity of their predicted routes.

2.6 CLUSTERING IN THE VBS

The technical specification ETSI TS 103 300 (ETSI, 2021c) determines that cluster creation on a VBS is triggered when a minimum of 3 to 5 VRUs with ITS capabilities are sufficiently close with maximum distance of 3 to 5 meters, and heading in a similar direction with similar speed, both constrained to a maximum difference of 5%. When these conditions are met, a ITS-Station that detects this opportunity can assign itself the role of a cluster leader if it possesses both transmitter and receiver functionalities within the VRU Basic Service. The current ETSI specification does not elaborate or provide details about the process of leader selection beyond this point. Thus, for simplicity, it is adopted an approach called "first come, first cluster" in this implementation, which simply means that the first capable VRU that detects a clustering opportunity will be the leader.

After cluster creation, the *Cluster Information Container* (CIC), as presented in FIGURE 3, is added to a VAM by the leader. This structure contains data that describes the cluster, such as a unique ID number, the shape of the bounding box (rectangular, circular, or polygon) that encompasses all the participant nodes, cardinality (i.e., amount of devices grouped), and the profile (e.g., pedestrian, cyclist, moped rider, etc) contained. The parameters that are needed to form each shape on the VAM are given below and also graphically represented in FIGURE 4. Once again, ETSI 103 300 does not describe methodologies on how to produce these geometries.

Circular:

C.a - Node center point;

C.b - Radius;

Rectangular:

R.a - Node center point;

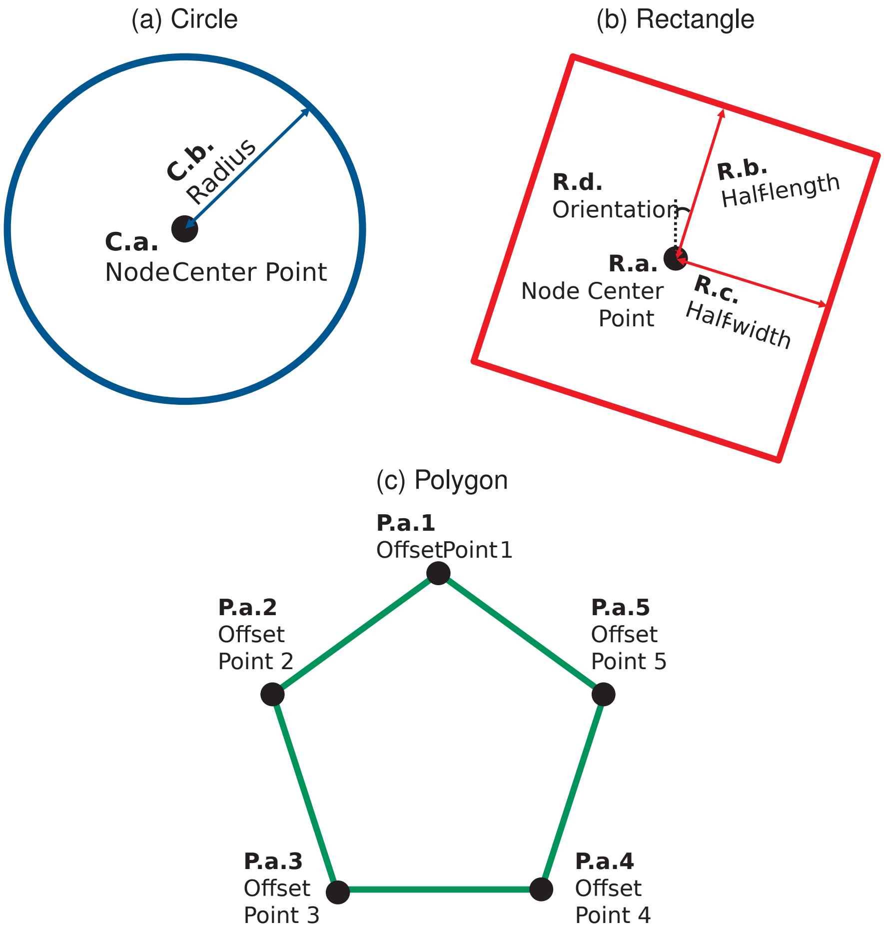
R.b - Half-length to the center point;

R.c - Half-width to the center point;

R.d - Angle of orientation

Polygon: List of sequential offset points (*P.a.1, ..., P.a.n*) describing the vertices of the shape. Each offset value is calculated based on two consecutive points on the sequence. There is no specified upper bound according to ETSI 103 300.

FIGURE 4 – Cluster shapes and their parameters



The *Cluster Operation Container* (COC), as presented in FIGURE 3, is the structure responsible for the events related to the cluster state and composition. Using this container potential member nodes can send messages to alert a leader that they want to join or leave the group. Also with the COC, the leader can send a VAM indicating that the cluster will be broken up. Both the breaking up and the leave cluster operations must also provide a reason for this action in the message.

For a VBS to contain the clustering functionalities detailed in (ETSI, 2021c), both of these containers must be utilized. Thus, the development and implementation of both in Artery is crucial for the proposed study, being a research gap to be detailed on Chapter 3.

Part 2 of the ETSI Specification (ETSI, 2021b) highlights clustering as one of the solutions for the Functional Communication Requirements related to channel efficiency and congestion control due to an increasing number of transmitting nodes. This solution can be suggested because each cluster created can reduce significantly the number of VRUs transmitting their VAMs in a given region.

3 LITERATURE REVIEW

To establish the baseline of how the clustering in the VBS should be functionally structured, a literature review is necessary to identify research gaps and potential points of improvement to the standard. With that in mind, this chapter is divided into two main focus areas: "Vulnerable Road User Basic Service Technical standards" and "Related Works". The first contains the current ETSI standards regarding the VBS and adjacent areas, pointing out what each publication adds to the service and which topics lack detailing, therefore requiring the author to suggest solutions to supplement the gaps. The second covers current research on VAMs and adjacent fields that can be used to determine clustering strategies that can be adapted to the VBS use case.

3.1 VULNERABLE ROAD USER BASIC SERVICE TECHNICAL STANDARDS

ETSI published some technical reports and specifications labeled ETSI 103 300 that define how the VRU Awareness messages and VRU Basic Service should be structured and what functionalities they should offer. Those specifications provide a good overview of how ETSI envisions the VBS to be included in the ITS-G5 paradigm, presenting possible multimodal use cases in which VAMs (and other standardized V2X messages) can be used to improve the safety of road users in a certain area.

3.1.1 Use Cases definition

ETSI 103 300 Part 1 (ETSI, 2021a) offers diverse definitions and terminologies that will be used in the whole series of technical papers about the VRU Basic Service based on the ITS-G5 technology. The report lists a series of actors to be considered Vulnerable Users, which can be categorized as:

- Pedestrians, emergency responders, safety and road workers;
- Riders of two-wheel motorcycles;
- Riders of three-wheel mopeds and quadricycles limited to 45 km/h;
- Wheelchairs users, prams, bicycles (includes e-bikes limited to 25 km/h);
- Users of skateboards and other personal transporters;
- Animals that can offer a safety risk to other VRUs or vehicles.

The main feature of (ETSI, 2021a), however, is the presentation of different use cases that revolve around the usage of ITS applications to enhance the safety of VRUs. Those suggested scenarios are divided into six categories:

Category A: VRUs communicate directly with each other.

Category B: VRU communicates directly with a vehicle.

Category C: A vehicle detects a VRU through its sensors and communicates the other vehicles.

Category D: A *Roadside Unit* (RSU) detects a VRU through its sensors and signals that to other vehicles.

Category E: A central system monitors VRUs (by direct communication or indirectly through an RSU's sensors) and sends alerts/instructions to vehicles or other VRUs to avoid collisions.

Category F: An RSU monitors VRUs by communicating with them, notifying vehicles in case of risk of collision.

All the communicating nodes involved in a use case can also be called actors and it is important to point out that for these scenarios to be possible, a VRU application must exist not only on the equipped VRUs but also on the other types of actors, such as vehicles or even infrastructure devices (RSUs). This is considered because some of the use cases take into account VRUs with limited or absent V2X communication capabilities, thus their protection involves their detection through different ITS stations. Another critical topic is the listing of situations that offer a risk of a collision involving a VRU and a vehicle. Those accidents often occur in one of the following scenarios: when a VRU crosses a road in a perpendicular direction to a vehicle, during an overtaking maneuver, or when either the VRU or the vehicle makes a turn in front of the other road user. The first of these is the most frequent scenario in which pedestrians collide with vehicles, especially if the direct view of the VRU is occluded by an object (ETSI, 2021a). This knowledge can then be used to develop simulation scenarios that best resemble the real-world applications of the VBS proposed by ETSI.

Despite (ETSI, 2021a) describing several different use cases, little is discussed with VRU clustering in mind. This means that the VBS examples presented by ETSI involve a small number of road users, and thus a cluster formation event would not be triggered. And even if it does, it will not contain enough transmitting nodes to produce a significant network load in the wireless channel, which is more manner to evaluate the benefits of the technique. Thus, some adaptations are necessary to simulate the scenario with VRUs crossing a road so that clustering can occur. For this, a simulation of Category B (i.e., VRU communicates with a vehicle) is best suited since nearby vehicles need to be aware of the VRUs' positions and cluster status, leading to a setup that resembles *Use Case B2* (UC-B2) from (ETSI, 2021a). Additionally, the VRUs must communicate with each other, broadcasting their status to form, join, and disband clusters, which could indicate some overlap with Category A.

3.1.2 Functional Architecture and Requirements definition

ETSI 103 300 Part 2 (ETSI, 2021b) continues to explore the categories of scenarios involving distinct types of actors defined in (ETSI, 2021a). This part aims to present the requirements for the exchange of messages in each of these situations to be viable and safe, listing the severity of the risks if the message is compromised. The VBS use cases presented in this part of the standard follow this flow:

1) **VRU detection and tracking:** Done indirectly by other node's sensors or directly through self-positioning;
2) **VRU risk assessment:** If any other road user or traffic situation can offer any danger to the VRU, evaluate if the VRU's position and kinematic data should be transmitted;
3) **Verification of the messaging environment:** Evaluate whether the VRU itself should transmit or not, for example because it is a passive node on a cluster;
4) **Transmission of VRU's data:** Can be done by the VRU itself, its cluster leader, or another node that detected it in Step 1;
5) **Receiver nodes evaluate the risk:** Based on data from messages sent by other devices, each receiver constructs a local dynamic map and assesses the likelihood of an accident;
6) **Protective action:** The device sends a warning message to protect the VRU. In the case of an automated vehicle, it performs an active maneuver.

This process serves as a useful blueprint, in conjunction with UC-B2, for developing a scenario that incorporates VBS with clustering. Moreover, (ETSI, 2021b) features functional and operational requirements to establish the VBS. Of particular relevance to this study are:

OSYS05: Considering that a VRU's position and velocity tend to change frequently, the age of the data exchanged in a VBS becomes crucial for maintaining accurate awareness of nearby road users. ETSI recommends a minimum end-to-end latency time of less than 300 ms and data sampling at 10 Hz to achieve this goal. It is essential to note that these parameter values are not standardized but are given by ETSI.

OSYS06: Still taking into account the age of data, it is important to consider that the position of a VRU may not be exactly the one broadcasted at a given moment. A suggested way to correct this uncertainty is by interpolation based on the trajectory and velocity contained in the message.

Another relevant topic covered by (ETSI, 2021b) is characterizing the VRU pedestrian profile. Its typical speed value is 5 km/h, with a communication radius of 50 m. These values can be used to make simulations more adherent to the ETSI standard.

One final aspect that deserves attention is the introduction of the *Local Dynamic Map* (LDM) on the VBS. This feature is handled on the Facilities Layer and can be described as a list of all the dynamic objects a device perceives based on the messages it has received. Those objects can be road users or infrastructure equipment e.g., traffic lights that provide dynamic data elements, which must be constantly updated. For this study, this functionality is fundamental for cluster management, since it can be used as the main tool to store the position of VRUs. This data is then used to trigger the formation of clusters and to determine the shape of bounding boxes.

3.1.3 Specification of VRU Awareness Basic Service

The functional aspects of the VAMs and the VBS are described in ETSI 103 300 Part 3 (ETSI, 2021c). Also, within the scope of the VBS, the involved communication interfaces are message formatting specifications are presented. The containers that form the VAMs are introduced, including parameter names, units, and range of accepted values. Those guidelines, along with the data elements in the ASN.1 format, provide the basis for the implementation of the VBS on Artery.

Regarding clustering, each VRU device can assume one out of four different roles: *Idle, Active Standalone, Active Cluster Leader, and Passive*. Idles are devices that are not considered VRUs at a particular moment, such as a person on a bus, and that therefore are not currently sending VAMs. Standalones are nodes that send messages (VAMs or CAMs) to represent only their presence, while the Cluster Leaders are VRUs that broadcast data regarding the whole cluster. Finally, Passives are the VRUs that do not send messages because they are cluster members, and thus they refrain from sending VAMs in favor of the Leader's broadcast.

Responsible for switching the active roles and carrying out their functionalities is an entity called VRU Cluster Management from the Facilities Layer. The manager monitors a set of parameters determined in the VBS to decide when it is necessary to perform cluster formation, joining, leaving, and breaking up operations. Key values that are relevant to this study are defined by (ETSI, 2021c) and given in TABLE 2.

In TABLE 2 the rows from the Cluster Decision category describe parameter values that determine if a cluster should be created/disbanded based on the number of nodes in a given area. For example, when a VRU notices through its LDM that there are no active clusters in a range *maxClusterDistance*, it looks for more VRUs with similar kinematics. If in that range there are at least *minClusterSize* VRUs with a difference of speed below the threshold *maxClusterVelocityDifference*, it decides to create a new

TABLE 2 – Cluster parameter values

	Parameter name	**Meaning**	**Value**
Cluster Decision	numCreateCluster	Minimum amount of nodes to form a cluster	3 to 5 VRUs
	minClusterSize	Minimum cluster size (only used at cluster creation)	1 VRU
	maxClusterSize	Maximum cluster size	20 VRUs
	maxClusterDistance	Maximum distance between VRU and edge of the cluster	3 to 5 m
	maxClusterVelocityDifference	Maximum velocity difference within the cluster	5 %
Cluster Membership	timeClusterJoinNotification	Standalone VRU emits a warning of a join operation for this duration	3 s
	timeClusterJoinSuccess	Standalone VRU waits this duration for leader to confirm its addition to the cluster	0.5 s
	timeClusterBreakupWarning	Leader VRU emits a warning of a break up operation for this duration	3 s
	timeClusterContinuity	Passive VRU waits this duration for a new VAM from the Leader before considering that the connection is lost	2 s
	timeClusterLeaveNotification	Passive VRU emits a warning of a leave operation for this duration	1 s

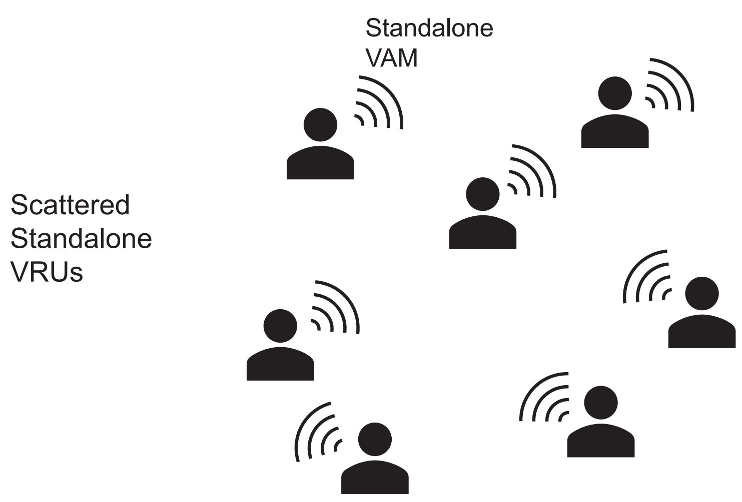
cluster. On the other hand, Cluster Membership parameters relate to the duration that a VRU remains transmitting or waiting for VAMs before changing their role.

3.1.4 Cluster's lifecycle

In this subsection, a cluster's lifecycle is presented, with supporting figures to better convey the events and messages exchanged. In them, the color black is used to indicate standalone VRUs and their VAMs, blue is related to a cluster leader and its management operations, red represents cluster operations and the VRUs triggering them, while grey is used for passive VRUs.

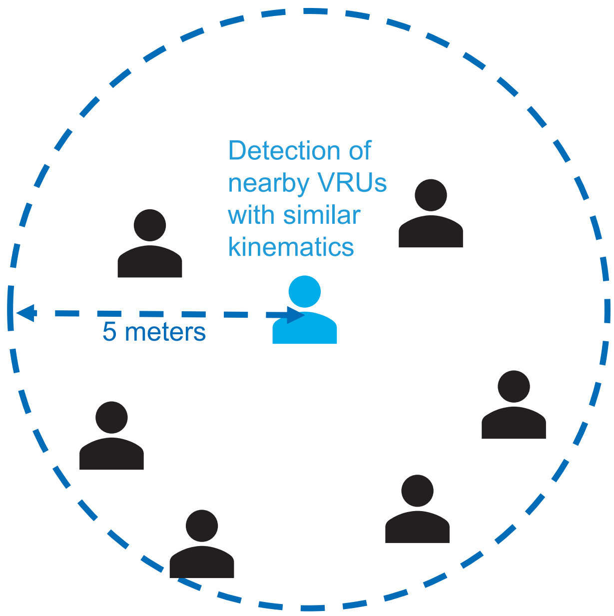
Based on the specifications and parameters from ETSI 103 300 Part 3 mentioned on subsection 3.1.3, a cluster can be created in networks with VRUs capable of sending and receiving VAMs and that are willing to cluster. Initially, these VRUs are assigned a standalone role, sending VAMs that disclose only their own position and kinematics, as shown in FIGURE 5.

FIGURE 5 – Standalone VRUs in a traffic system



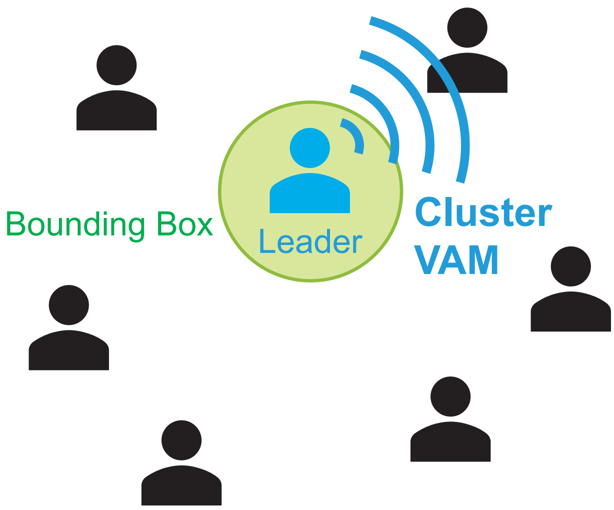
These standalone VRUs constantly parse VAMs from other VRUs, comparing their position and kinematics to themselves, searching for compatible standalone VRUs that fit the parameters *maxClusterDistance* (default: 5 meters) and *maxClusterVelocityDifference* (default: 5%) from TABLE 2. Once a VRU determines that there are at least two other standalone VRUs that suffice both these values, it concludes that a cluster of minimum size *numCreateCluster* (default: 3 VRUs) can be created. This process is illustrated by FIGURE 6.

FIGURE 6 – A VRU detects compatible VRUs in range



The first standalone VRU that detects this opportunity, changes its role to cluster leader and generates a bounding box containing only itself. It then starts sending VAMs indicating a clustering opportunity to the nearby nodes in the traffic system. In FIGURE 7, the initial bounding box is indicated in green.

FIGURE 7 – The Leader VRU broadcasts a clustering opportunity

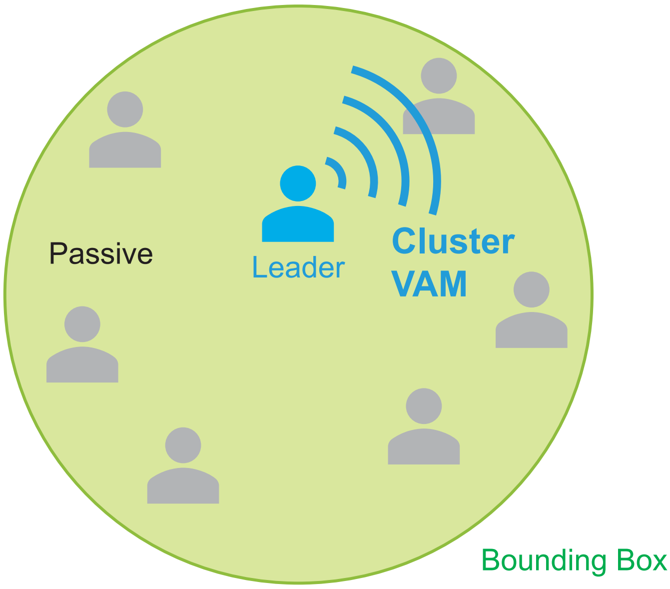


Upon receiving Cluster VAMs, the remaining standalone VRUs can decide whether they want to participate in the cluster or not based on the speed difference and distance. If they decide to take part on it, they must send VAMs indicating their intention of joining said cluster for the duration *timeClusterJoinNotification* (deafault: 3 s). FIGURE 8 shows a group of VRUs sending join VAMs to a compatible leader.

FIGURE 8 – Remaining VRUs can request to join the cluster

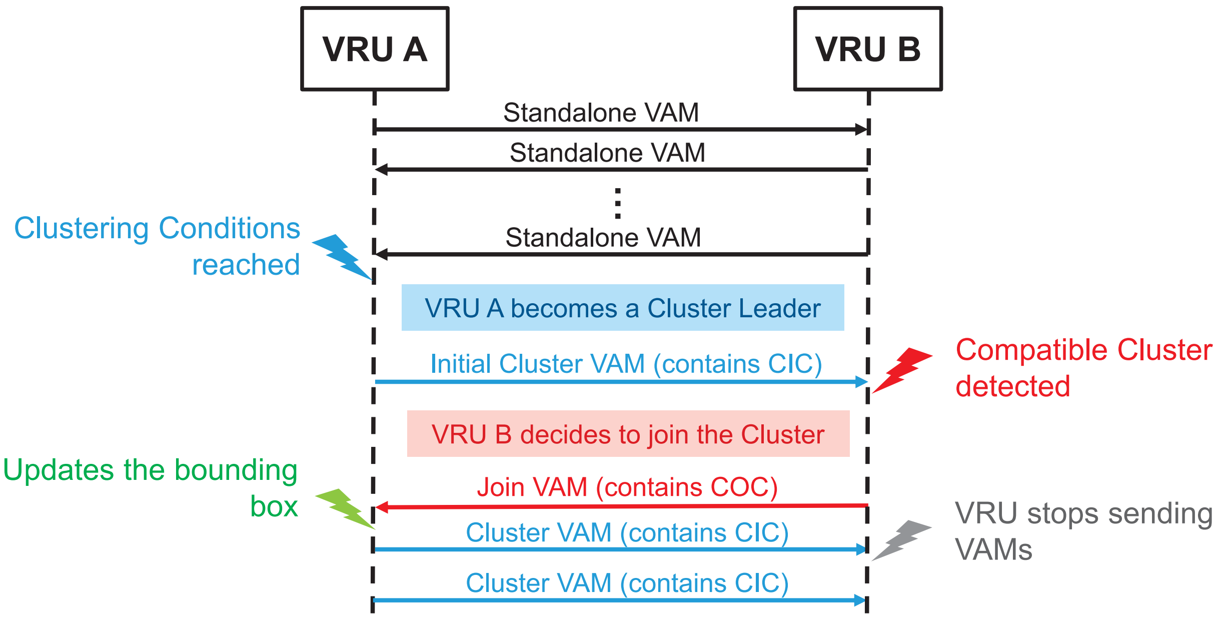
The leader must then receive these join notifications and adjust its bounding box to cover the compatible VRUs, adopting them as new cluster members. It must always verify if the amount of contained members respects the upper limit *maxClusterSize* (default: 20 VRUs). After updating the bounding box, the leader sends VAMs containing the new cluster shape and cardinality. The VRUs that notified the intention to join a cluster wait for the duration *timeClusterJoinSuccess* (default: 0.5 s) for the VAMs from the leader of the target cluster to be updated. Through these messages, if the standalone VRU confirms that it is now clustered, it must assume the passive role, ceasing to transmit messages. The now passive VRUs are shown in grey on FIGURE 9.

FIGURE 9 – Leader adjusts the cluster bounding box to fit the new members



At this point, a cluster is maintained for an indeterminate amount of time. The leader keeps broadcasting the cluster data, while also managing any new join notifications. The exchange of messages between a VRU cluster leader (A) and a standalone VRU turned cluster member (B) during a cluster formation event are shown in FIGURE 10.

FIGURE 10 – Messages exchanged between a potential leader (VRU A) and a compatible standalone VRU (B) during a cluster creation event



While in the cluster maintenance state, the passive nodes monitor the messages sent by the leader, verifying if they themselves are still compatible with the cluster, and also measuring how much time has been elapsed since the last cluster VAM based on the *timeClusterContinuity* (default: 2 s) parameter. If the passive VRU determines incompatibility or that the connection to the leader has been lost, it will decide to leave the cluster, sending a notification for a fixed amount of time *timeClusterLeaveNotification* (default: 1 s). This operation is indicated by FIGURE 11, in which a VRU upon noticing that it is further than 5 meters from the bounding box center, sends a leave VAM. After the leave notification duration has been elapsed, the VRU resumes its operations as a standalone node. This exchange of messages from a leader (A) and a member VRU (B) that desires to leave a cluster is summarized in FIGURE 12.

FIGURE 11 – Cluster member sending a leave notification

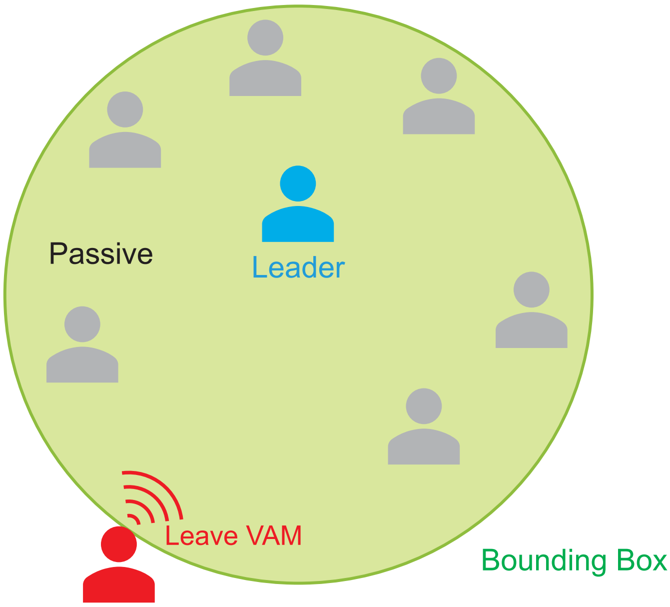
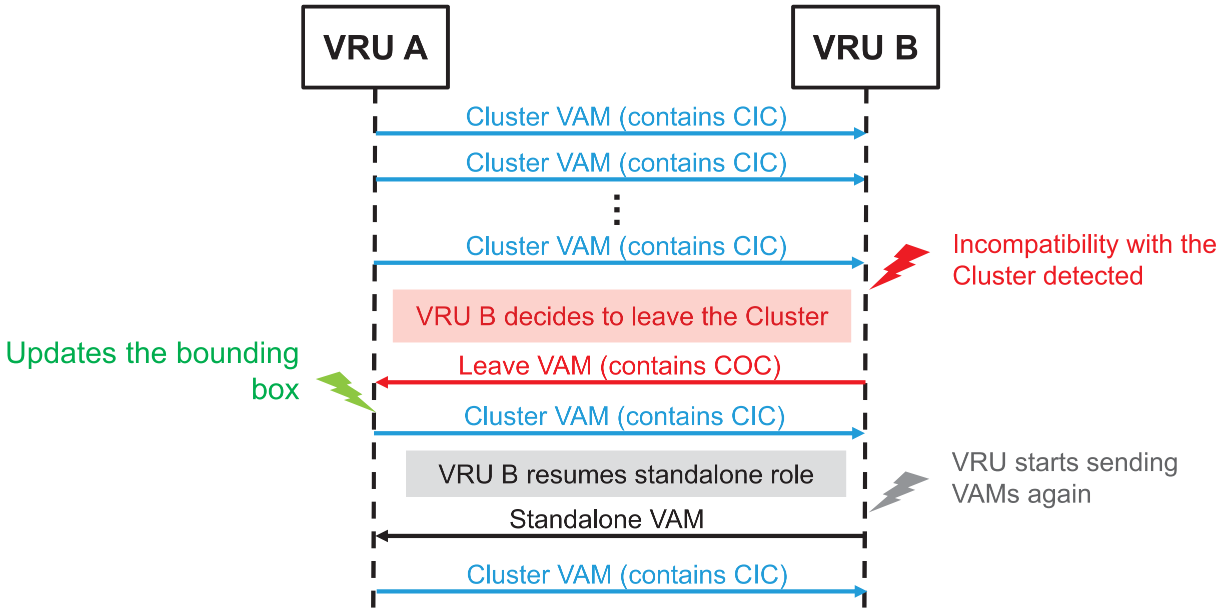


FIGURE 12 – Messages exchanged between a cluster leader (VRU A) and a member (VRU B) during a leave operation



If a leader determines that the cluster must be dissolved, for any of the reasons listed in (ETSI, 2021c), it must send a warning during the *timeClusterBreakupWarning* (default: 3 s), letting its members trigger a leave operation and resume standalone mode before the cluster is broken-up, as illustrated in FIGURE 13. After this period, the leader will also return to the role of standalone. This process is indicated in FIGURE 14, where a leader (A) notifies a member (B) that the cluster will be terminated. From this point, new clusters can be formed involving the now dispersed former member VRUs, or they can choose to join other existing clusters.

FIGURE 13 – A leader sends a cluster breakup notification due to insufficient cardinality

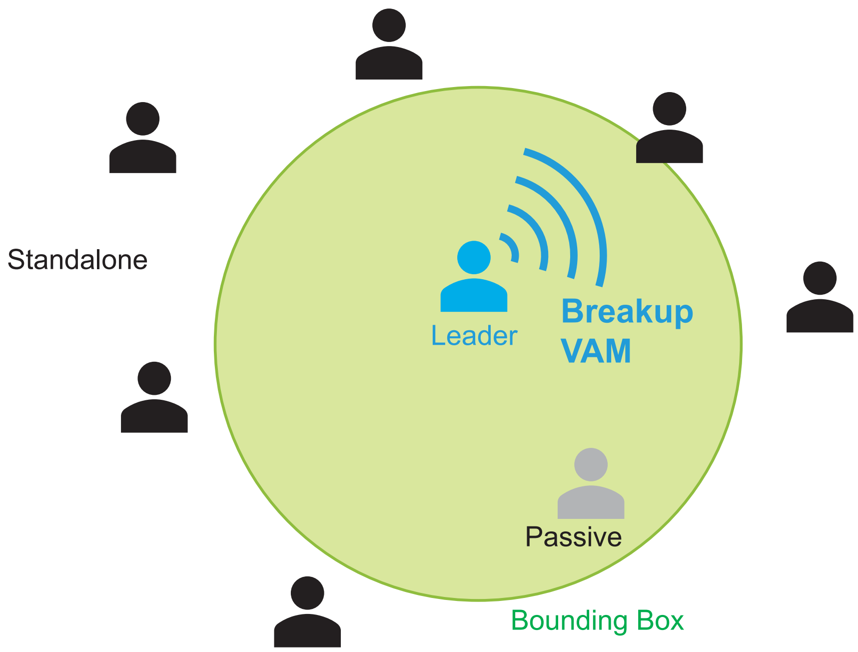
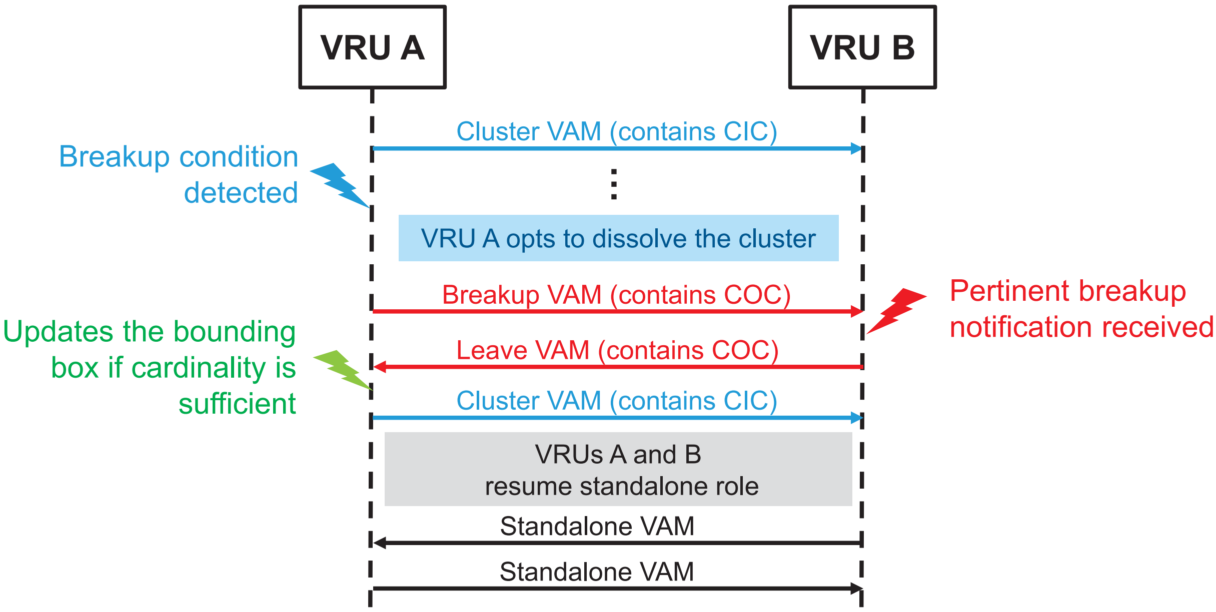


FIGURE 14 – Messages exchanged between a cluster leader (VRU A) and a member (VRU B) during a breakup operation



3.1.5 Functional Gaps on the VBS

Despite the specifications from (ETSI, 2021c) however, some aspects regarding cluster formation and its maintenance are left vague. This leads to some research gaps:

RG 1: How to select the cluster leader?

RG 2: How to determine the initial cluster bounding box shape and size, considering that it must contain only the leader?

RG 3: Is the generation time of leader cluster VAM messages adequate?

RG 4: How long should a leader wait for join notifications before deciding to break up a cluster due to insufficient cluster members?

RG 5: How to prevent the node's kinematics and geo-position data inaccuracies from producing unsafe bounding box edges?

RG 6: How to form each cluster bounding bounding box type?

Since these aspects are fundamental to developing a functional service, it fell upon the artery-VRU project development team to devise some possible alternatives to cover those gaps in the proposed standard. The research gaps **RG 1**, **RG 2**, **RG 3**, **RG 4**, and **RG 5** are covered through the suggestions first presented in Lobo, Barbosa da Silva, and Facchi (2024), in which the author of this Thesis has contributed as a co-author. This Master's Thesis focuses in discussing and covering the remaining gap **RG 6**, which will be presented in the Methodology chapter of this document.

As noted in Lobo, Barbosa da Silva, and Facchi (2024), the current specification for the VRU Basic Service has functional gaps hindering complete clustering functionality. This study incorporates the following enhancements proposed in this paper: establishment of a strategy to form the initial bounding box, use of an exclusive message generation time for *cluster leaders*, duration limit for the cluster creation process, and introduction of a shape padding method. These solutions are presented in the remainder of this subsection, including an additional improvement introduced by this Thesis.

In this implementation, the leader follows a *"first come, first cluster"* rule to cover the **RG 1**. This means that the first VRU equipped with the VBS and with clustering enabled, that detects sufficient compatible VRUs in its range, will assume a cluster leader role. This verification is done during the normal operation of the VBS and the cluster creation process is conducted as described in subsection 3.1.4.

Since the standard does not offer a method to construct the initial cluster bounding box as noted in **RG 2**, it is proposed that this geometry shall always be a circle with a radius of half the distance to the closest compatible VRU. This is because circles need the least number of data elements to be described, and the suggested radius ensures non-negative areas capable of covering only the leader.

On heavily occupied channels, consecutive message generations from a VRU using the VBS might take up to 5 s. If the VRU in question is a *cluster leader*, this would lead the members to wrongfully trigger a "leader-lost" leave operation due to a lack of received leader VAMs in the span of *timeClusterContinuity* (default: 2 s). To prevent this issue, indicated in **RG 3**, the reduction of the maximum generation delta time (*T_GenVamMax*) for *cluster leaders* from 5 to 2 s was proposed.

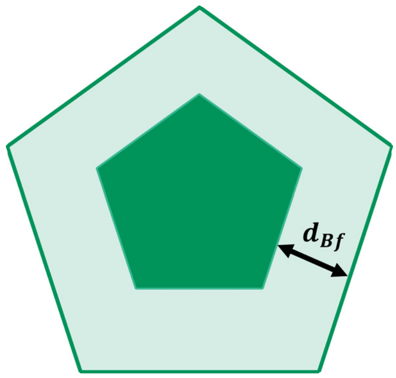
Another suggested addition is a "Cluster creation failed" event. Currently, a leader creating a cluster will wait indefinitely for join VAMs while the cardinality is below *numCreateCluster*, as in **RG 4**. This state, if kept for prolonged durations, is inefficient as it drains the VRU's power resources and overloads the channel more than in *standalone* mode. The suggested event interrupts cluster creation if a leader does not receive sufficient valid join VAMs in the span of *timeLeaderWaiting* (default: 2 s).

As a solution to **RG 5**, the use of an optional buffer distance

$$d_{Bf} = v_{VRU} \cdot t_{VAM}, \tag{3.1}$$

can be added to the edge of all bounding boxes, increasing the covered area as shown in FIGURE 15. It uses the highest VRU velocity within the cluster members (v_{VRU}) and the VAM assembly time (t_{VAM}) as parameters (LOBO; BARBOSA DA SILVA; FACCHI, 2024). This functionality prevents VRUs from being at the exact border of the geometry, leading to them being wrongly determined inside/outside the bounding box due to latency, time delays, or position errors.

FIGURE 15 – Bounding box buffering



Furthermore, this present Thesis identifies and solves an additional gap by introducing another breakup condition: when a leader receives leave notifications from its cluster members, it recalculates the bounding box and reduces the cardinality. However, if that cluster contains less than three VRUs, it is impossible to form a rectangle or polygon since the VRUs' positions would amount to a single line segment. This study suggests that a cluster should dissolve whenever a leader identifies that the cardinality is below *numCreateCluster* for the same duration as *timeLeaderWaiting*. This parameter provides some time for join requests to be received and processed by the leader while also matching the suggested timeout for the initial formation of the cluster.

3.2 RELATED WORKS

Regarding the related works to this study, it has been observed that not much is currently available on the topic of VRU clustering, especially when related to the bounding boxes. subsection 3.2.1 presents current research involving VAMs, highlighting that the topics covered differ from the focus of the present report. subsection 3.2.2 lists other fields of research that also employ clustering, describing why their approaches are not suitable for the VBS.

3.2.1 State of the Art for VRU Awareness Messages

Current research on VAMs often focuses on the VRU in a *standalone* capacity. For instance, Lobo, Festag, and Facchi (2023) discuss the enhanced VRU detection time provided by the VBS and explore the advantages of using this service along with ETSI's other solutions, such as the Collective Perception Message.

Through field testing, Lusvarghi et al. (2023) suggests that the rules for triggering VAM generation should differ depending on the profile of the VRUs. Zoghlami, Kacimi, and Dhaou (2022) advocate for a context-aware message transmission scheme based on position and kinematics, leading to adaptive VAM generation.

Concerning VRU Clustering, Rupp and Wischhof (2023) show that this VBS feature reduces the number of VAMs sent in traffic scenarios, particularly for higher cardinalities. However, it also leads to an increase in position error when compared to individual VAMs. Finally, Rupp and Wischhof recommend improving cluster effectiveness by either increasing *maxClusterVelocityDifference* to a minimum of 25% or adjusting it to consider velocity averages rather than instantaneous values.

Also analyzing clustering performance on the VBS, Lobo, Barbosa da Silva, and Facchi (2024) compare simulation results on a scenario with and without clustering enabled. The obtained data shows that clustering reduces channel occupation, thus making communication more reliable by minimizing the message latency and Packet Error Rate. This behavior is related to the DCC function of the service operating closer to its threshold when only individual VAMs are present, leading to the formation of queues and message drops.

3.2.2 Clustering in Other Fields

To the best of the author's knowledge, at the moment, no other research related to the geometrical shaping of the VRU clusters has been published. This topic is particularly challenging since the ETSI 103 300 standard (ETSI, 2021c) does not offer strategies to produce the cluster bounding box shapes. Furthermore, through the literature review of the state of the art on wireless network technologies, the production of these geometries appears to be an unsolved issue to the VBS since most clustering functionalities do not need to worry about the shape that their list of members produces.

For example, *Wireless Sensor Networks* (WSN) involve resource-constrained sensing devices that transmit information at a high frequency. In this field, the main goal of clustering is to reduce channel consumption while also being concerned about power efficiency and data redundancy mitigation (ZHU et al., 2021). Clustering strategies in the WSNs act as topology managers, focusing on grouping nodes based on improving network routing (SHAHRAKI et al., 2020). That means that the clusters in this application are geometrically amorphous since it does not involve the generation of a bounding box based on the position of its members.

In *Radio Frequency Identification* (RFID), electronic tags are attached to objects for tracking purposes in an indoor environment. These tags produce a recognizable response to a signal sent by an RFID reader. Clustering these tags based on their placement can improve the effectiveness of their precise localization whenever the density of objects is high (GOMES et al., 2022). Even though the clustering in RFID focuses on grouping nearby devices, the application aims to define a hierarchy for clustering static items, not worrying about kinematics or the shape formed by the set of objects.

In image processing, bounding boxes are used to contour the result of a classification algorithm (LAKSHMANAN; GÖRNER; GILLARD, 2021). They are, however, limited to forming only axis-aligned rectangles.

The *Bounding Volumes* (BV) concept presents a similar goal to the cluster bounding boxes within the VBS. The formation of BVs uses a variety of computational geometry strategies to represent one or more complex geometries through simpler shapes (e.g., circles, rectangles, and polygons), both in 2D and 3D. This approach is used in collision detection algorithms to produce objects that are easier to process, which is ideal for ray-tracing and hitbox detection in physics simulators, computer animations, and video games (ERICSON, 2004).

Research in *Light Detection and Ranging* (LiDAR) also benefits from BV strategies, using them to represent a perceived object in a 3D space. In autonomous driving, some applications of LiDARs are object detection, assessment of object orientation, and collision detection (LIU; LIU; ZHANG, 2021; V; PANKAJ, 2022; WANG et al., 2020).

4 METHODOLOGY

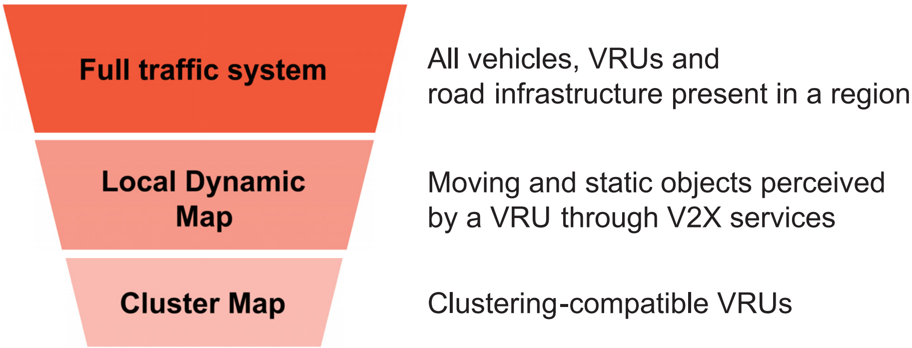
To evaluate the viability of clustering on the VBS and the particular effect of each shape, previous implementations of the standard on *Artery* were extended. In Lobo, Festag, and Facchi (2023), the VAM containers and the message transmission triggers were developed for *standalone* VRUs, providing the basis for the VBS in the framework (LOBO; FESTAG; FACCHI, 2023). Continuing the VBS development, Lobo, Barbosa da Silva, and Facchi (2024) implements the cluster management functionality, enabling VRU clustering with polygonal shapes (LOBO; BARBOSA DA SILVA; FACCHI, 2024). Each device with VBS has a cluster management instance responsible for storing the VRU's role, parsing the Cluster Information and Operations containers, managing the bounding box construction, and monitoring the LDM for conditions to trigger cluster events (creation, join, leave, and breakup).

This work enhances Lobo, Barbosa da Silva, and Facchi (2024) by implementing all three cluster bounding box shapes described in ETSI's standard (ETSI, 2021c), determining strategies based on computational geometry to form these structures. The introduction of a new data structure, the *Cluster Map*, supports the current VBS clustering functionality. This object stores data from cluster-compatible VRUs, providing a *standalone* VRU with means to determine if it should create a cluster while also providing the *cluster leader* with data regarding its members. Additionally, an *Artery* simulation scenario enables the evaluation of the effects of clustering using the different shape types, comparing the benefits and drawbacks of each on a VBS.

4.1 CLUSTER MAP CONCEPT

This study introduces the *Cluster Map* (CM) concept as a tool to support the cluster management functionality of the VBS. A CM uses the data objects of the LDM to store perceived VRUs that are cluster-compatible, containing information such as position, kinematics, and station ID. The purpose of the CM, as illustrated in Figure 16, is to generate a specialized data structure that is simpler to iterate as part of the VBS functional cycle, supporting the creation and management of clusters. A CM is instantiated in every VRU as it assumes a *standalone* or *cluster leader* role.

FIGURE 16 – Cluster Map as a specialized LDM.



Initially, the CM stores a list of cluster-compatible VRUs perceived in the LDM, with the ego *standalone* VRU using this structure to evaluate if it could create a new cluster and become a leader. In these situations, the CM is used to determine the expected cardinality and calculate the radius of the first iteration of the bounding box. Cardinality is determined by summing the number of VRUs on the CM and adding one to the total to account for the leader. After successful cluster creation, the initial CM is purged and is only updated by the leader through valid join and leave VAMs from compatible nodes. This change ensures proper cluster management by the leader who must validate the received operations and keep track of the members' positions, enabling it to construct an appropriate bounding box.

4.2 BOUNDING BOX GENERATION

The development of VRU Clustering on *Artery* presented in this work addresses the absence of specified Bounding Box formation strategies in the standard (ETSI, 2021c) by determining and implementing its methodologies. The adopted approaches stem from the literature review of bounding boxes in the realm of computational geometry, being the first to offer strategies for clustering on a VBS.

The VBS incorporates the generation of bounding boxes through an instance of the cluster management module in each VRU. A device shall only have access to this functionality if either they are a *standalone* VRU attempting to generate a new cluster or if they are already a leader. In both these cases, a *Cluster Map* is used to produce a point cloud that indicates the position of a set of VRUs. The CM is the primary data source for the proposed bounding box generation strategies. For simplicity, this study assumes that all devices in a test scenario use the same default shape type and that no type change is possible except the one that occurs at the transition between the cluster formation and the maintenance phases. The name "ego VRU" refers to the VRU that is currently in focus on the explanation of a VBS functionality. This VRU possesses information obtained through sensors or received messages that enables it to make an informed decision on the present V2X service. For example during the creation of a new cluster, the VRU responsible for identifying potential members and building the initial cluster shape is the ego.

4.2.1 Initial bounding box

For the initial bounding box, ETSI 103 300 determines that the first iteration of a cluster must contain only the leader, with a cardinality of one (*minClusterSize*), and a bounded area covering only the VRU creating the cluster (ETSI, 2021c). At this moment, the *standalone* ego VRU creates an empty *initial Cluster Map* (iCM). Next, it iterates the LDM, searching for all the perceived VRUs and comparing if they are compatible with the ego. The first parameter evaluated for this decision is the Euclidean distance

$$d = \sqrt{(x_E - x_P)^2 + (y_E - y_P)^2}\ , \tag{4.1}$$

calculated between each perceived VRU (P) and the ego VRU (E). Since this study does not consider elevation differences, a simple implementation with the device's positions in 2D xy coordinates is sufficient. The nodes E and P are compatible if the distance is less than 5 m. According to ETSI's specification, the recommended maximum distance is between 3 to 5 meters (ETSI, 2021c). Through testing of this functionality in Artery simulation scenarios, it was observed that the maximum advised value facilitated cluster formation, since nodes are more easily considered compatible through this choice.

The second parameter, speed difference, is given by

$$v_{diff} = \left| \frac{v_E - v_P}{v_E} \right| \cdot 100\%\ , \tag{4.2}$$

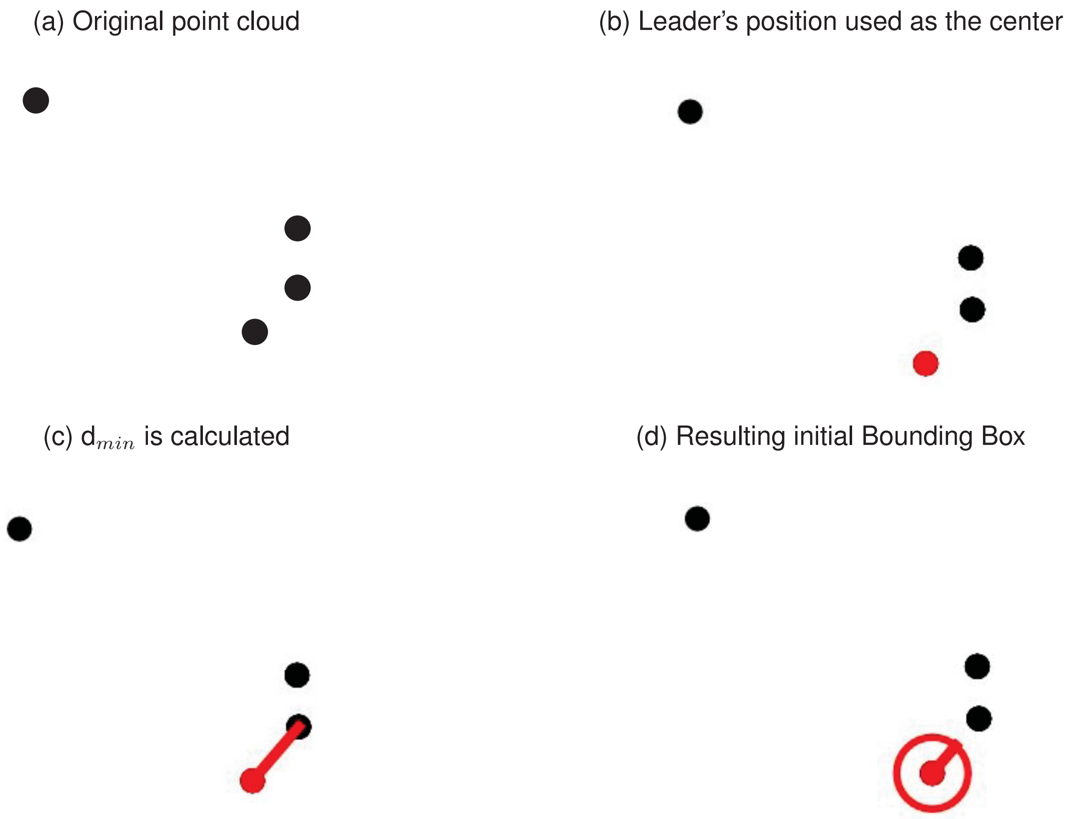
the ratio of the absolute values for speed v of both E and P nodes, with a maximum suitable difference of 5%, as per required in the standard. If P has both d and v_{diff} within the acceptable ranges, the VRU's data object is appended to the iCM. During this assessment, the ego VRU must store the overall smallest valid distance (d$_{min}$) among all the d's calculated from the iCM, using it later to generate the initial cluster bounding box. After evaluating all the VRUs in the LDM, the ego VRU verifies the size of the resulting iCM. If it contains at least two VRUs, the potential cluster reaches a cardinality of *numCreateCluster*, and a cluster is possible. Next, the ego VRU switches roles and becomes a leader following the "first come, first cluster" criteria, starting the cluster-creation process. Algorithm 1 summarizes this decision workflow.

Algorithm 1: Decision to create or not a cluster.

```
Data: Ego VRU's status and LDM
Result: Cluster Creation Decision
initialization;
while VRU's role is standalone do
    update LDM;
    check LDM for compatible VRUs;
    generate iCM with compatible VRUs;
    if iCM size ≥ 2 then
        set VRU's role as cluster leader;
        produce initial cluster bounding box;
        generate random cluster ID;
        include CIC in VAM;
        send initial cluster VAM;
    else
        send standalone VAM;
    end
end
```

For the first cluster VAM, the initial shape shall always be a circle since this is the shape that needs the least amount of data elements, being described by a node center point (*C.a*) and a radius (*C.b*). The first is a tuple of x and y offset distances in centimeters between the *cluster leader* and the actual center of the proposed bounding box. Since the initial geometry must contain only the leader, the bounding box is centered on its position, resulting in a node center point of (0,0). The radius is given in decimeters and calculated by dividing d_{min} by half, placing the edge of the shape at a balanced distance between the leader and the closest compatible VRU. An example of this process for a point cloud containing four VRUS is illustrated by Figure 17.

FIGURE 17 – Initial Bounding Box formation



(a) Original point cloud

(b) Leader's position used as the center

(c) d_{min} is calculated

(d) Resulting initial Bounding Box

The leader then adds *C.a* and *C.b* to the CIC, starting to send cluster VAMs shortly after. The broadcast of the initial shape continues until the cluster creation is concluded by having at least *numCreateCluster* members or is interrupted due to an elapsed time above the *timeLeaderWaiting* threshold.

After generating the initial cluster VAM, the cluster manager discards the iCM, and an empty *maintenance Cluster Map* (mCM) is created to store the data objects of the cluster participant VRUs. The main difference between these CMs is that the iCM contains a list of all the VRUs that are cluster-compatible within the LDM, acting as a list of potential members, while the mCM possesses only valid member VRUs that have actively sent join VAMs to the leader.

When the mCM has sufficient members (*numCreateCluster*) added through join VAMs, the cluster management enters a maintenance mode and stops sending the initial cluster VAM. At this point, the leader must send VAMs every generation time (*T_GenVam*) with a CIC containing a bounding box and cardinality representing all its participants. These participants are added or removed based on received VAMs sent to the leader from other VRUs, with the criteria for compatibility once again based on the Euclidian distance and velocity difference given in 4.1 and 4.2, respectively. Whenever the leader receives a valid join VAM and the cluster has not reached its maximum cardinality, the VRU is added to the mCM and the bounding box shape is rebuilt to contain the new member. Conversely, on receiving a valid leave VAM, the VRU is removed from the mCM and the bounding box is updated to cover the new state of the Cluster Map. The manager keeps repeating this maintenance cycle until a breakup occurs due to insufficient cardinality or other reasons mentioned in (ETSI, 2021c). The routine executed by the leader is described in Algorithm 2.

Algorithm 2: Cluster maintenance by the leader.

```
Data: mCM and received VAMs
Result: Updated cluster
initialization;
while VRU Role is leader do
    parse received VAMs;
    if VAM's COC contains cluster ID then
        if Operation is Join and Cluster is not full then
            add new VRU to mCM;
            update bounding box;
        else if Operation is Leave then
            remove VRU from mCM;
            if mCM size < 3 for over 2 s then
                trigger breakup Operation;
                include COC to VAM;
                set VRU's role as standalone;
            else
                update bounding box;
            end
        end
    end
    include CIC to VAM;
    send cluster VAM;
end
```

The *update bounding box* segments of Algorithm 2 vary depending on the chosen shape type. Sections 4.3, 4.4, and 4.5 describe the strategies employed to generate each geometry type. All of these approaches initiate from the mCM, which serves as the initial reference point, supplying the positions of all cluster members. These methods aim to generate the essential bounding box data elements for the CIC.

The use of the *Axis Aligned Bounding Box* (AABB) supports the formation of the circle and rectangular shapes. This strategy, present in Algorithm 3, has time complexity $\mathcal{O}(n)$ and uses the cartesian coordinates of the VRUs in the mCM to determine the lowest-leftmost and the highest-rightmost VRUs, calling these points *min* and *max*, respectively. These two points describe the minimum possible non-rotated rectangular envelope that covers all the mCM nodes (ERICSON, 2004).

Algorithm 3: Axis Aligned Bounding Box formation.

Data: mCM
Result: min and max from AABB
initialization;
for *VRU object in mCM* **do**
    get VRU position (x_p, y_p);
    if $x_p < x_{min}$ **then**
        $x_{min} \leftarrow x_p$;
    else if $x_p > x_{max}$ **then**
        $x_{max} \leftarrow x_p$;
    else if $y_p < y_{min}$ **then**
        $y_{min} \leftarrow y_p$;
    else if $y_p > y_{max}$ **then**
        $y_{max} \leftarrow y_p$;
    end
end
$min \leftarrow (x_{min}, y_{min})$;
$max \leftarrow (x_{max}, y_{max})$;
return min and max;

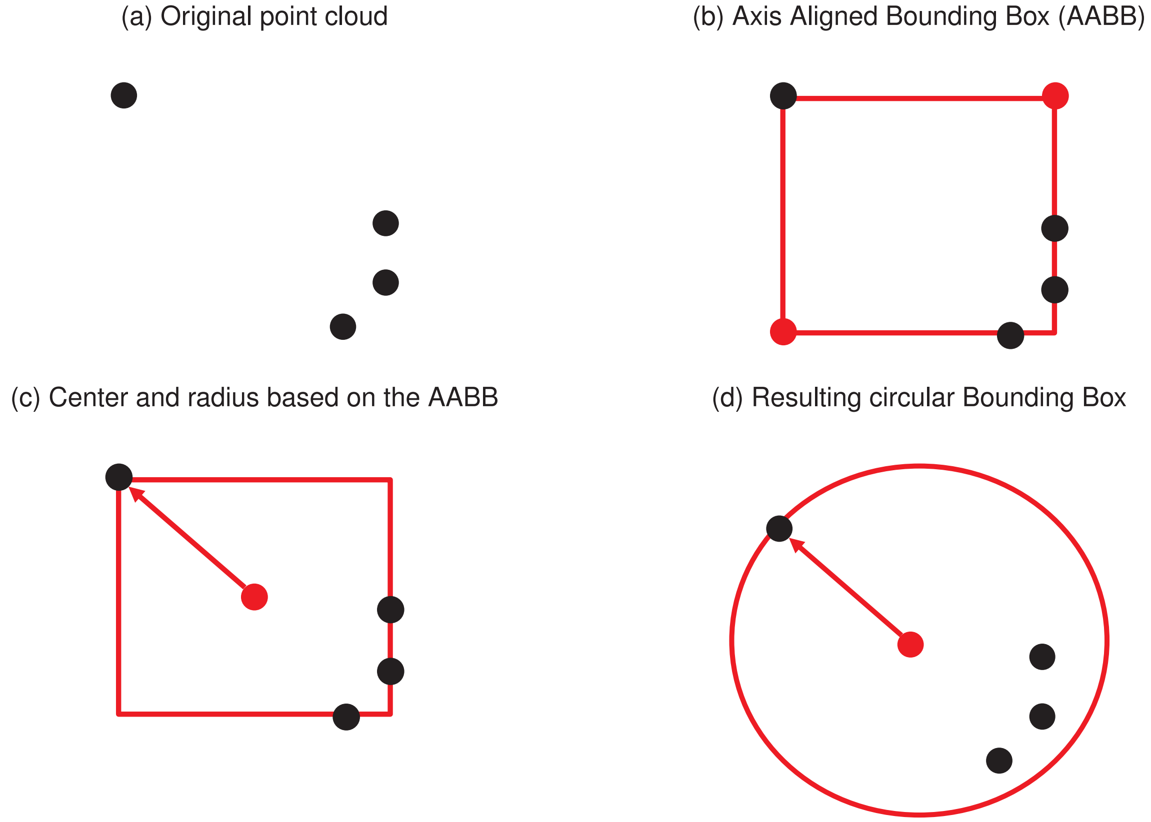
4.3 CIRCULAR BOUNDING BOX

A circular geometry on the VBS needs two data elements: the node center point *C.a* and the radius *C.b*. The most straightforward strategy to determine the shape's center would be to take the average of the point cloud coordinates. However, this approach may result in a radius twice as large as necessary when the points are not uniformly distributed. Thus, another method to determine the center of a point cloud is to first create an AABB around it and consider the center of the resulting geometry as the cloud center (ERICSON, 2004). For the circle radius, it is only necessary to determine the distance d_{max} between the furthest VRU of the mCM and the AABB center. The creation of this box follows Algorithm 4 and is illustrated by FIGURE 18. The time complexity of this algorithm is also $\mathcal{O}(n)$.

Algorithm 4: Circular Bounding Box formation.

```
Data: mCM
Result: Circle CIC data elements
initialization;
calculate AABB from mCM;
obtain min and max from AABB;
get center C from min and max;
// Get largest distance to center
for each VRU object in mCM do
    get current VRU position P;
    get distance d_cp between C and P;
    if d_cp > d_max then
        d_max ← d_cp;
    end
end
obtain leader position P_leader;
C.a ← offset between P_leader and C;
C.b ← d_max;
return C.a and C.b;
```

FIGURE 18 – Circular Bounding Box formation using Algorithm 4



4.4 RECTANGULAR BOUNDING BOX

A rectangular bounding box is described by a node center point *R.a*, half-length *R.b*, half-width *R.c*, and orientation *R.d*. The strategy adopted to build an orientation-dependent rectangle is to iterate the point cloud from the mCM at different rotations and generate an AABB each time, calculating the resulting area on each step and selecting the smallest (AAABB$_{min}$). To obtain the half-length and half-width, the (*min*, *max*) pair from AAABB$_{min}$ are used by comparing the *x* and *y* coordinates separately. The process follows Algorithm 5, an example is shown in FIGURE 19, and has time complexity $\mathcal{O}(n)$.

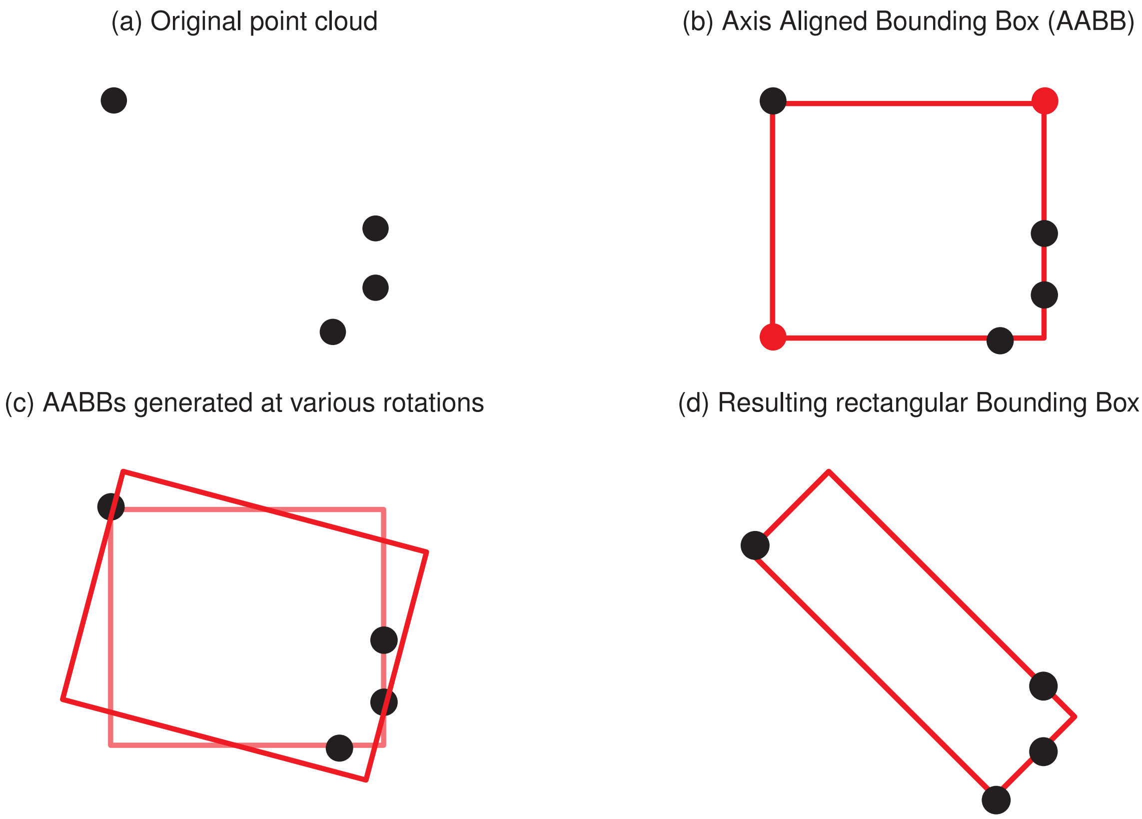
Algorithm 5: Rectangular Bounding Box formation.

Data: mCM

Result: Rectangle CIC data elements

initialization;
create a point cloud from mCM;
`// Get smallest rotated cloud area`
for θ *within (0, 2π)* **do**
    rotate cloud θ counterclockwise;
    calculate AABB from rotated cloud;
    calculate area A from AABB;
    if $A < A_{min}$ **then**
        $A_{min} \leftarrow A$;
        AABB$_{min}$ $\leftarrow$ current AABB;
        $\theta_{min} \leftarrow$ current θ;
    end
    increment 0.1 to θ;
end
get min and max from AABB$_{min}$;
$R.a \leftarrow$ center from min and max;
$R.b \leftarrow (max.x - min.x)$ * 0.5;
$R.c \leftarrow (max.y - min.y)$ * 0.5;
$R.d \leftarrow \theta_{min}$;
return $R.a$, $R.b$, $R.c$, and $R.d$;

FIGURE 19 – Rectangular Bounding Box formation using Algorithm 5



4.5 POLYGON BOUNDING BOX

The VBS defines a Polygon in the CIC through a list of offsets from one vertice to the next. In this implementation, the VRUs positions from the mCM are all candidates to form the geometry's vertices, resulting in a max number of 20 (*maxClusterSize*) offsets. However, implementing a strategy to select which points to use is fundamental, as a simple ordered list with all the candidates could lead to holes and spikes in the bounding box.

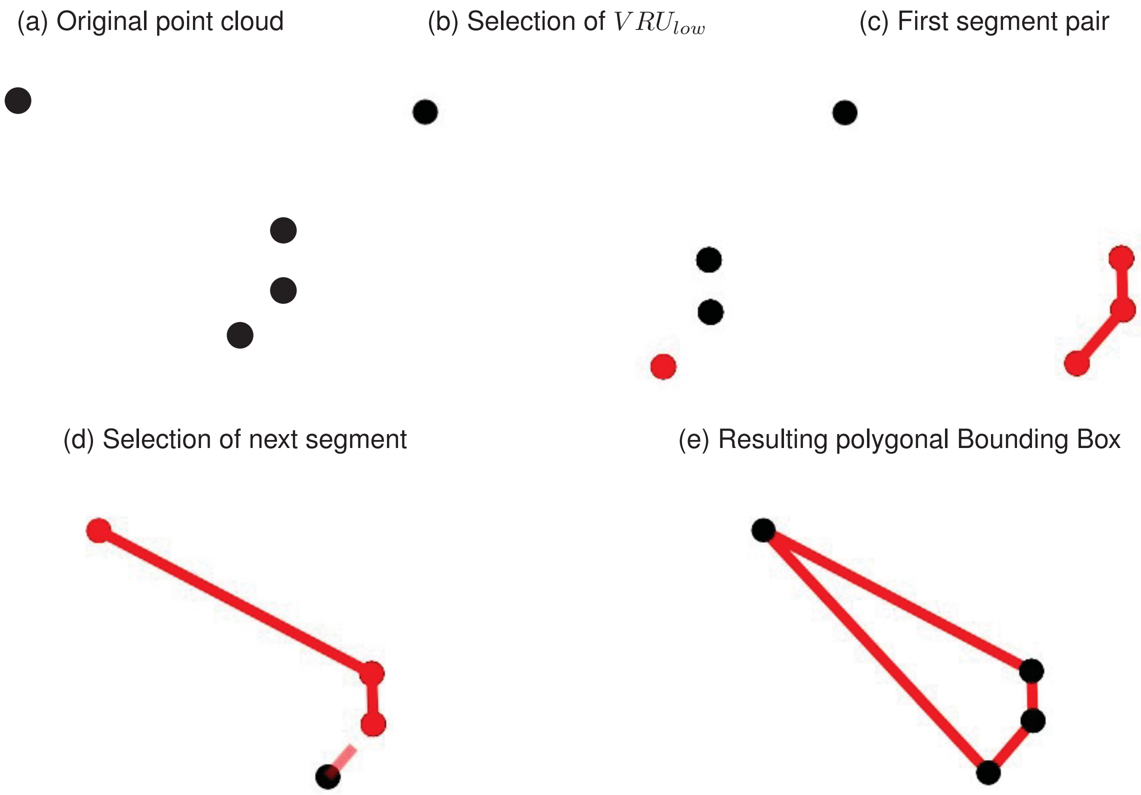
To avoid holes and spikes in a polygon, it must be convex, meaning that all the interior angles must be less than 180 degrees. To ensure convex polygon generation, a Convex Hull algorithm that implements the Graham Scan is used (SHAMOS, 1978). This approach uses the point cloud from the mCM, first searching for VRU_{low}, the single lowest vertical position, or the leftmost if multiple points share the y-lowest position. It then produces a list by sorting the cloud based on the polar coordinates related to this reference point. Next, it iterates the resulting list starting at VRU_{low}, selecting triplets of consecutive points, with the central point as a vertex candidate.

At every loop iteration, it evaluates through a cross-product if the two segments formed between the candidate and the neighbor points generate a left (counterclockwise) turn, meaning it has an interior angle less than 180 degrees (SHAMOS, 1978). If true, the algorithm moves one position down the list and checks the next triplet. If not, the candidate generates a right turn or is collinear to its neighbors and should thus be removed from the list. Next, the selection is backtracked in one position, using a previously approved vertex and checking if, with the new right neighbor point, it still produces a left turn (SHAMOS, 1978). The scan, summarized in Algorithm 6 and illustrated by FIGURE 20, ends when the list of potential vertices is exhausted, containing only left turns and with the last segment reaching the first point. The lowest point selection is of time complexity $\mathcal{O}(n)$, while the scan is $\mathcal{O}(n \log n)$.

Algorithm 6: Polygonal Bounding Box formation.

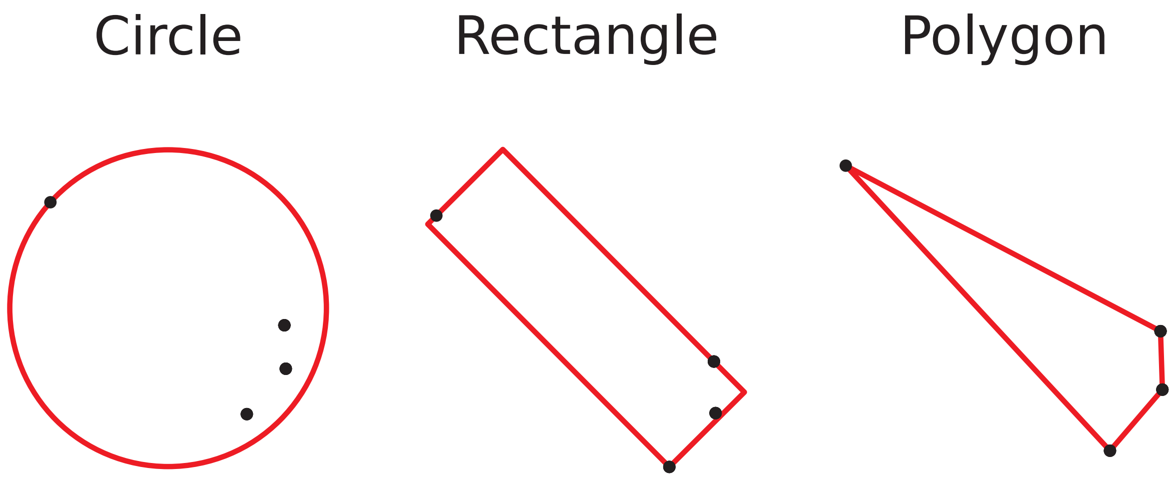
Data: mCM
Result: Polygon CIC data elements
initialization;
get point list from mCM;
iterate list and get VRU_{low};
`// Produce the convex hull`
sort list by polar coordinates to VRU_{low};
list starts and ends at VRU_{low};
while *right of vertex not* VRU_{low} **do**
    get vertex candidate C;
    get left L and right R adjacent points;
    $\overline{CL} \leftarrow (x_L - x_C,\ y_L - y_C)$;
    $\overline{CR} \leftarrow (x_R - x_C,\ y_R - y_C)$;
    $P \leftarrow \overline{CL} \times \overline{CR}$;
    if $P >$ *0* **then**
        keep candidate in the list;
        get next candidate;
    else
        remove candidate from the list;
        move to previous vertex;
    end
end
get the list of $n \in N$ vertices;
calculate the offset of consecutive vertices;
each offset is assigned to a $P.a.n$;
return all $P.a.n$ offsets;

FIGURE 20 – Polygonal Bounding Box formation using Algorithm 6



A comparison of the resulting bounding boxes obtained by using the strategies from Algorithms 4, 5, and 6 for the same set of four VRUS is shown in FIGURE 21. In an initial assessment, one can observe that the three methods can construct shapes that optimally cover all the points while occupying the smallest possible area. However, it is noticeable that a circular shape yields a larger bounding box for the same point cloud, whereas the rectangle and polygon produce a more well-fitted perimeter.

FIGURE 21 – Bounding boxes for the same set of points.



4.6 METRICS DEFINITION

Since ETSI's 103 300 does not discuss in which situations each bounding box shape type should be used, it is necessary to determine parameters to compare some of the effects of the different bounding boxes on the VBS. The goal is to verify if selecting a particular shape offers significant pros and cons, leading to a geometry being better suited for an use case. Due to the uniqueness of VRU clustering in the state of the art for wireless communication, and the fact that the VBS is a V2X service still in ongoing development, the comparison metrics proposed focus on this study aim to introduce a discussion on how the selection of a bounding box shape can affect the service. More complex parameters, such as the impact of a shape on the likelihood of a collision, even though fundamental for the real-world implementation of the VBS, are out of the scope of the present work.

Since cluster VAMs do not disclose the exact position of their members within a bounding box, the complete geometries must be treated by other road users as solid objects. And thus, touching any part of the shape should be intrepreted as a collision and must be avoided at all costs in order to ensure VRU safety. This means that other road users that encounter a cluster bounding box should brake or move around them to avoid a crash. As such, it is crucial to refrain from generating a shape much bigger than what is needed to cover all participants, as large boxes might block traffic on segments of sidewalks or streets. Another potential issue of excessive size is the overlap of nearby cluster geometries, which could confuse a VRU about which cluster to join and a vehicle about how to avoid the clusters properly. To compare how good each shape type is at representing its members without occupying an excessive area, the cluster density

$$D = \frac{cluster\ cardinality}{bounding\ box\ area} \tag{4.3}$$

is introduced. It is calculated from the ratio of the cardinality of a cluster per its area in VRUs/m^2. The higher the density, the better a shape type is at representing a set of VRUs without contributing to the blockage of pathways and risk of bounding box overlap.

Each geometry type implies a different amount of data elements added to the CIC of a VAM. A circular bounding box in the VBS takes two parameters, offering the smallest increase to the CIC. Polygons, on the other hand, lead to the largest potential increment in VAM size since they need between three and twenty offsets to form, depending on cluster cardinality and the number of vertices selected by the Convex Hull. As an intermediate option, the rectangle requires four variables, needing more data than circles but less than most polygons.

Thus, a method to compare the geometry types is by evaluating the increment to the VAM size. Since the clustering functionality as implemented by this work is present only on the Facilities layer, the message increment of only this layer is taken into consideration. This metric is relevant because, in a crowded environment, it is particularly relevant to reduce as much as feasible the message size so that the sent VAMs contribute less to channel congestion. It is expected that the selection of a shape type results in little difference to the Facilities layer increment, since a small variation in data elements is not sufficient to severely impact the message size. However, since the amount of offsets that describe the polygonal bounding boxes do not possess an upper bound, it is relevant to observe if this could not lead to issues on a crowded scenario.

Moreover, evaluating the average number of clusters and the number of operations triggered is relevant to observe their influence on the message size and on the amount of VAMs sent. Both of these are associated with Quality of Service, since both more VAMs being transmitted and larger increments to the VAMs can lead to an increase on the occupation of the wireless channel. The metric *Channel Busy Ratio* (CBR) defined as

$$CBR = \frac{T_{Busy}}{T_{CBR}}, \tag{4.4}$$

where T_{Busy} is the time a single channel is busy with transmissions, and T_{CBR} is the interval standardized by the ITS-G5 (normally 100 ms) to calculate this ratio. The CBR is used in V2X services to determine how occupied a channel is. One of its main uses is in the DCC mechanism, responsible for queueing and dropping messages to reduce the congestion when the CBR reaches the threshold of 0.65 (ETSI, 2018).

4.7 SIMULATION ENVIRONMENT

To test the presented shaping strategies and compare the geometries using the metrics defined in Section 4.6, a simulation stack based on traffic and network simulators was used for this study. The following sections offer a brief overview of the tools and the setup parameters.

4.7.1 Microscopic Traffic Simulation using SUMO

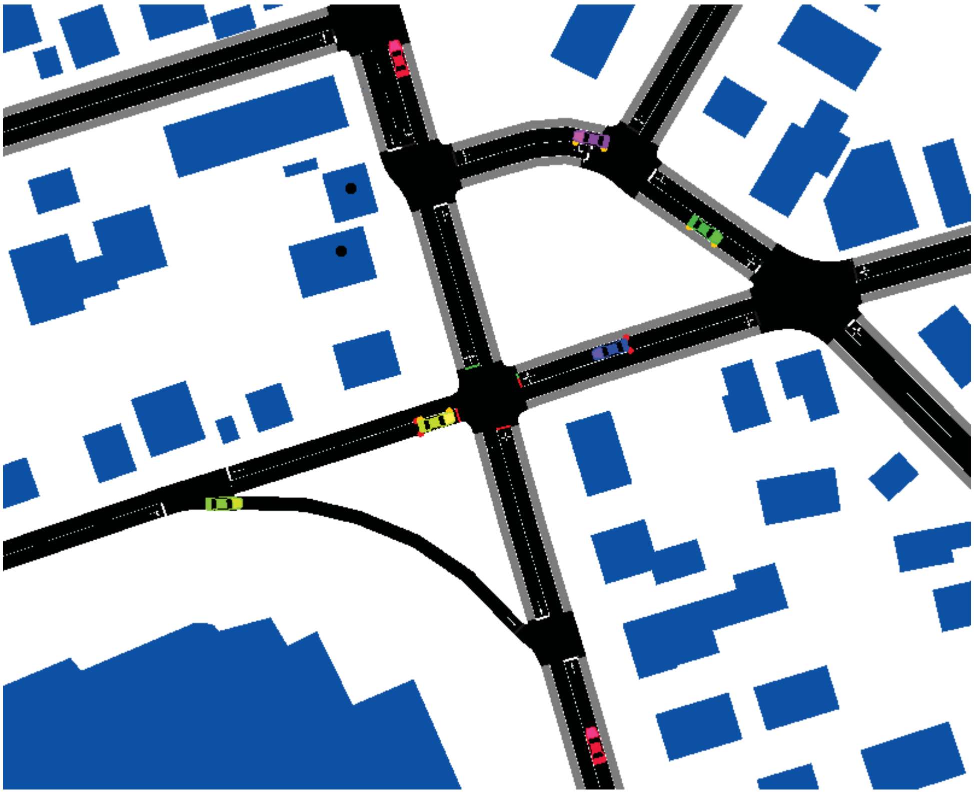
Simulation of Urban MObility (SUMO)[1] is an open-source traffic simulation suite that supports intermodal transport systems. The streets for the simulations are defined through net files, which are Extensible Markup Language (XML) files. Through the *netconvert* tool, the SUMO package offers plenty of options to convert from other

1 https://eclipse.dev/sumo/

file formats to the net XML, enabling the user to import real maps to produce traffic simulations based on real-world cities.

This tool can contain various road user types, such as cars, pedestrians, trains, and buses that interact with each other. Road user dynamics can be customized through programmable traffic lights, setting up parking areas, and even public transport lines. The movement of road users can be modeled through predefined paths or routing algorithms. SUMO files can also contain building geometries, which can be used to improve wireless communication simulations. That is because any physical infrastructure between a transmitter and a receiver can affect performance by blocking direct lines of sight or due to multipath propagation. FIGURE 22 illustrates a SUMO simulation with modeled streets, sidewalks, traffic lights, buildings, and vehicles.

FIGURE 22 – SUMO simulation



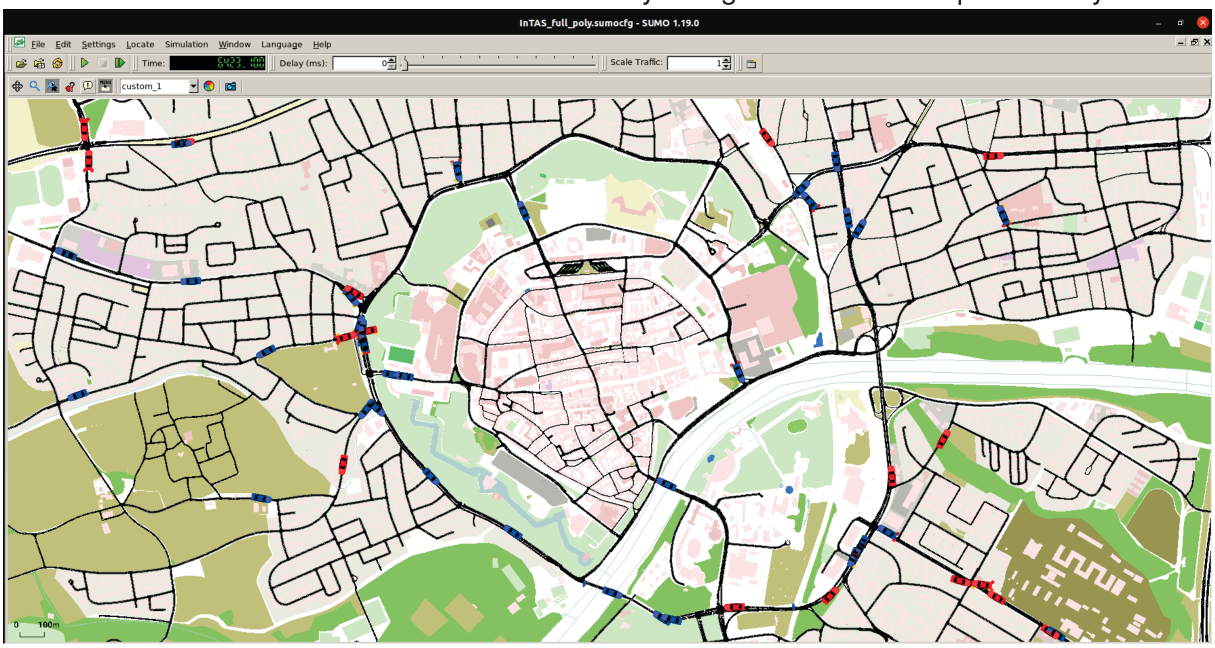
Moreover, SUMO offers a *Traffic Control Interface* (TraCI) that allows developers to control the behavior of the simulation and its actors through a TCP interface, which is then used by the other tools from the simulation stack to synchronize time steps and make the road users respond to data received from V2X messages.

4.7.2 InTAS - The Ingolstadt Traffic Scenario for SUMO

The Ingolstadt Traffic Scenario (InTAS)[2] is a model developed in SUMO that aims to present a realistic traffic scenario for the city of Ingolstadt, Bavaria, Germany, of the year 2020. The scenario includes road topology, building geometries, parking lots, traffic light behavior, public transport routes and timetables, real traffic demand numbers, routing of pedestrians based on demographic data provided by Ingolstadt's city hall, etc. The main goal of the project is to provide a model scenario for *Vehicular Ad-hoc Networks* (VANETs) computational simulations, enabling the use of a scenario based on the real-world and that has been validated using real-traffic data (LOBO; NEUMEIER, et al., 2020).

The InTAS model (FIGURE 23) is then shown as a viable alternative to real-world testing of large-scale transport systems and that can be easily repeated as many times as needed and with adjustable parameters. Based on this model, it is possible to select different moments of the day and sections of the city depending on the goal of the simulation, for example, to evaluate how clustering would behave at a busy crossing during rush hour.

FIGURE 23 – View of the traffic model of the city of Ingolstadt in SUMO provided by InTAS



4.7.3 OMNeT++

The *Objective Modular Network Testbed in C++* (OMNeT++)[3] is also an open-source simulation tool, however, its focus is on network simulations, covering the whole

[2] https://github.com/silaslobo/InTAS
[3] https://omnetpp.org/

communication stack of various wired and wireless standards. It is structured as a set of independent modules that can be connected to form complex systems and is also extensible with additional libraries to support other protocols. Each module's behavior is configured as C++ source files, while the hierarchy and connections are done through a *Network Description* (NED) file.

Simulations can be started through a graphical interface or command line window, and since a lot of the network simulations have stochastic behavior, many executions using different seeds can be set up. This is also a useful resource when it comes to parameter iterations. Moreover, the results of each simulation can be stored in many formats, including in comma-separated values or writing them in a database, making the data processing a process that can be tailored by the developer. Finally, since it is based on discrete-event simulations, each step is executed sequentially and with an associated timestamp reflecting its real-world counterpart. This makes it possible to synchronize with the SUMO traffic simulation to provide a bidirectional flow of data between the two tools.

4.7.4 Vanetza

Vanetza[4] was developed at the Car2X Laboratory from the CARISSMA Research Center, which is part of Technische Hochschule Ingolstadt, as an open-source implementation of the C-ITS standardized by ETSI (RIEBL, 2021). It operates mainly on the Network and Transport layers of the protocol stack, offering ready-to-use: *GeoNetworking* (GN), *Basic Transport Protocol* (BTP), and *Decentralized Congestion Control* (DCC) features. Additionally, the library provides support for *Abstract Syntax Notation One* (ASN.1) messages for the Facilities layer, which can be used to format, generate, and send the VAMs. What is also interesting is that this library can be used not only for the Artery simulations but also to handle the messaging on the communication device installed on the vehicle available at the Car2X laboratory. Using the same manager helps to ensure that results obtained in the computational models can have a good equivalency to the real world.

4.7.5 Artery framework

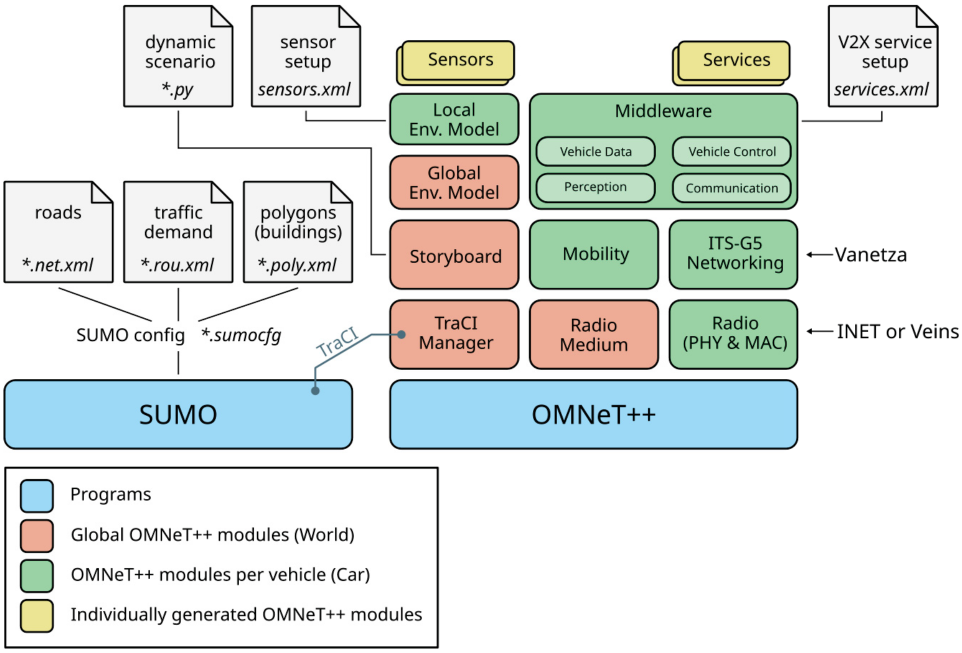
Riebl, Obermaier, and Günther (2019) introduced the Artery framework [5] as an open-source simulation environment that is freely available online, and specifically developed for the European standard of V2X messaging (RIEBL; OBERMAIER; GÜNTHER, 2019). Another important distinction is that this tool can handle in parallel more than one service type in the same setup, thus allowing the creation of more complex scenarios.

4 `https://www.vanetza.org/`
5 `http://artery.v2x-research.eu/`

This for example allows the simulation of ITS-G5 Day 1 deployment, in which vehicles are expected to be capable of handling both Cooperative Awareness and Decentralized Environmental Notification Basic Services. What also is interesting is the fact that the traffic simulation is based on SUMO and the network simulation on OMNeT++, both open-source simulators on their own.

Artery was first introduced in 2015 as an extension of the Veins[6] framework (SOMMER; GERMAN; DRESSLER, 2011), aiming to enable ITS-G5 simulations. Nowadays, however, it is a framework that can use Veins as its radio medium but is not dependent on it. Artery architecture is shown in FIGURE 24.

FIGURE 24 – Artery V2X framework architecture (RIEBL, 2020)



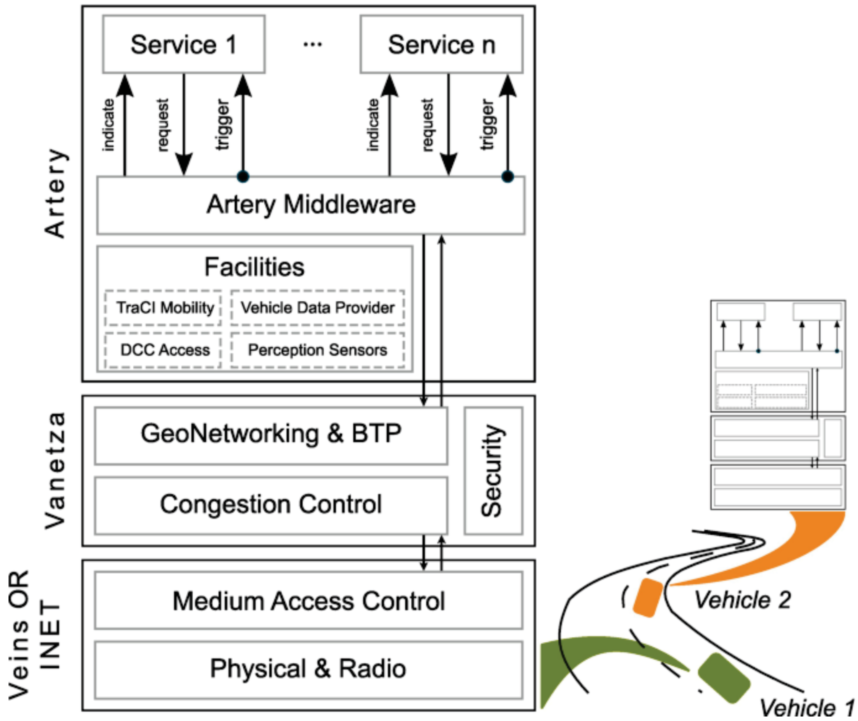
Furthermore, it is shown in FIGURE 24 that the traffic and network simulations are synchronized through TraCI's *Application Programming Interface* (API), which guarantees that both software work on the same time increments of simulation time. Through this API it is also possible to send commands bidirectionally, for example, to control the speed of a vehicle based on a logical verification that occurred on the network simulation. The blocks in red are elements that will be instantiated once in the simulation ("World"). The elements in green are instantiated once for each ITS-compatible vehicle ("Car"). It is important to highlight that the ITS-G5 Networking (BTP and GeoNetworking), responsible for the packet routing and Ad Hoc communication, as well as DCC and the Facilities messages, are all handled by an external API called Vanetza.

Each person and vehicle with V2X capabilities present in an Artery simulation

[6] https://veins.car2x.org

contains an instance of the ITS-G5 communication stack. This means that each simulated ITS-S contains and uses independently every layer from the stack as shown in FIGURE 25, handling the V2X services individually and interacting with the other nearby devices.

FIGURE 25 – Instantiated communication stack for every simulation actor (RIEBL; OBERMAIER; GÜNTHER, 2019)



4.7.6 VRU Basic Service on Artery

At the moment, the official release of Artery does not support VAMs and as a consequence, it is still unable to simulate VRU Basic Services. However, an extension has been developed by the CARISSMA Car2X Laboratory team to enable a basic service based on VAMs. At the moment, this version provides a "Person" object derived from the "Car" actor. It contains its own middleware and is capable of running standalone services involving VRUs using the Basic, High Frequency, and Low Frequency Containers envisioned by ETSI.

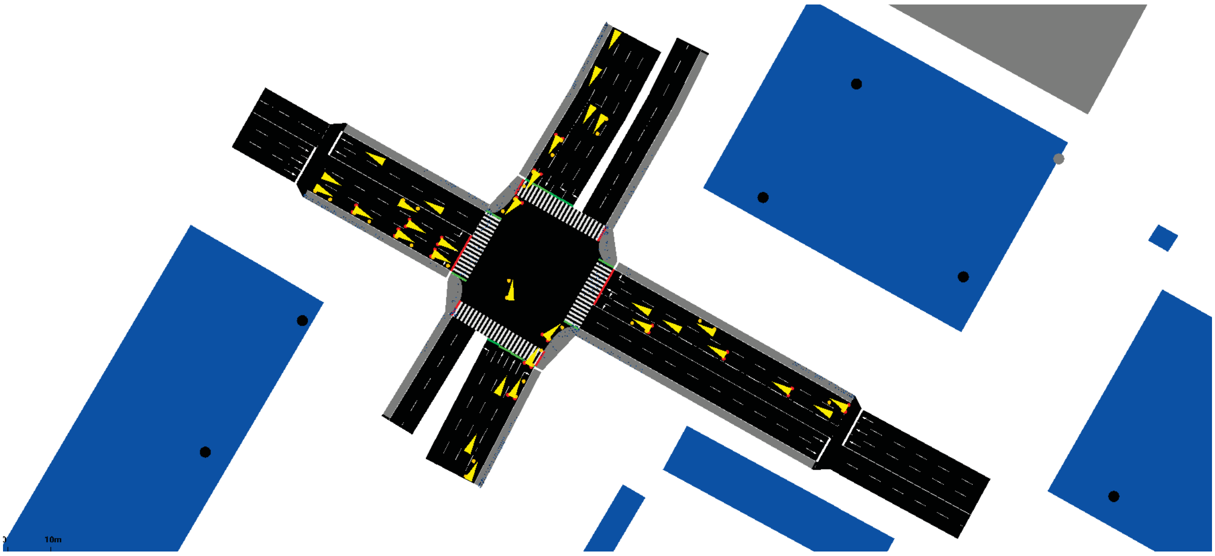
As part of this development team, the author of this thesis proposal has already implemented the Clustering Containers in this repository through the use of the C++ headers and source files available through the Vanetza library. As such, the required steps revolve around the creation of objects that will contain each data field of these containers and also design the functions that assign cluster leaders and manage the cluster states (create, join, leave, and break up). It is important to note that those additions work seamlessly with the already existing standalone VBS and that the optional VAM containers can be enabled/disabled in the simulation configurations.

Regarding the ITS-G5 stack, the messaging protocol *Vanetza* manages the GeoNetworking and DCC features. The V2X simulator *Artery* (RIEBL; GÜNTHER, et al., 2015) handles the Application layer, with the services deployed and managed in each node through middleware modules. This VBS implementation builds upon the work from Lobo, Barbosa da Silva, and Facchi (2024), who first introduced VRU clustering to *Artery*. This present work's main contribution is an extension of the framework, with the introduction of the Cluster Map structure, through which the proposed bounding box generation techniques are possible.

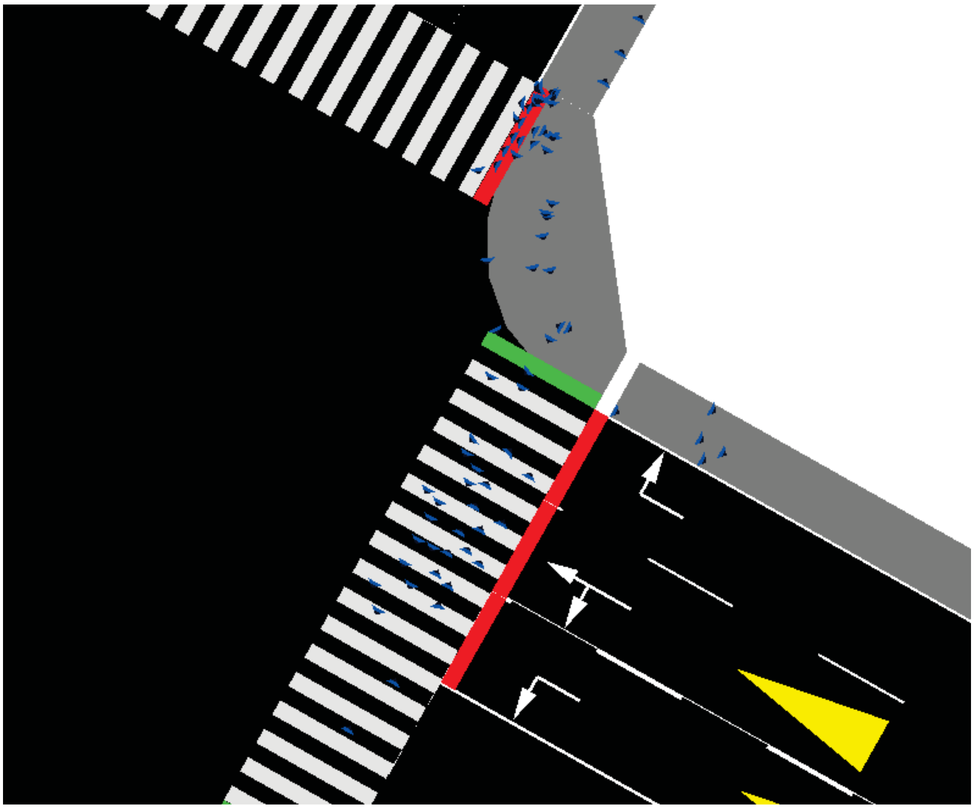
Aiming to create a traffic scenario resembling a real-world application, a crowded pedestrian crossing from Ingolstadt (Germany) provided by InTAS (LOBO; NEUMEIER, et al., 2020) was used as seen on FIGURE 26. With the goal of creating a stress-scenario with an always increasing number of VRUs, a set of 3,042 pedestrians and 70 vehicles were generated at random instants and with random start and end coordinates in the simulated crossing to produce various clustering opportunities and operation triggers. The simulation spans 30 s, processed in steps of 0.10 s each. At its maximum 2,947 pedestrians and 49 vehicles were simultaneously in the crossing. One known issue of this tool is that pedestrians do not have an associated physical size, meaning they are a single point in the simulation and might overlap.

FIGURE 26 – InTAS-based traffic scenario

(a) Overview of the crossing under test



(b) Detail of the crossing, with pedestrians in blue and cars in yellow



The simulation runs for 30 s, with a 10 s warm-up to register the results of the system after the CBR has grown up to the DCC threshold and with clusters already formed. The triggering of events on *OMNeT++* does not occur at the exact instant in every run, having an innate probabilistic behavior. Results originate from the average of simulations using six different seeds.

In every simulation, all present VRUs have the clustering function enabled and will actively look for opportunities to interact with existing clusters or create new ones. As an introduction to the study of the impact of geometry types on the service, this work considers only one shape type per simulation run. So, for example, there is no scenario in which rectangular and polygonal bounding boxes coexist. Table 3 contains the parameters of the simulation stack.

TABLE 3 – Simulation parameters.

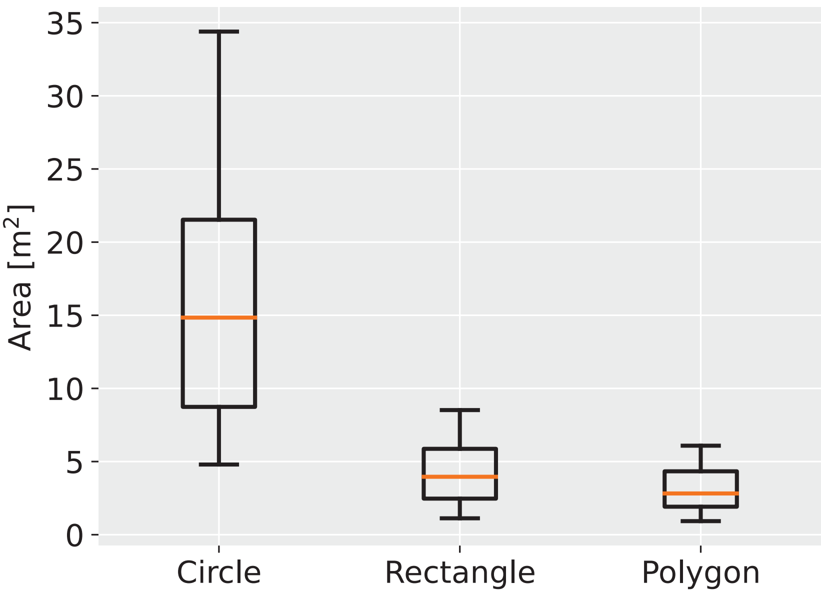
Parameter	**Value**
Total simulated area	5,082 m^2
Total simulation time	30.00 s
Simulation step	0.100 s
Warm-up time	10.00 s
Seeds	[0, 23, 42, 1337, 0815, 4711]
Traffic model	InTAS
Total pedestrians	3,042
Total vehicles	70
Min. cluster size	3 VRUs
Max. Cluster Size	20 VRUs
Cluster Distance	5 m
Speed Difference	5%

5 RESULTS

The VBS containing clustering functionalities and the proposed shaping strategies was deployed on the developed InTAS-based traffic scenario. Simulations using the stack from section 4.7 were executed for the three distinct bounding box types, and also without clustering enabled, using six random seeds for each configuration. This setup resulted in twenty-four simulations, with the following results being the average values obtained from each iteration. The parameters measured are the ones proposed in section 4.6.

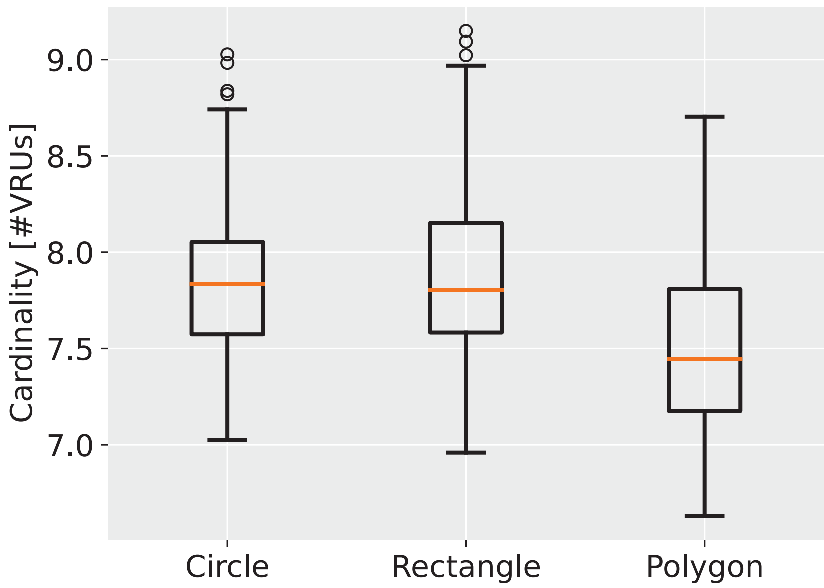
When observing the area occupied by each shape (FIGURE 27), it can be observed that circles not only present the highest median, but also the biggest interquartile range, suggesting a larger variation in the recorded area values. This can be associated with the fact that circles can only adjust their radius to fit new members, leading to quadratic increases to the area. On the other hand, rectangles and polygons have similar minimum values and medians. Rectangles however, have a larger interquartile range and maximum value. The difference between rectangles and polygons can be attributed to the second having more possible vertices, resulting in a more tight-fitting bounding box shape. It is also relevant to mention that the maximum value of the polygon has a similar value to the minimum of the circle, highlighting how polygons are capable of covering member nodes while occupying less area.

FIGURE 27 – Cluster area for each shape



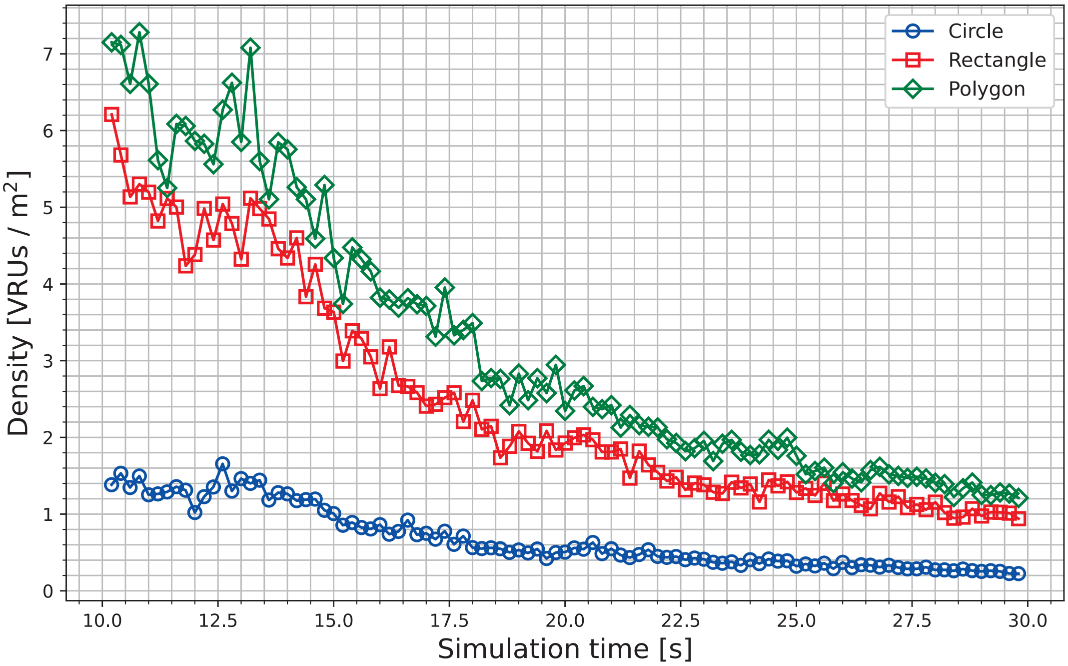
Evaluating the cardinality for each shape type through FIGURE 28, it can be pointed out that circles and rectangles oscillate on a range of 7 to 9 VRUs, while the polygon ranges from roughly 6.5 to 8.5 VRUs. Considering that clusters can have cardinalities ranging from 3 to 20 VRUs, the difference between the shape types indicate that in average the geometries cover a similar amount of VRUs per cluster. However, the upper outliers in the circles and rectangles also show that with these shapes, cluster with higher cardinalities were generated, with more occurrences with circles. This could mean that these shapes, while not as tight-fitting as the polygon, are capable of encompassing more VRUs.

FIGURE 28 – Cluster cardinality for each shape



It can be observed in FIGURE 29 that circular bounding boxes offer the lowest cluster density values. This behavior suggests that this shape type needs to occupy larger areas to cover the participating nodes of a cluster. The polygon presents the highest density overall, reaching more than five times the amount of VRUs per squared meter as the circle at 12.4 s. The rectangle is the second best, presenting similar values at roughly 11.5 and 13.5 s. These results indicate that both the rectangle and the polygon have better fitness to the original point cloud when compared to the circle since there is less area occupied without necessity. These high-density values are also associated with the limitations of pedestrian simulations in SUMO since it is unrealistic for twelve VRUs to occupy the same squared meter. However, the results should still be interpreted as an upper bound of the service, illustrating the higher clustering potential of rectangles and polygons.

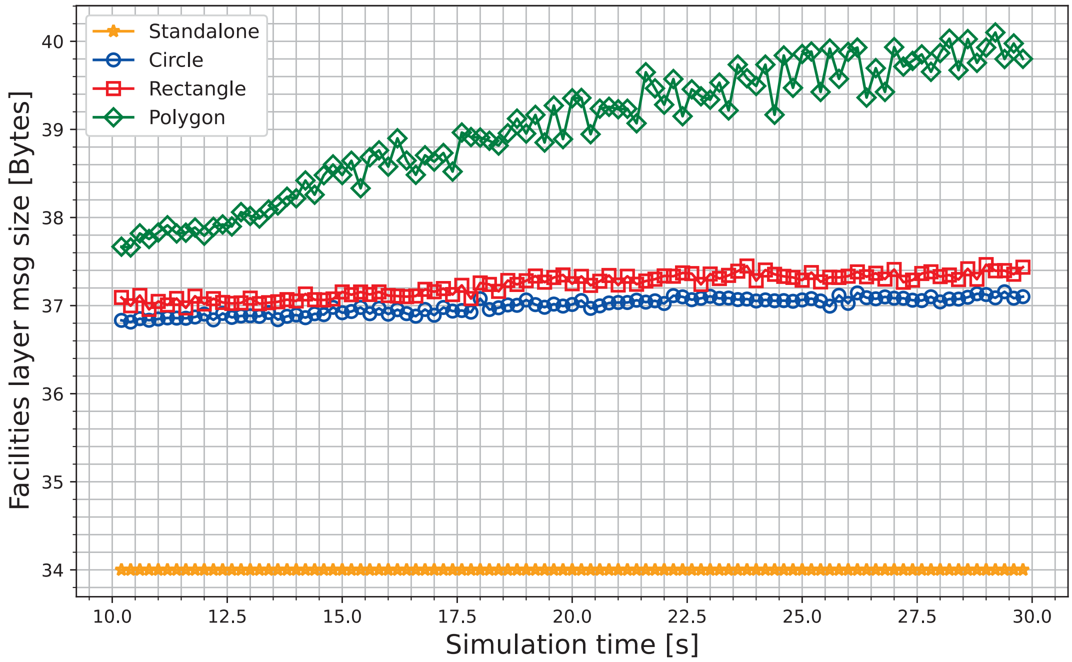
FIGURE 29 – Cluster density for each shape.



Moreover, the rectangles and polygons also present more spikes in the curves, as shown in FIGURE 29. A possible cause is that the shapes are more susceptible to sudden cardinality changes as the VRUs join and leave clusters due to the bounding box being tighter-fitted around the point cloud. After 12.4 s, there is a noticeable decline in the densities for the three shape types, which can be associated with the VRUs drifting apart due to them taking different routes, resulting in a more sparse point cloud.

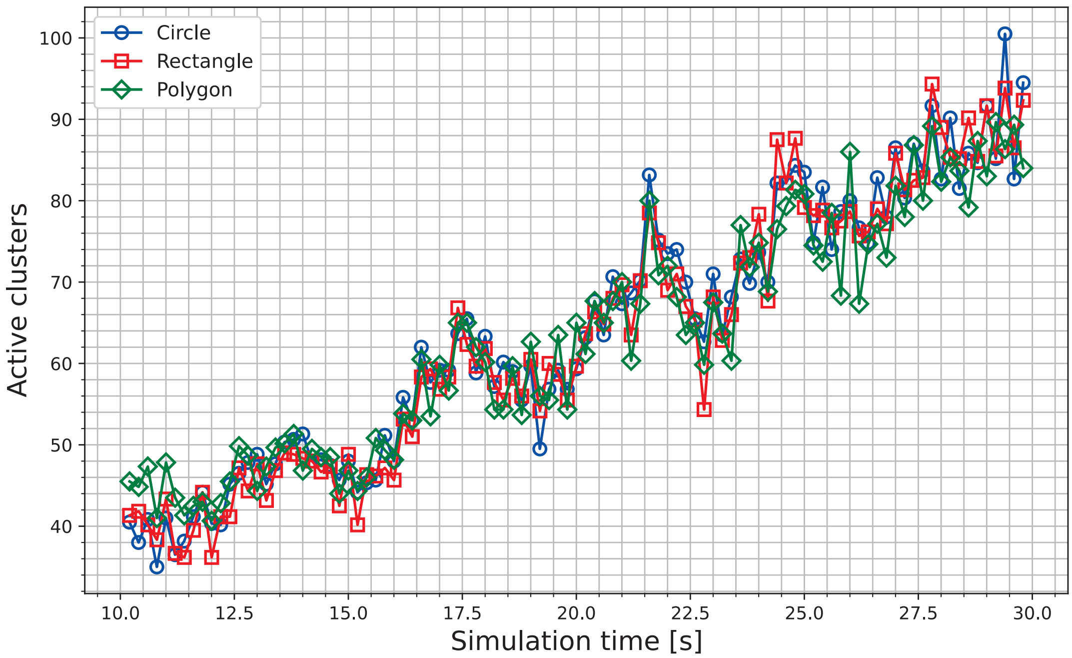
Regarding the average message size, FIGURE 30 shows that the standalone scenario, in which no clustering is involved, offers the lowest increment to the message, at a constant 34 bytes. The second lowest is the circular bounding boxes, offering an average of about 36.85 bytes, confirming the assumption that this type yields the smallest increment to the CIC out of the shape types. Polygons result in the largest VAM sizes among the geometries and the highest difference between minimum and maximum reached values, respectively, 37.35 up to nearly 40 bytes. This gap could be associated with polygons being the only shape type that changes the amount of data elements included in the message depending on each cluster. As the average cluster cardinality grows due to more VRUs entering the crossing, the number of data elements needed also increases. Once again, rectangles are an intermediate option, with an average message size larger than circles but smaller than polygons, standing approximately at 37.05 bytes.

FIGURE 30 – Average facilities layer message size increment for each shape and standalone.



When observing the number of active clusters in a simulation, FIGURE 31 indicates that as time passes, the number of clusters present in the crossing increases similarly for all shape types. This increase helps explain the rising trend in average message size, with varying results based on the increment that each geometry type adds to the CIC.

FIGURE 31 – Average number of active clusters for each shape.



Evaluation of these parameters highlights an interesting aspect of the clustering of VRUs on the VBS. The circular bounding boxes offer the smallest message increment and, therefore, are suited for applications in which channel efficiency is desired, with the drawback that the generated shape has low density. These characteristics mean that when using circular bounding boxes, it is hard to determine the position of the VRUs within the cluster.

Polygons exhibit larger average message sizes, which escalate along with cardinality. However, they offer increased cluster density, indicating a better-fitted resulting geometry. This accuracy improves safety as it is easier for other road users to avoid colliding with the member VRUs described by a polygon. Thus, this shape type is advantageous in less crowded scenarios where message sizes and channel occupation are less critical.

Rectangles offer a compromise between circles and polygons, with the second-best density and message size. New metrics can prove fundamental for this line of study, as further research into the impact of rectangles on the VBS is necessary to determine use cases in which this shape type can be beneficial. Some parameters to evaluate in the future are the position error among the members, average cluster lifetime, and rate of VRUs clustered versus non-clustered.

When the average number of cluster creation, join, leave, breakup, and regular cluster maintenance events are taken into account, it can be noted that the shape choice also affects the clustering dynamics. For instance, TABLE 4 demonstrates that more creation and breakup operations occurred with polygon shapes, then rectangles and lastly circles. This can be related to the cluster area offered by each shape type, since larger areas lead to passive VRUs being covered for longer by a geometry before route differences lead to new cluster creations due to the VRUs stepping out of a bounding box. On the other hand, circles present more maintenance VAMs, while rectangles are second, and polygons have the least amount. This is because circles offer the least amount of creation and breakups, suggesting that cluster of that shape are maintained for longer. This results are also consistent with the amount of active clusters observed in FIGURE 31, as the polygons present slightly less simultaneous clusters as the other shapes.

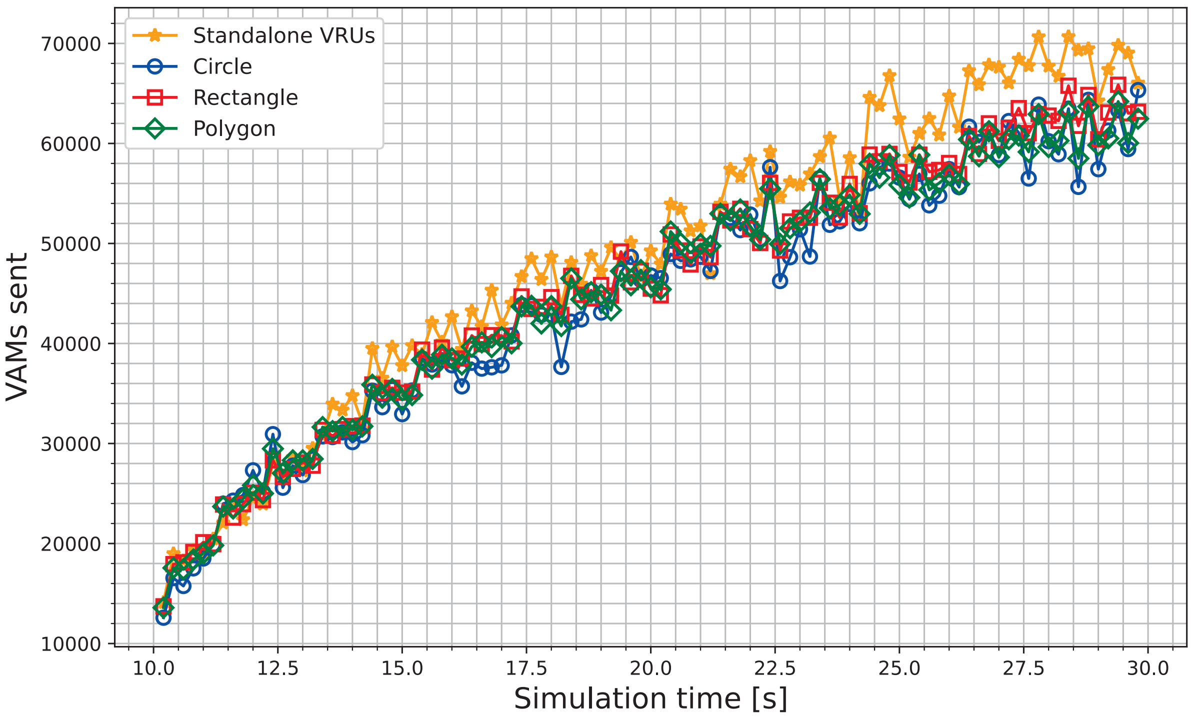
Rectangles exhibit more join and leave occurrences, which fits the assumptions related to the spikes from Figure 29, that the tight-fitting nature of this shape leads to more VRUs entering and leaving a cluster coverage despite the use of the padding distance d_{Bf} (Equation 3.1). This issue is also probably due to the constraints that this shape type presents when covering a point cloud due to its limited number of vertices.

TABLE 4 – Total amount of cluster events per shape type.

Event	Circle	Rectangle	Polygon
Creation	697.83	671.00	731.00
Join	42,115.00	42,423.67	41,885.67
Leave	41,308.67	41,623.50	41,153.00
Breakup	449.00	430.67	466.83
Maintenance	7,349.67	7,252.00	7,182.00

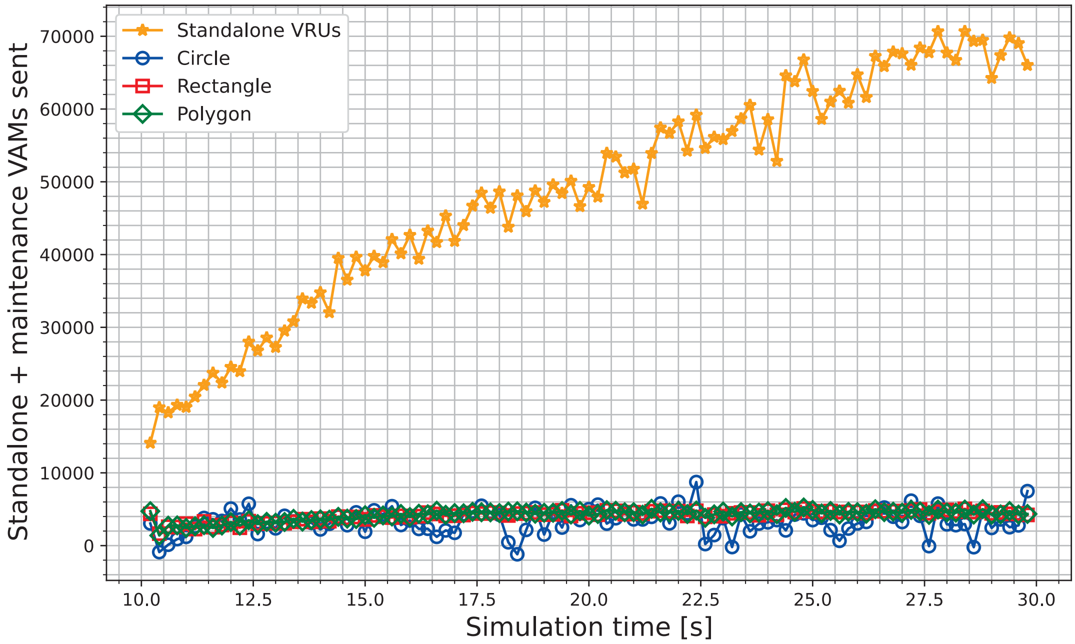
Another important observation from the sum of the transmitted operation VAMs from TABLE 4 is that the amount of events triggered during the recorded 20 seconds of simulation is very large. This issue becomes more evident in the FIGURE 32, in which the amount of VAMs sent in a standalone scenario are compared to the total VAMs sent for each geometry type. As expected, the simulations with only standalone VRUs is the one with most VAMs sent. However, all the three shape types presented a similar curve, with more than 60 thousand VAMs sent. This indicates that despite the cardinalities of the cluster being in the range of 7 to 9 VRUS, the amount of transmitted messages has not been greatly reduced.

FIGURE 32 – Average number of VAMs sent for each shape and for the standalone baseline



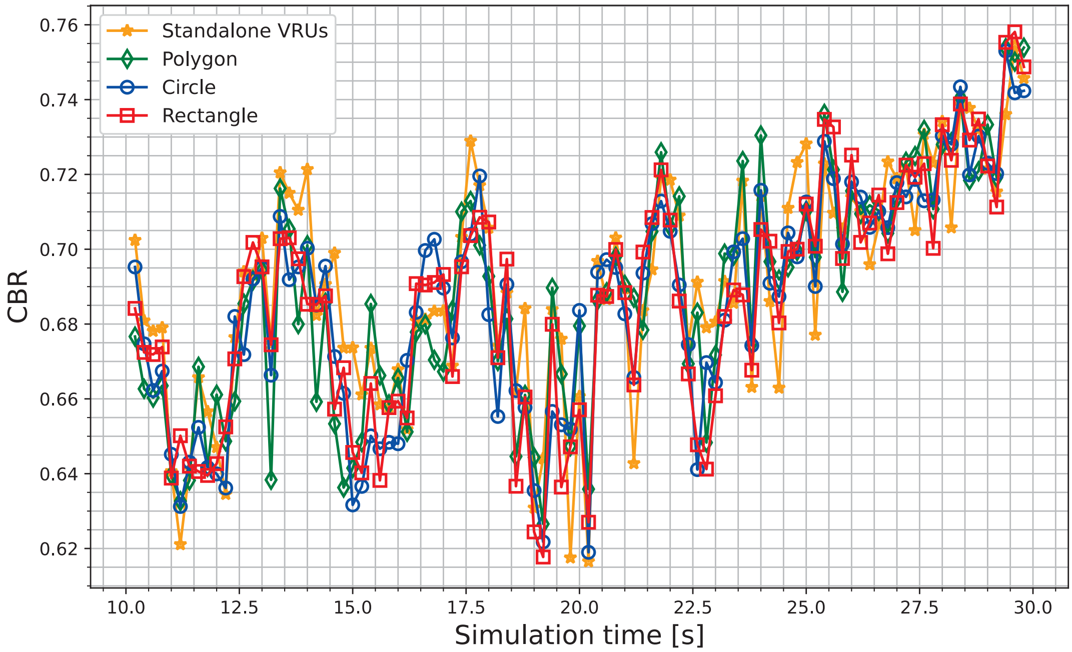
The contribution of the operation VAMs to this result is even more apparent when tallying only the VAMs from standalone VRUs and the maintenance VAMs from leaders. FIGURE 33 makes it clear that the largest amount of VAMs sent with clustering enabled is due to the cluster operations. This indicates that in order for VRU clustering to actually reduce effectively the number of messages transmitted in a network, the conditions for triggering operations has to be changed.

FIGURE 33 – Average number of VAMs sent for each shape and for the standalone baseline (excluding operations)



Furthermore, due to this issue, the assessment of the CBR for the VBS when using each of the shape types is inconclusive, since all geometries suffer from the large amount of VAMs sent in the simulated scenario, in a manner that the CBR is constrained to values close to the DCC threshold of 0.65 as seen in FIGURE 34.

FIGURE 34 – Channel Busy Ratio for each shape and for the standalone scenario



This means that in order to properly discuss in depth the Quality of Service provided by the different shape types, functional changes to the current ETSI VBS are in order. This result is very interesting, as it shows that the current technical specification requires more testing and development to be truly applicable. Changes such as different timings for the join and leave notifications, or the use of other values of distance and velocity difference could influence the amount of operations triggered. These analyses are outside the scope of this work, since present focus is on testing the methodologies to form the shapes, however they can prove to be very valuable future work in the realm of VRU Clustering.

Moreover, simulating different traffic scenarios could confirm if different bounding box types are more suited for particular use cases. This confirmation could lead, for example, to the definition and standardization of parameters to implement each of the shape-generation strategies proposed in this study. Another topic to be explored is the overlap of bounding boxes, evaluating how many times this occurs and for how long these situations persist. Long and frequent overlaps could indicate a shape as inadequate for a use case since it could pose a safety or operational issue due to the region of uncertainty generated by two clusters occupying the same area.

6 CONCLUSION

The present work has contributed to the VRU Basic Service from the ITS-G5 standard by offering bounding box formation strategies for all the shape types determined by ETSI TS 103 300. The developed methods, use a new data structure called *Cluster Map* that specializes the Local Dynamic Map, and adopt computational geometry strategies to build the bounding boxes. Summarizing these methods are Algorithms 4, 5, and 6. Additionally, a new condition to start a cluster breakup operation is introduced: a timeout for an unsuccessful cluster creation event. This event is triggered by the leader when it detects an insufficient cardinality for an elapsed time proportional to the parameter *timeClusterContinuity*.

Cluster area, cardinality, density, and also the average message size increment are selected to evaluate how each bounding box shape performs in a crowded traffic scenario. In this study, cluster density measures how well a shape type contains all its member VRUs. A higher density indicates that a cluster is capable of occupying less space to protect its participants. Average increment to the message is used to compare which shapes offer the largest increment to the VAM.

The Cluster Map, bounding box strategies, and metrics were then implemented in the *Artery* V2X simulation framework, with the use of a traffic scenario based on the InTAS model simulated through SUMO. In order to provide many opportunities to form clusters, the modeled simulation contains several pedestrians and vehicles circulating in a single crossing, and interact with each other. Twenty-four simulations were executed, encompassing the three shape types and a standalone control scenario, each with six different seeds.

Simulation results show that circular bounding boxes produce larger cluster areas and lower cluster density. The amount of cluster creations and breakups point to cluster with longer lifetimes. The shape is indicated for a use case that prioritizes smaller messages over spatial efficiency or shape representation accuracy. This is adequate for busy wireless channel scenarios, so reducing congestion is the priority. Polygons are recommended for the opposite scenarios, where the service can afford to send slightly larger messages with the benefit of representing the VRUs contained in the cluster with more detailed bounding box. Rectangles offer moderate cluster density and message size. Identifying use cases where rectangular bounding boxes are best suited poses a considerable opportunity for future work.

As future steps for this research, it is fundamental to improve strategies for the triggering of the cluster operations, as the current manner that ETSI 103 300 suggests its implementation defeats the purpose of clustering, as it results in nearly the same amount of messages as the standalone scenario. Some possible approaches are: the adjustment of the timings of the message operation notifications, so that these messages are sent less frequently; the implementation of cooldowns, so that a VRU that recently left a cluster waits some time before attempting to join a new cluster; or the use of the measurement of the CBR already done by the DCC mechanism to assess if the channel is available enough to send an operation VAM.

After such corrections are developed, the evaluation of the Quality of Service is a research path that could lead to new insights into the management of the clustering operations. For instance, an intriguing investigation would involve assessing the viability of a specific shape type when the channel occupation exceeds a designated CBR threshold.

Furthermore, another set of performance metrics needs to be determined to quantify if the safety of the pedestrians was not compromised by these techniques. For example, a potential candidate for this is the Age of Information, which is used to measure how fresh is the information received by a node. This metric takes into consideration the latency of transmission and also the inter-delivery time for data updates (LI; XIANG; GE, 2022). Using this metric it could be possible to analyze if clustering added a considerable overhead in the generation of awareness messages, thus resulting in more risk-prone situations for the clustered pedestrians. This overhead can be evaluated then, with multiple cluster shapes and cardinality sizes. One other manner to observe the increase in complexity of the cluster messages containing VAMs is through the overall simulation time using different shape configurations.

Depending on the obtained results of the simulations with these additional parameters, a new functionality for clustering on the VBS is planned: the introduction of a shape selection tool. This mode aims to assist the decision of which cluster shape type a leader should adopt based on the perceived current status of the traffic system. For example, if a VRU notices that the channel is not busy, it could opt to create a polygon bounding box to better fit its members as a larger VAM size would not be as detrimental. Some moments later, this same leader could decide to change the bounding box to a circle due to environmental changes, such as entering a region in which the channel is more congested.

REFERENCES

ARMOOGUM, Jimmy; GARCIA, Cédric; GOPALAPILLAI, Yamini; BORGATO, Stefano; FIORELLO, Davide; MAFFII, Silvia; MARS, Kees Jan; POPOVSKA, Tanja; SCHLEMMER, Lucia; VINCENT, Victoire; BOGAERT, Matthieu; GAYDA, Sylvie. **Study on New Mobility Patterns in European Cities**. Directorate-General for Mobility and Transport (European Commission). July 2022. DOI: 10.2832/728583. Available from: <https://op.europa.eu/en/publication-detail/-/publication/adfc18f1-80e1-11ed-9887-01aa75ed71a1/language-en>.

BARBOSA DA SILVA, Leonardo; LOBO, Silas; MARTÍN GARCÍA FERNÁNDEZ, Evelio; FACCHI, Christian. What Is the Right Bounding Box of a VRU Cluster in V2X Communication? How to Form a Good Shape? In: 10TH INTERNATIONAL CONFERENCE ON VEHICLE TECHNOLOGY AND INTELLIGENT TRANSPORT SYSTEMS - VEHITS, 2024, Angers. **Proceedings**. Ed. by Alexey Vinel, Karsten Berns, Jeroen Ploeg and Oleg Gusikhin. Angers: SciTePress, 2024. v. 1. INSTICC, p. 144–155. ISBN 978-989-758-703-0. DOI: 10.5220/0012699100003702.

CAR 2 CAR COMMUNICATION CONSORTIUM. **C-ITS Glossary**. 2024. Available from: <https://www.car-2-car.org/fileadmin/documents/Basic_System_Profile/Release_1.6.6/C2CCC_TR_2053_Glossary_R166.pdf>.

CAR 2 CAR COMMUNICATION CONSORTIUM. **Guidance for day 2 and beyond roadmap**. Braunschweig, Sept. 2019. Available from: <https://www.car-2-car.org/fileadmin/documents/General_Documents/C2CCC_WP_2072_RoadmapDay2AndBeyond.pdf>.

CAR 2 CAR COMMUNICATION CONSORTIUM. **The handbook for Vehicle-to-X cooperative systems simulation**. Braunschweig, 2015. Available from: <https://www.car-2-car.org/fileadmin/documents/General_Documents/C2C607001110_Handbook_for_Vehicle-to-X.pdf>.

CAR 2 CAR COMMUNICATION CONSORTIUM. **Use Cases**. Braunschweig, 2023. Available from: <https://www.car-2-car.org/fileadmin/documents/General_Documents/C2CCC_UC_2097_UseCases_V1.0.pdf>.

DECAE, Rob. **Annual statistical report on road safety in the EU 2022**. Brussels, Feb. 2023. Available from: <https://road-safety.transport.ec.europa.eu/statistics-

and-analysis/statistics-and-analysis-archive/annual-accident-report-archive_en>.

ERICSON, Christer. **Real-Time Collision Detection**. USA: CRC Press, Inc., 2004. ISBN 1558607323.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Automotive Intelligent Transport Systems (ITS)**. Sophia Antipolis: ETSI, 2022. Available from: <https://www.etsi.org/technologies/automotive-intelligent-transport?jjj=1670948393123&jjj=1670966318888>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band; V1.2.1**. Sophia Antipolis, July 2013. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/302663/01.02.01_30/en_302663v010201v.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Communications Architecture**. Sophia Antipolis, Sept. 2010. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/302665/01.01.01_60/en_302665v010101p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems operating in the 5 GHz range; Access layer part**. Sophia Antipolis, Apr. 2018. Available from: <https://www.etsi.org/deliver/etsi_ts/102600_102699/102687/01.02.01_60/ts_102687v010201p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band; V1.3.1**. Sophia Antipolis, Oct. 2019. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/302663/01.03.01_30/en_302663v010301v.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Users and applications requirements; Part 1: Facility layer structure, functional requirements and specifications**. Sophia Antipolis, Aug. 2013. Available from: <https://www.etsi.org/deliver/etsi_ts/102800_102899/10289401/01.01.01_60/ts_10289401v010101p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); V2X Applications; Part 1: Road Hazard Signalling (RHS) application requirements specification**. Sophia Antipolis, Aug. 2013. Available from: <https://www.etsi.org/deliver/etsi_ts/101500_101599/10153901/01.01.01_60/ts_10153901v010101p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2**. Sophia Antipolis, Dec. 2019. Available from: <https://www.etsi.org/deliver/etsi_tr/103500_103599/103562/02.01.01_60/tr_103562v020101p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service**. Sophia Antipolis, Apr. 2019. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.04.01_60/en_30263702v010401p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service**. Sophia Antipolis, Apr. 2019. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.03.01_60/en_30263703v010301p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking**. Sophia Antipolis, Apr. 2014. Available from: <https://www.etsi.org/deliver/etsi_en/302600_302699/30263601/01.02.01_60/en_30263601v010201p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 1: Use Cases definition; Release 2**. Sophia Antipolis, Apr. 2021. Available from: <https://www.etsi.org/deliver/etsi_tr/103300_103399/10330001/02.02.01_60/tr_10330001v020201p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 2: Functional Architecture and Requirements definition; Release 2**. Sophia Antipolis, Apr. 2021. Available from: <https://www.etsi.org/deliver/etsi_ts/103300_103399/10330002/02.02.01_60/ts_10330002v020201p.pdf>.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Intelligent Transport Systems (ITS); Vulnerable Road Users (VRU) awareness; Part 3: Specification of VRU awareness basic service; Release 2**. Sophia Antipolis, Apr. 2021. Available from: <https://www.etsi.org/deliver/etsi_ts/103300_103399/10330003/02.01.02_60/ts_10330003v020102p.pdf>.

FESTAG, Andreas. Standards for vehicular communication from IEEE 802.11p to 5G. **e & i Elektrotechnik und Informationstechnik**, Springer Science and Business Media LLC, v. 132, n. 7, p. 409–416, Sept. 2015. DOI: 10.1007/s00502-015-0343-0. Available from: <https://doi.org/10.1007/s00502-015-0343-0>.

FOLLMER, Robert; GRUSCHWITZ, Dana. **Mobility in Germany – short report. Edition 4.0**. DLR, IVT, infas 360 on behalf of the Federal Ministry of Transport, and Digital Infrastructure (BMVI). Sept. 2019. Available from: <https://bmdv.bund.de/SharedDocs/DE/Anlage/G/mid-2017-short-report.pdf?__blob=publicationFile>.

GOMES, Eduardo Luis; FONSECA, Mauro; LAZZARETTI, Andre Eugenio; MUNARETTO, Anelise; GUERBER, Carlos. Clustering and Hierarchical Classification for High-Precision RFID Indoor Location Systems. **IEEE Sensors Journal**, v. 22, n. 6, p. 5141–5149, Mar. 2022. ISSN 1530-437X. DOI: 10.1109/JSEN.2021.3103043.

LAKSHMANAN, Valliappa; GÖRNER, Martin; GILLARD, Ryan. **Practical Machine Learning for Computer Vision**. 1. ed. Sebastopol: O'Reilly Media, Inc, July 2021. ISBN 9781098102333.

LI, Zipeng; XIANG, Lin; GE, Xiaohu. Age of Information Modeling and Optimization for Fast Information Dissemination in Vehicular Social Networks. **IEEE Transactions on Vehicular Technology**, v. 71, n. 5, p. 5445–5459, 2022. DOI: 10.1109/TVT.2022.3154766.

LIU, Yang; LIU, Bingbing; ZHANG, Hongbo. Estimation of 2D Bounding Box Orientation with Convex-Hull Points - A Quantitative Evaluation on Accuracy and Efficiency. In: 2020 IEEE INTELLIGENT VEHICLES SYMPOSIUM (IV), 2020, Las Vegas. **Proceedings**. Las Vegas: IEEE, Jan. 2021. P. 945–950. ISBN 978-1-7281-6673-5. DOI: 10.1109/IV47402.2020.9304788.

LOBO, Silas; BARBOSA DA SILVA, Leonardo; FACCHI, Christian. To Cluster or not to Cluster: A VRU Clustering Based on V2X Communication. In: 2023 IEEE 26TH INTERNATIONAL CONFERENCE ON INTELLIGENT TRANSPORTATION SYSTEMS (ITSC), 2023, Bilbao. **Proceedings**. Bilbao: IEEE, Feb. 2024. P. 2218–2225. DOI: 10.1109/ITSC57777.2023.10422659.

LOBO, Silas; FESTAG, Andreas; FACCHI, Christian. Enhancing the Safety of Vulnerable Road Users: Messaging Protocols for V2X Communication. In: 2022 IEEE 96TH VEHICULAR TECHNOLOGY CONFERENCE (VTC2022-FALL), 2022, London. **Proceedings**. London: IEEE, Jan. 2023. P. 1–7. DOI: 10.1109/VTC2022-Fall57202.2022.10012775.

LOBO, Silas; NEUMEIER, Stefan; FERNÁNDEZ, Evelio; FACCHI, Christian. InTAS - The Ingolstadt Traffic Scenario for SUMO. In: SUMO USER CONFERENCE 2020, 2020. **SUMO Conference Proceedings**. Virtual event: TIB Open Publishing, June 2020. P. 73–92. DOI: 10.52825/scp.v1i.102. Available from: <https://sumo.dlr.de/2020/SUMO2020_paper_24.pdf>.

LUSVARGHI, Luca; GRAZIA, Carlo Augusto; KLAPEZ, Martin; CASONI, Maurizio; MERANI, Maria Luisa. Awareness Messages by Vulnerable Road Users and Vehicles: Field Tests Via LTE-V2X. **IEEE Transactions on Intelligent Vehicles**, p. 1–15, 2023. ISSN 2379-8904. DOI: 10.1109/TIV.2023.3280744.

MANNONI, Valerian; BERG, Vincent; SESIA, Stefania; PERRAUD, Eric. A Comparison of the V2X Communication Systems: ITS-G5 and C-V2X. In: 2019 IEEE 89TH VEHICULAR TECHNOLOGY CONFERENCE (VTC2019-SPRING), 2019, Kuala Lumpur. **Proceedings**. Kuala Lumpur: IEEE, June 2019. P. 1–5. ISBN 978-1-7281-1217-6. DOI: 10.1109/VTCSpring.2019.8746562.

MOLINA-MASEGOSA, Rafael; GOZALVEZ, Javier; SEPULCRE, Miguel. Comparison of IEEE 802.11p and LTE-V2X: An Evaluation With Periodic and Aperiodic Messages of Constant and Variable Size. **IEEE Access**, v. 8, p. 121526–121548, 2020. ISSN 2169-3536. DOI: 10.1109/ACCESS.2020.3007115.

RIEBL, Raphael. **Artery Architecture**. 2020. Available from: <http://artery.v2x-research.eu/architecture/>.

RIEBL, Raphael. **Vanetza**. 2021. Available from: <https://www.vanetza.org/>.

RIEBL, Raphael; GÜNTHER, Hendrik-Jörn; FACCHI, Christian; WOLF, Lars. Artery: Extending Veins for VANET applications. In: 2015 INTERNATIONAL CONFERENCE ON MODELS AND TECHNOLOGIES FOR INTELLIGENT TRANSPORTATION SYSTEMS (MT-ITS), 2015, Budapest. **Proceedings**. Budapest: IEEE, Aug. 2015. P. 450–456. DOI: 10.1109/MTITS.2015.7223293.

RIEBL, Raphael; OBERMAIER, Christina; GÜNTHER, Hendrik-Jörn. Artery: Large Scale Simulation Environment for ITS Applications. In: **Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem**. Ed. by Antonio Virdis and Michael Kirsche. Cham: Springer International Publishing, 2019. P. 365–406. ISBN 978-3-030-12842-5. DOI: 10.1007/978-3-030-12842-5_12. Available from: <http://link.springer.com/10.1007/978-3-030-12842-5_12>.

RUPP, Matthias; WISCHHOF, Lars. Evaluation of the Effectiveness of Vulnerable Road User Clustering in C-V2X Systems. In: 2023 IEEE INTERNATIONAL CONFERENCE ON OMNI-LAYER INTELLIGENT SYSTEMS (COINS), 2023, Berlin. **Proceedings**. Berlin: IEEE, July 2023. P. 1–5. ISBN 979-8-3503-4647-3. DOI: 10.1109/COINS57856.2023.10189204. Available from: <https://ieeexplore.ieee.org/document/10189204/>.

SHAHRAKI, Amin; TAHERKORDI, Amir; HAUGEN, Øystein; ELIASSEN, Frank. Clustering objectives in wireless sensor networks: A survey and research direction analysis. **Computer Networks**, v. 180, p. 107376, Oct. 2020. ISSN 13891286. DOI: 10.1016/j.comnet.2020.107376.

SHAMOS, Michael Ian. **Computational Geometry**. 1978. PhD thesis – Yale University.

SLOOTMANS, Freya. **Facts and Figures – Pedestrians - 2023**. Brussels, Oct. 2021. Available from: <https://road-safety.transport.ec.europa.eu/statistics-and-analysis/data-and-analysis/facts-and-figures_en>.

SOMMER, C; GERMAN, R; DRESSLER, F. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. **IEEE Transactions on Mobile Computing**, v. 10, n. 1, p. 3–15, Jan. 2011. ISSN 1536-1233. DOI: 10.1109/TMC.2010.133.

UNITED NATIONS - DEPARTMENT OF ECONOMIC AND SOCIAL AFFAIRS. **World Urbanization Prospects: The 2018 Revision**. 2018. Available from: <https://population.un.org/wup/Download/>.

V, Priya M; PANKAJ, Dhanya S. 3DYOLO: Real-time 3D Object Detection in 3D Point Clouds for Autonomous Driving. In: 2021 IEEE INTERNATIONAL INDIA GEOSCIENCE AND REMOTE SENSING SYMPOSIUM (INGARSS), 2021, Ahmedabad. **Proceedings**. Ahmedabad: IEEE, June 2022. P. 41–44. ISBN 978-1-6654-4249-7. DOI: 10.1109/InGARSS51564.2021.9791912.

WANG, Zhangfei; YU, Bo; CHEN, Jian; LIU, Chunyang; ZHAN, Kun; SUI, Xin; XUE, Yujun; LI, Jishun. Research on Lidar Point Cloud Segmentation and Collision Detection Algorithm. In: 2019 6TH INTERNATIONAL CONFERENCE ON INFORMATION SCIENCE AND CONTROL ENGINEERING (ICISCE), 2019, Shanghai. **Proceedings**. Shanghai: IEEE, June 2020. P. 475–479. ISBN 978-1-7281-5712-2. DOI: 10.1109/ICISCE48695.2019.00101.

XIOG, Guangming; YANG, Tian; LI, Minghong; ZHANG, Yu; SONG, Weilong; GONG, Jianwei. A Novel V2X-based Pedestrian Collision Avoidance System and the Effects Analysis of Communication Delay and Packet Loss on Its application. In: 2018 IEEE INTERNATIONAL CONFERENCE ON VEHICULAR ELECTRONICS AND SAFETY (ICVES), 2018, Madrid. **Proceedings**. Madrid: IEEE, Nov. 2018. P. 1–6. ISBN 978-1-5386-3543-8. DOI: 10.1109/ICVES.2018.8519600.

ZHU, Rongbo; YU, Mai; LI, Yuanli; WANG, Jun; LIU, Lu. Edge Sensing-Enabled Multistage Hierarchical Clustering Deredundancy Algorithm in WSNs. **Wireless Communications and Mobile Computing**, v. 2021, p. 1–14, Mar. 2021. ISSN 1530-8677. DOI: 10.1155/2021/6664324.

ZOGHLAMI, Chaima; KACIMI, Rahim; DHAOU, Riadh. Dynamics of Cooperative and Vulnerable Awareness Messages in V2X Safety Applications. In: 2022 INTERNATIONAL WIRELESS COMMUNICATIONS AND MOBILE COMPUTING (IWCMC), 2022, Dubrovnik. **Proceedings**. Dubrovnik: IEEE, July 2022. P. 853–858. ISBN 978-1-6654-6749-0. DOI: 10.1109/IWCMC55113.2022.9824685.