Performance of Opportunistic Data Networks Based on Public Transportation Systems

Eduardo R. Manika, Emilio C. G. Wille, Joilson Alves Jr

***Abstract*—Smart Cities are those that leverage technology to enhance the welfare of their citizens, foster economic development, and promote sustainability. The application of these technologies in road transportation services has been conducted within programs known as *Intelligent Transportation Systems* (ITS). *Vehicular Ad Hoc Networks* (VANETs) are a key part of the ITS framework. A *Public Transportation System* (PTS) is a system of vehicles, such as buses, that operate regularly on fixed routes and are used by the public. However, the transportation literature does not provide acceptable models for bus movements in an urban environment since they are affected by vehicular and passenger traffic conditions, lane organization, traffic signal management, company policies, etc. In this paper, we are interested in assessing the performance of the data network deployed on a real-world PTS with a view mainly to real-time communications. Therefore, we use the *Engine for Map Matching to SUMO* (EMMS) – a computational tool that automatically imports the PTS characteristics into a simulation environment – to study the system performance. Specifically, geolocation data from public transportation buses in Curitiba (Brazil) are used. The obtained results show that the formation of small temporary networks is possible, allowing data exchange between buses.**

***Index Terms*—Smart cities, Public transportation system, Geopositioning data, Map-matching, Statistical analysis.**

I. INTRODUCTION

SMART Cities are those which use technology to promote the well-being of the population, economic growth and, at the same time, improve sustainability. Besides technological development, other sectors are involved, such as urban planning, social housing, urban mobility, transportation, energy, air pollution control, among others [1], [2]. As a result of technological advances in electronic communication and information equipment, the automation of transportation and end-user information systems has been experiencing a rapid evolution. The application of these technologies in road transportation is being conducted within programs known as *Intelligent Transportation Systems* (ITS) [3]. In general, ITS must provide an intelligent connection between users of the transportation systems, the vehicles, and the infrastructure.

Vehicular Ad Hoc Networks (VANETs) are an important part of ITS and comprehend communication between vehicles themselves and between vehicles and fixed components along the way (routers, gateways, and services) [4]–[6]. Each vehicle in the network is equipped with a wireless communication device that can communicate with other devices in other vehicles via radio signals. Besides, vehicles can also act as a routers, receiving and resending messages to their destination, thereby passing on information.

In a *Public Transportation System* (PTS), vehicles (buses) travel between stops according to a predetermined schedule. Each vehicle has an itinerary, which is the ordered sequence of stops that the vehicle traverses. A bus line corresponds to a set of different vehicles with the same itinerary. Understanding mobility of vehicles in PTSs is of utmost importance. For example, the possibility of wireless communication between buses (nodes) is strongly influenced by the vehicle mobility [7], [8]. This provides a bus-based VANET; a kind of opportunistic data network. However, the transportation literature does not provide acceptable models for bus movements in an urban environment since they are affected by vehicular and passenger traffic conditions, lane organization, traffic signal management, company policies, and others.

VANET-related research is deeply associated with the use of mobility simulators, as real-world testing takes time and financial resources. Among the mobility simulators frequently mentioned in the literature, it is worth mentioning the *Simulator of Urban Mobility* (SUMO) [9], which is an open-source tool that provides functions that allow the generation of vehicular traffic according to a probability distribution or through external data inputs (obtained from actual traffic data collections). Importing the actual maps of cities and their characteristics is a task often performed manually. However, manually creating such scenarios can make simulations impractical and unreliable. Regarding scenarios involving PTS in large centers, manual creation is still more costly due to unique characteristics such as previously established vehicle trajectories, specific bus stops, and defined departure schedules. Accordingly, to perform the transcription between actual (georeferenced) data and a digital map, *map-matching* techniques (a part of the so-called *trajectory data mining* [10]) should be employed. However, when PTSs are considered, employing incomplete models severely risks to produce unreliable results.

In order to facilitate the development of scenario simulations involving a PTS, this paper presents a tool (developed by the authors) called *Engine for Map-Matching to SUMO* (EMMS). This tool automatically imports the following characteristics of the transportation system into the simulated environment (SUMO): bus line routes, bus stop locations, and bus departure times. That allows for more agile, reliable, and realistic simulations. Given the availability of this tool, numerous studies can be conducted. We are interested in assessing the performance of the data network deployed on the PTS

E.R. Manika, E.C.G. Wille and J. Alves Jr are with Federal University of Technology – Paraná (UTFPR), Av. Sete de Setembro 3165, 80230-901, Curitiba (PR), Brazil. ORCID: 0000-0002-0247-8417, 0000-0003-1916-0031, 0000-0001-9934-4441. E-mail: manika@utfpr.edu.br, ewille@utfpr.edu.br, joilson@utfpr.edu.br

Digital Object Identifier: 10.14209/jcis.2024.9

with a view mainly to real-time communications. Real-time communications offer some guarantee regarding the delay or speed with which messages are delivered to their destination. Moreover, we have considered actual data from the geolocation records of public transportation buses of Curitiba (a southern Brazilian city well-known for its bus-based transportation system).

The main contributions of the paper are summarized as follows.

- It describes a computational tool to automate and facilitate the construction of scenarios for simulating PTSs. It should be noted that manually creating scenarios involving a large number of buses is very difficult.
- It considers actual bus geolocation data from a PTS to conduct simulations. This leads to much more realistic simulations than those based on classical randomness-mobility models.
- It validates the proposed system by conducting a data comparison between results obtained from simulations and realistic data.
- It conducts extensive simulations considering a set of metrics to analyze the data network deployed on a real-world PTS.

The remainder of this paper is organized as follows. Section II gives a review of related work. The description of EMMS is given in Section III. Section IV presents the study methodology adopted, and the results obtained through computational simulations are given in Section V. Finally, Section VI draws conclusions and future work.

II. RELATED WORK

One of the challenges related to vehicular network studies is the definition of a suitable model that can provide an accurate and realistic description of vehicular mobility. A realistic mobility model should consider the following characteristics [11], [12]:

- Realistic topological maps.
- Existence of obstacles.
- Departure and arrival points.
- Acceleration and braking.
- Day periods.
- Non-random vehicle distribution.
- Driving behavior.

Vehicular mobility models may be classified considering (a) *detailing level* and (b) *motion generating*. According to the level of detail they provide, a mobility model may be classified as *macroscopic* and *microscopic* [12]. The macroscopic model considers a more general approach, dealing with the road topology and characteristics like lane speed and density. These models are rarely used in network simulation, as they cannot capture the individual node behavior [13]. On the other hand, the microscopic model focuses on the individual behavior of drivers when interacting with other drivers or with the road infrastructure. According to how vehicle movements are generated, a mobility model may be classified as *synthetic*, *trace-based*, and *simulator-based* [14]. In synthetic type, the mobility is based on mathematical models of the physical motion of vehicles. Trace-based models depend on collecting motion traces employing communication devices and then extracting mobility patterns from these traces. In the simulator-based case, computer simulators are used to model microscopic traffic over an urban map, thus restraining the movement of vehicles and providing a more realistic approach. The *Random Way-point* and the *Random Direction* mobility models, in which the vehicles move in a purely random motion, are examples of synthetic models [15]. The SUMO software is an example of a simulator-based model.

Due to the particular characteristics of the movement of the buses belonging to a PTS, randomness-based models turn out inappropriate. The solution consists of simulating bus lines using SUMO, which enables the simulations from the simplest networks to more complex environments. The micro-mobility presented in SUMO offers a collision-free system in which a vehicle's speed is determined by the speed of the vehicle straight ahead. In a scenario with pathways with more than one lane, this tool accepts the action of overtaking. Another important feature of SUMO is the possibility of manually creating simulation maps or importing accurate maps (road scenarios from matching information previously obtained from other services, such as *OpenStreetMap* [16]). In the latter case, the *map-matching* method can come into play. The map-matching method consists of identifying a match in a sequence of points (geolocation coordinates) to a digital map [17], [18]. The material presented in [19] shows a literature review of map-matching methods.

Some recent studies, mainly investigate bus-based VANETs, are listed below. The study conducted in [20] investigates whether a public transportation network used as a VANET backbone can be a viable solution, as it presents a large coverage area and pre-programmed paths (itineraries) repeatedly run. This work aimed to compare the routing performance of a distance-vector when applied to a more realistic environment. Experiments were conducted using data from a real-world urban area in Milan, Italy. A large number of nodes were simulated at actual time-tables of the public transportation.

In [21], the authors study how to perform efficient routing in an urban architecture where buses and cars represent two different tiers in the network. They propose a topology-based routing approach, which utilizes buses as the mobile backbone. The simulation area is based on the real map of a district in Beijing, China. Mobility traces (based on synthetic bus routes) were generated by MOVE.

In [22], the authors presented contributions to enabling vehicular network simulation by adopting more realistic road traffic representations. These contributions provide a ready-to-use realistic dataset that can be input to network simulators and facilitate the generation and validation of vehicular mobility traces for networking research. They consider as a scenario a portion of the city of Bologna, Italy. The resulting dataset represents the mobility of tens of thousands of vehicles in a 25-km area during one traffic peak hour in a typical morning.

TABLE I: Main characteristics of the simulation models.

Paper	Detailing Level	Motion Generating	Topological Map	Vehicle Type	Vehicle Route	Vehicle Scheduling	Real-time Comm.
[20]	microscopic	trace-based	real-world	bus	real-world	time-table	no
[21]	microscopic	simulator-based	real-world	bus / car	synthetic	random	no
[22]	microscopic	simulator-based	real-world	car	real-world	real-world	no
[23]	microscopic	trace-based / simulator-based	real-world	bus	real-world / synthetic	real-world / synthetic	no
[24]	microscopic	simulator-based	real-world	bus / RSU	real-world	random	no
[25]	microscopic	simulator-based	synthetic	bus / car	real-world	random	yes
[26]	microscopic	simulator-based	real-world	bus / car	synthetic	random	no
[27]	microscopic	trace-based	real-world	bus / taxi	real-world	synthetic	no

The study in [23] focuses on the geocast in bus-based VANETs and presents a geocast routing mechanism named Vela. Vela exploits bus trajectories to build a probabilistic spatial-temporal graph model and provides the available routing paths with the best possible quality-of-service levels for data delivery requests. Based on the real map of Haidian district, Beijing, experiments with real-world and synthetic bus trajectories were conducted. The results of the experiments show that Vela performs much better in terms of delivery ratio and delay and has more robust scalability than other solutions.

The authors in [24] propose GeOpps-N, a hybrid routing protocol for communications between buses and operation control centers in a PTS. They model the system as a VANET, which includes buses and RSUs. GeOpps-N searches for relaying nodes that can efficiently transport or relay the data to the closest RSU. A vehicular traffic scenario using a geographical area of the Transantiago (Chile) PTS was used to evaluate the performance of GeOpps-N. SUMO software was used to model the physical movement of buses.

In [25], the authors consider public transport buses that travel through exclusive lanes with relatively regular schedules. This fact was used to establish a cheap and reliable wireless communication infrastructure (called MOB-NET). The paper also presents the P-DSDV, a new routing protocol that exploits the inherent connectivity of MOB-NET, minimizing the effects of lack of connectivity and improving network performance. A central avenue and neighboring streets (in Curitiba, Brazil) were reproduced using the VanetMobiSim software. The study also considered standard vehicles. An extensive group of simulations has shown the superior performance of the proposal.

The authors in [26] propose a new routing scheme called *Bus-based Routing Technique* (BRT), which exploits the periodic and predictable movement of buses to learn the required time (the temporal distance) for each data transmission to Road-Side-Units (RSUs) through a dedicated bus-based backbone. BRT selects the best combinations of buses and other types of vehicles to optimize packet delivery ratios and end-to-end delays. The paper considers a topology based on a real-world map generated from OpenStreetMap and mobility traces (based on synthetic bus routes) generated by SUMO.

The effect of several parameters (message copies, buffer size, message interval, and node mobility) on the performance of opportunistic networks is evaluated in [27] using simulation and machine learning techniques. The authors collected data from real-world scenarios in three cities using GPS coordinates of taxis and buses. The ONE tool was used to simulate the scenarios [28].

Table I summarizes the characteristics of the simulation models presented in related works (i.e., what is the mobility model used, the topology, and vehicle characteristics) and how the system is evaluated (i.e., real-time communications or not).

Accordingly, the main shortcomings of the previous works are as follows:

- The mobility models considered, although accounting for real topological maps, make use of some synthetic or random characteristic. Such incomplete models increase the risk of producing unreliable results.
- In some cases, the network takes into account not only buses but other vehicles (cars, taxis, and also RSUs). In other words, it is no longer a purely bus-based network. This significantly alters connectivity and, consequently, network performance.
- These works do not usually consider real-time communications.

In this work, we conducted a comprehensive evaluation study by developing more realistic simulation scenarios (using the EMMS tool) and real-time communications metrics.

III. THE EMMS SOFTWARE TOOL

This section describes the *Engine for Map-Matching to SUMO* (EMMS), a software tool that automatically imports the characteristics of the actual vehicle mobility environment to the simulated environment (SUMO), including the routes of bus lines, the location of bus stops and the departure times. The EMMS was developed using the Python programming language [29], which allows the integration with SUMO through a set of programming libraries called *SUMOLib* [9] and is divided into four architectural layers:

- **First Layer**: Data entry and processing.
- **Second Layer**: Matching of bus itinerary maps.
- **Third Layer**: Matching of bus stops maps.
- **Fourth Layer**: Creation of configuration files for execution in SUMO.

The flowchart in Fig. 1 provides an overview of the EMMS, depicting the data processing carried out at each layer of the tool.

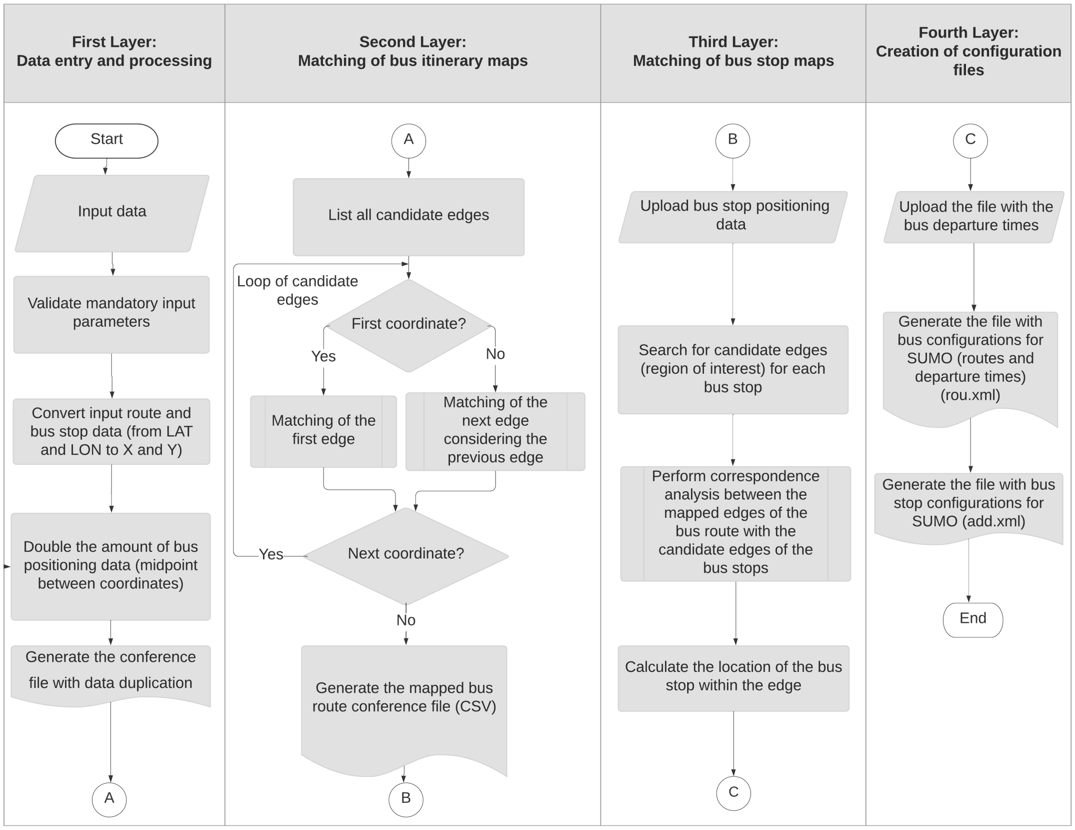


Fig. 1: EMMS flowchart.

A. *First Layer: Data Entry and Processing*

In this layer, data are entered and processed. The mandatory data are:

- Unique bus line identifier (identifying code).
- File containing the topology of the area considered for simulation (road network map), in the format already converted for SUMO (*net.xml*).
- File containing the route coordinates of the bus in GPS format, with latitude and longitude information.
- File containing the bus line departure timetables.
- File with the location of the bus line stops.

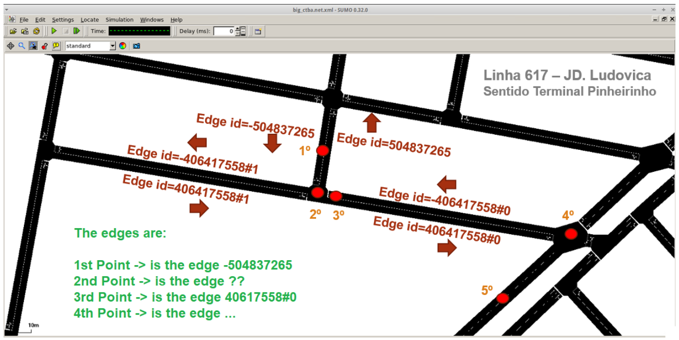
Input files must be in JSON format (text format), following a predefined pattern. After validating the mandatory parameters, the conversion of itinerary positioning and bus stops data is carried out from geographical coordinates (GPS - latitude and longitude) to Cartesian coordinates (X and Y). In this conversion, SUMO libraries are used, taking into account the file loaded into the data entry containing the area topology. In order to increase the accuracy and improve efficiency in matching maps, EMMS doubles the number of positioning points for bus routes provided in the data entry. This duplication increases the matching accuracy since urban regions have a high density, and in some situations, positioning points (GPS) provided present a significant dispersion. Such duplication is carried out by calculating the midpoint between two Cartesian coordinates. As a result, this layer generates a conference file representing bus line routes (with the data already duplicated). This file will be used by the other EMMS layers.

B. *Second Layer: Bus Itinerary Matching*

The second layer deals specifically with matching bus itineraries (routes). This itinerary is the trajectory of vehicles (buses) on a particular bus line. In order to get this trajectory in a virtual environment (such as SUMO), it is necessary to detect the route segments (edges) in a digital environment, which correspond to GPS coordinates (latitude and longitude) for that route.

Fig. 2 graphically exemplifies the matching of SUMO maps for a particular bus line. The red dots are positioning data (GPS) of the buses, representing the itinerary of a particular bus line. Each of the *edges* on the map has a unique identifier (*id*), which represents a street segment and can be formed by one or more pathways (called *lanes*). Map-matching consists of identifying which *id* corresponds to the positioning data point. Finally, a list of chained edges is generated (represented in green in the figure), which comprise the bus trajectory, i.e., the bus line itinerary within the simulation environment corresponding to the actual data. This chain of edges depends on a “physical” (interconnection between edges) and “logical” (related to the direction of edges) interconnection.

The physical interconnection refers to the connection of the edges in the digital map, i.e., the end of a selected edge must be the beginning of the subsequent edge. In SUMO, this interconnection is validated through identifiers called *nodes*, which represent the ends of an edge. The logical interconnection, on the other hand, refers to the direction and conversion permissions of the edge, i.e., it represents traffic signs. In SUMO, the nodes are used to identify the direction of the edges. During the edge configuration each node receives a unique identifier, the source node (called *from*) and the target node (called *to*) are specified. The configuration of conversion permissions is performed through the element called *connection*, which references the allowed edges.



The edges are:

1st Point -> is the edge -504837265
2nd Point -> is the edge ??
3rd Point -> is the edge 40617558#0
4th Point -> is the edge ...

Fig. 2: Map-matching example (route edge determination).

In order to automate the matching of the bus itinerary maps, EMMS takes the following three steps:

1) Listing of all candidate edges: The algorithm determines a region of interest for each entry positioning point (GPS). Such region corresponds to a perimeter with a known radius (predefined in EMMS in 15 meters). The edges within this region are selected as candidates [19]. The Euclidean distance between the positioning point and the line drawn between the nodes ("from" and "to") of the edges is used to accomplish this discovery.

2) Matching of the first edge: The matching of the first edge is essential for the whole map-matching process. In this situation, there is no previous reference point (like another edge), thus requiring a specific treatment. Most of the time, it is not possible to separately identify the initial direction the bus will move, so it is necessary to consider the next positioning point. It is known that the bus moves from the first towards the second positioning point and that the sequence is always from the node "from" (origin) to the node "to" (destination). Therefore, it is possible to correctly map the first edge by validating whether the edge direction corresponds to the actual direction of the vehicle. One way of analyzing the edge direction is by checking the Euclidean distances between positioning points (first and second) against the nodes ("from" and "to") of the candidate edge selected. In general, EMMS weighs the initial matching, first selecting the shortest distance edge among the candidates. Subsequently, EMMS verifies whether the positioning points are approaching or moving away from the candidate edge node. If the selected edge does not satisfy any of the conditions mentioned above, the selected edge is discarded, and the process restarts with the next candidate edge.

Fig. 3 shows one of the criteria analyzed, and if the second positioning point is closer than the first one to the node "to", this represents a high probability that the edge corresponds to the actual environment because the node "to" is always the final destination of the edge. Another situation of correct matching is presented in Fig. 4, where the first positioning point is closer to the node "to" and the second positioning point is moving away from the node "to", also on another edge. As a result of this step, EMMS returns the first matched edge, which will be used to define the next matchings.

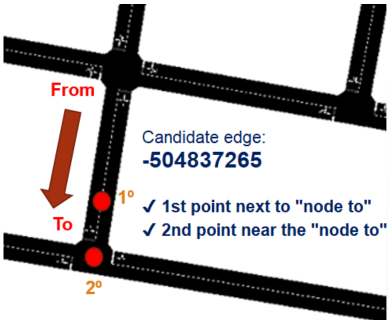


Fig. 3: Analysis for matching the first edge (first case).

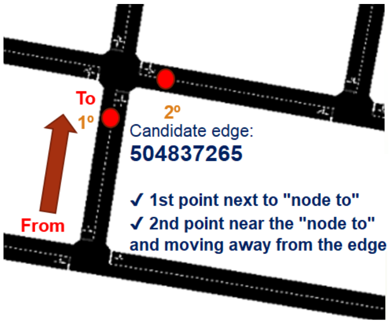


Fig. 4: Analysis for matching the first edge (second case).

3) Matching of the next edge, considering the previous one: This step consists of matching all the other positioning points, considering the edge selected in the matching process of the immediately preceding positioning point. Notably, the dependence between the edge connections, both physical and logical, is one of the operational characteristics of SUMO. Both conditions need to be met so that the simulation will be performed successfully. For this reason, the manual matching process is costly, sometimes making the generation of simulations unfeasible. On the other hand, EMMS enables the automation of these matchings, performing the necessary validations (both physical and logical), making the whole process more agile.

Thus, the logic of this stage is divided into two phases. Firstly, the candidate edge with the shortest distance to the positioning point is retrieved, and subsequently, the physical and logical interconnections between the edges (candidate and selected in the previous process) are analyzed. The physical interconnection is validated by comparing the *ids* of node "to" of the matched edge of the previous positioning point and the node "from" of the selected candidate edge. Once this step is validated, the analysis of the logical interconnection is performed, checking for connection registration on the digital map (*net.xml*). If it does not exist, it means that although there is a physical connection between the edges, the continuity towards the candidate edge is not permitted, or conversion is not allowed. Therefore, the selected candidate edge is discarded, and the process restarts, seeking the next candidate edge. If the logical interconnection exists (has a record), the edge is added to the chained list of edges and will be used to match the subsequent positioning point.

At the end of the matching process, the sequence of edges representing the buses' itineraries in the real environment will be ready for SUMO. In addition, a file for manual conference will be generated. Fig. 5 shows a comparison between an actual bus line route and the same itinerary matched in EMMS. It is noteworthy that the plotted coordinates (red points with blue borders) are already duplicated in the actual itinerary, considering the midpoint. Conversely, the coordinates plotted as a result of EMMS consider the positioning of the nodes of the matched edges.

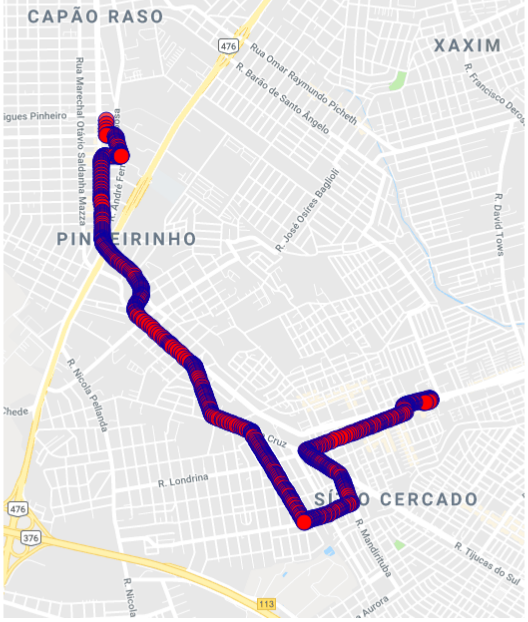
C. Third Layer: Matching the Bus Stop Points

This layer consists of matching the bus stop points to the corresponding edges of the itinerary matching process carried out in the second layer, as well as locating the bus stop along the edge matched.

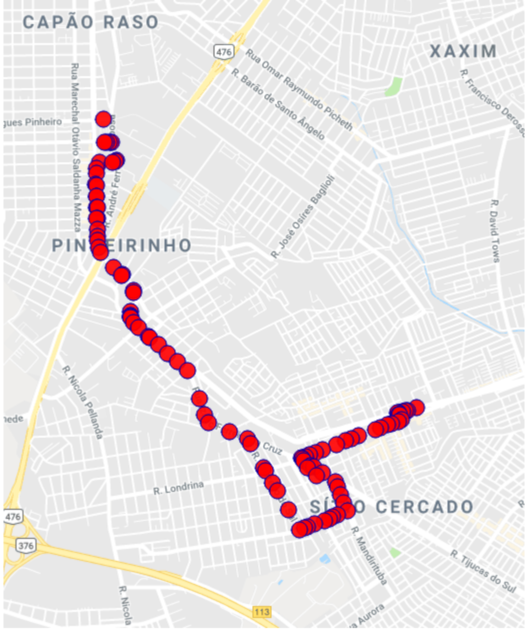
Firstly, EMMS loads the bus stop location data (provided in the first layer). Subsequently, it carries out a search of the candidate edges within the region of interest, close to each bus stop point indicated, considering the same radius configured at the data entry (Section III-A). After the candidate edges search, an analysis of the correspondence between the candidate edges and the edges mapped in the bus line route matching process (Section III-B) is performed. If there is a correspondence between them, the bus stop is matched. The configuration of the bus stop location within the edge is performed by calculating the (Euclidean) distance from the bus stop to the node "from" of the corresponding edge. Thus, it is possible to identify the starting point of the bus stop in the virtual environment.

D. Fourth Layer: Creation of Configuration Files for SUMO

The fourth layer is responsible for automating the generation of configuration files required to run simulations in SUMO. In that process, are generated two files: one is responsible for setting up the buses in the simulation (*rou.xml*) and the other is responsible for configuring the bus stops for the simulation environment, in addition to the configuration of the specific location of the stop point inside the edge (*add.xml*).



(a) Actual itinerary of a bus line.



(b) EMMS matched itinerary.

Fig. 5: Graphical comparison between a real bus route versus a route matched by EMMS.

IV. METHODOLOGY

We adopted the following methodology to simulate more realistically a data network deployed on a public transport system. We have considered the city's regions (urban areas) with the most prominent bus terminals. All bus lines that serve these terminals were simulated. We used EMMS to determine the routes of all the buses. Finally, we performed simulations and analyzed the results from the NS-2 software. Details about the scenarios considered, the metrics, and the setting of computational simulations are given in the following.

A. Urban Scenarios

The urban areas considered include the two largest bus terminals in Curitiba. In the first area (the central one) buses depart from *Rui Barbosa Square*, totalizing 21 bus lines and 98 buses traveling in the morning and 100 in the afternoon (Fig. 6). In the second region (the southern one) buses depart from *Pinheirinho Terminal*, totalizing 18 lines and 53 buses in the morning and 52 buses in the afternoon (Fig. 7). The simulations were carried out using the actual geolocation data of the public transportation buses of the city of Curitiba (these data were made available by the public transportation company – URBS [30]). The analyzed data refer to October 25th, 2022 (Tuesday) from 06:00 to 10:00 and from 16:00 to 20:00. These are high-traffic periods, allowing us to evaluate the system in the best possible condition.

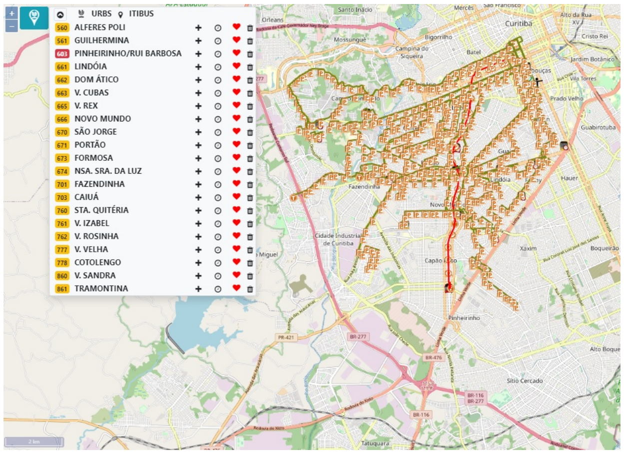


Fig. 6: Rui Barbosa Square - Bus lines and itineraries.

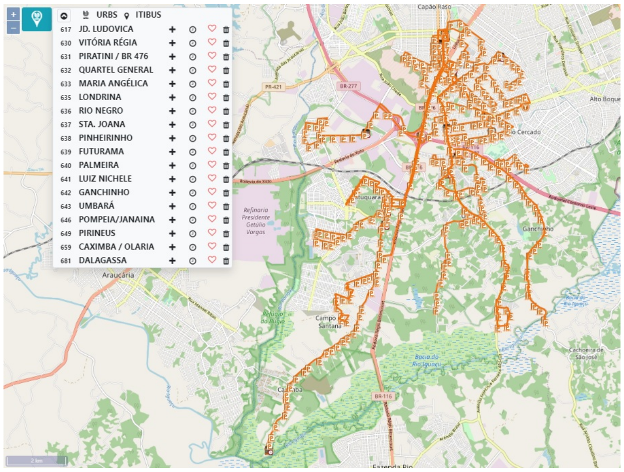


Fig. 7: Pinheirinho Terminal - Bus lines and itineraries.

For the validation of the mobility simulation environment, compatibility analyses between the buses' actual travel times (made available by URBS) and the simulated travel times (considering the *mobility trace* generated by SUMO) were carried out. The bus routes (necessary for SUMO) were obtained using EMMS. In general, the central region (Rui Barbosa) presented a compatibility percentage of 90.56%, while in the southern region (Pinheirinho), it was 91.62% in the morning. In the afternoon, the compatibility percentage in the central region was 95.24%, and in the southern region it was 89.21%. In that way, the results obtained from the simulation environment are considered reliable due to the high degree of compatibility in varied cases.

B. Metrics for Evaluating Vehicle Communication

As stated before, the buses belonging to a PTS can be seen as nodes of a kind of opportunistic data network. The opportunistic nodes collectively form dynamic networks that are built from short, unpredictable contact times as nodes move in and out of connectivity. Each node participating in the routing process is responsible for bringing the packet closer to the destination. If no such neighbor exists, the message is stored at the node until it enters into the direct transmission range of the destination. These networks operate with a best-effort paradigm, and leads to long delays due to packet buffering [31].

However, in this study, we are interested in understanding the performance of the network during the period in which end-to-end communication is established, with a view mainly to real-time applications. Accordingly, a set of conventional metrics is evaluated regarding end-to-end communication. A routing protocol is required to establish transmission routes between source and destination pairs (we opted for a conventional unicast routing protocol). Each source-destination pair represents a direction in the bus route, i.e., the bus departing from the bus terminal towards the neighborhood or vice-versa. At the end of the journey, the source-destination pair terminates, and a new pair is started. Let N be the total number of source-destination pairs generated in the simulation, then we can define the following metrics:

- **Packet Delivery Rate (PDR)**: It is the average ratio between the total number of packets received by the destination (*Nr*) and the total number of packets sent by the source (*Ns*). It is defined by

$$PDR = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{Nr_i}{Ns_i} \right). \tag{1}$$

- **Average End-to-End Delay (AED)**: It is the average end-to-end delay (in seconds) experienced by data packets to propagate from origin to destination. It is defined by

$$AED = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{1}{Nr_i} \sum_{j=1}^{Nr_i} (tr_j - te_j) \right), \tag{2}$$

  where *tr* is the instant of time the destination receives the packet, *te* is the instant of time the source sends the packet and *Nr* is the total number of packets received considering a source-destination pair.
- **Throughput (THR)**: It is the average amount of data (in bps) transferred between two nodes and the time interval these nodes remain connected. It is defined by

$$THR = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{Nr_i \times packetSize}{stopTime_i - startTime_i} \times \frac{8}{1000} \right), \tag{3}$$

where *Nr* is the number of data packets successfully received, *packetSize* is the size of the data packet (in bytes), *stopTime* is the time the last (data) packet was received by the destination bus, and *startTime* is the time when the first (data) package was sent by the source bus.

- **Normalized Routing Overhead (NRO)**: It is the average ratio of the total number of routing packets sent by the origin and the total number of data packets received by the destination. It is defined by

$$NRO = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{NDs_i}{Nr_i}\right), \tag{4}$$

where *NDs* is the total number of routing packets sent, and *Nr* is the total number of data packets received by the destination.

Each considered metric reflects a different aspect of the network's functionality and quality. Network efficiency is measured through analysis obtained from PDR, THR, and AED. Besides, the performance of a routing protocol over the network is measured by NRO.

C. Simulation Configuration

For the evaluation of the network performance, simulations were performed with the *NS*-2 software package, an open source discrete event simulator [32]. For the configuration of data transmissions, a stationary bus (node) was included in each terminal (Rui Barbosa Square and Pinheirinho Terminal) and their locations did not change throughout the simulation period. Moreover, they acted as the destination in the data communication.

Network performance was also analyzed, considering only the vehicle-densest areas, i.e., around the terminals. Regions between 2 km and 4 km radius were analyzed and simulations considered the two mentioned time periods. The movements of the vehicles were the same as used previously, but no movements outside these perimeters were considered, that is, when vehicles move out of these ranges (2 km and 4 km) they leave the simulation.

The traffic pattern considered corresponds to *User Datagram Protocol* (UDP) connections, with constant bit rate (CBR - with 4 messages/s). The radio broadcast model is the *twoRay ground*, and the MAC layer follows the IEEE 802.11p standard [33]. The routing protocol is AODV [34], [35]. Simulations were carried out for 3,600 seconds (1 hour, according to SUMO mobility trace). Of all the vehicles present in this simulation, half of them were chosen as origin elements in the formation of origin-destination pairs. The origin is always chosen at random and the destination is always the stationary node (bus) at the terminal, according to the region analyzed. The radio transmission ranges used were 150, 250, 350, and 700 m [33]. All results presented are the averages of 35 simulations, and the confidence interval considered is 95%.

V. RESULTS

In this section, the results of simulations are presented with the aim of evaluating the communication performance of data exchanged between vehicles.

A. Scenario 1: Rui Barbosa Square

Figs. 8, 9, 10, and 11 show the results obtained for Packet Delivery Rate, Normalized Routing Overhead, Average End-to-End Delay and Throughput as a function of the variation of the transmission radius, considering the Rui Barbosa Square. Considering initially the morning period, Fig. 8a shows that the PDR increases as the transmission radius broadens, as expected. In this case, the PDR ranges from 7.4% to 38.5%. Fig. 9a shows the results for the afternoon, where the PDR varies between 4.3% and 35.3%.

Fig. 8b shows that the NRO for the 150 m transmission radius is the largest because, in this case, the network connectivity is low (many routing packets are sent, but few data packets are received). The NRO significantly decreases for 250 m and 350 m radius compared to the 150 m radius. However, for the 700 m radius, the NRO increases because by increasing the transmission radius, the number of routing packets sent is not proportional to the number of data packets received. When the radius goes from 350 m to 700 m, the number of routing packets sent increases about three times, while the number of packets received increases approximately two times. Similar behavior is observed in the afternoon, according to Fig. 9b.

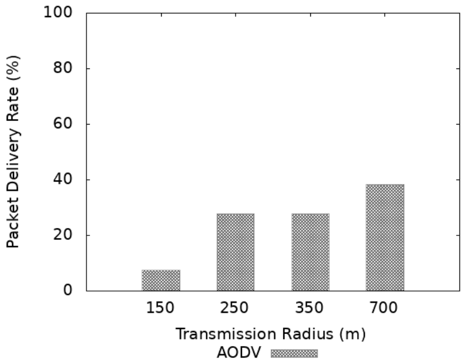
Fig. 10a shows that the 150 m transmission radius presents a higher AED due to low network connectivity and a higher number of routing packets, contributing to a higher delay. The increase in the transmission radius contributes to an improvement in delay time (with delays on the order of seconds). Similar behavior is noticed in the afternoon, as shown in Fig. 11a.

Fig. 10b shows that the THR does not change significantly for transmission radii of 250 m, 350 m, and 700 m (in the order of a few kbps), with a slight inferiority for 350 m and 700 m. For the 150 m radius, however, the THR is lower due to a lower network connectivity and a higher NRO. Similar behavior is observed in the afternoon, as shown in Fig. 11b.

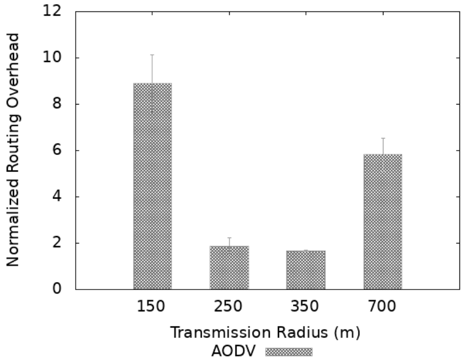
B. Scenario 2: Pinheirinho Terminal

Figs. 12, 13, 14, and 15 show the results obtained considering the Pinheirinho Terminal. Figs. 12a and 13a illustrate a low PDR, ranging from 6.5% to 36.7% (morning) and between 6.8% and 32.7% (afternoon), respectively. Fig. 12b shows the NRO, with the highest overload for the 150 m radius, while the other radii showed results very close to each other, in the range of 1.66 to 1.75. Similar behavior is observed in the afternoon, as shown in Fig. 13b.

Fig. 14a shows the AED and the radius transmission of 150 m presents a higher delay when compared to the other values, as expected, due to the low PDR presented. Fig. 14b shows that the THR remained very similar to each other in the transmission radii of 150 m, 250 m, and 350 m, with a slight superiority for 250 m. With the 700 m transmission radius, on the other hand, the THR was in the 13 kbps range. Similar behavior was obtained in the afternoon, according to Figs. 15a and 15b.

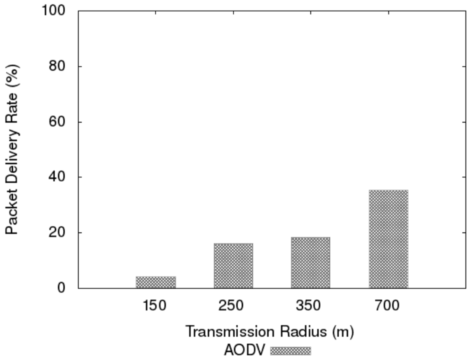


(a) Packet Delivery Rate

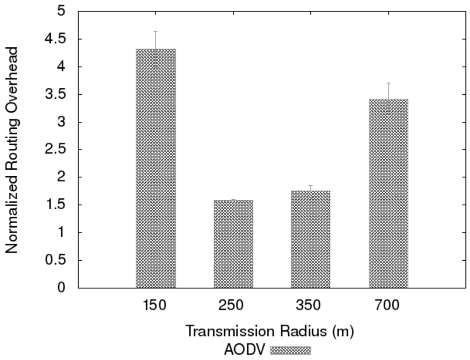


(b) Normalized Routing Overhead

Fig. 8: PDR and NRO for Rui Barbosa Square (morning).

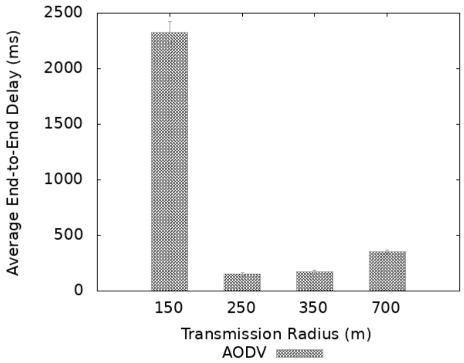


(a) Packet Delivery Rate

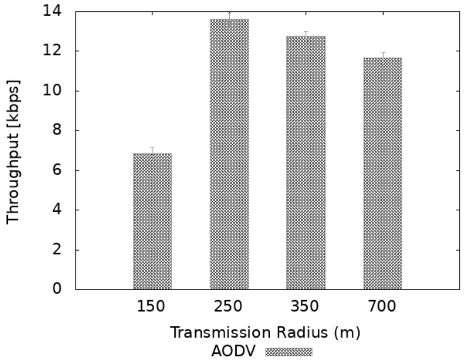


(b) Normalized Routing Overhead

Fig. 9: PDR and NRO for Rui Barbosa Square (afternoon).

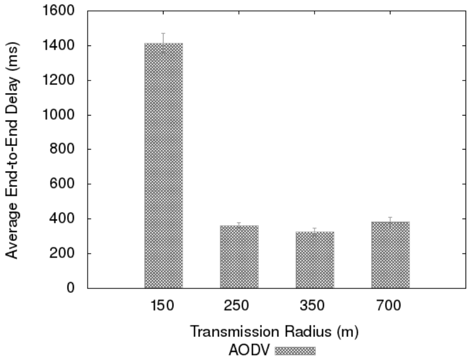


(a) Average End-to-End Delay

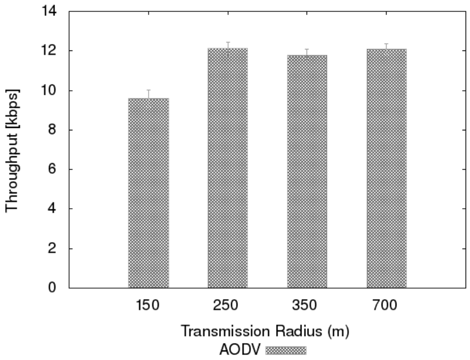


(b) Throughput

Fig. 10: AED and THR for Rui Barbosa Square (morning).

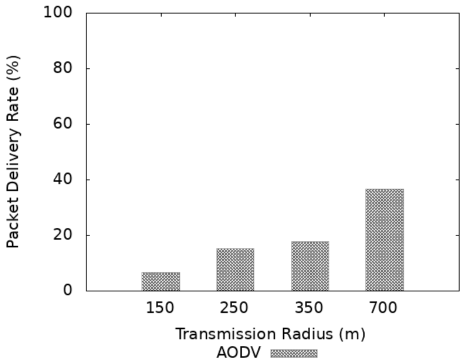


(a) Average End-to-End Delay

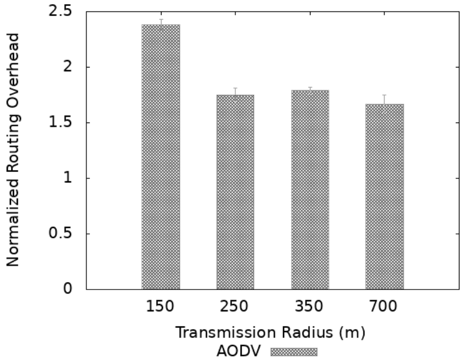


(b) Throughput

Fig. 11: AED and THR for Rui Barbosa Square (afternoon).

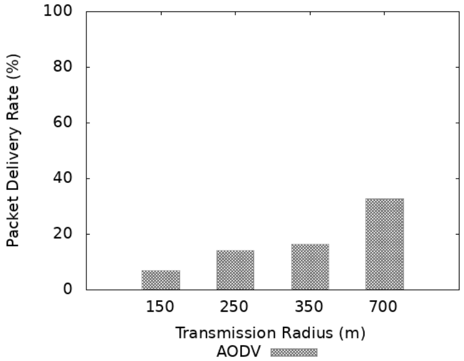


(a) Packet Delivery Rate

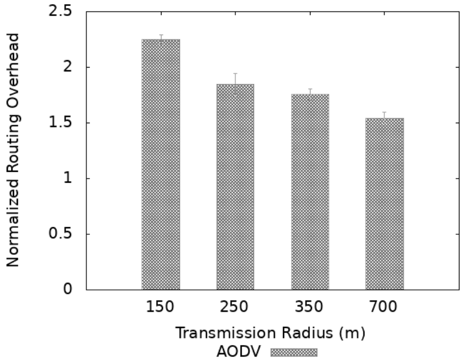


(b) Normalized Routing Overhead

Fig. 12: PDR and NRO for Pinheirinho Terminal (morning).

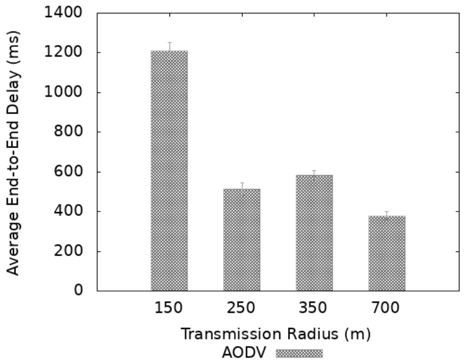


(a) Packet Delivery Rate

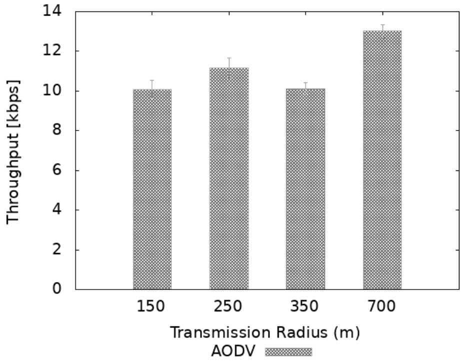


(b) Normalized Routing Overhead

Fig. 13: PDR and NRO for Pinheirinho Terminal (afternoon).

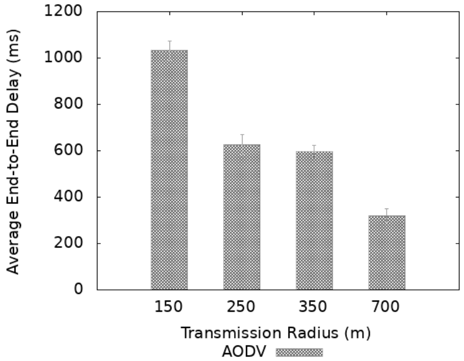


(a) Average End-to-End Delay

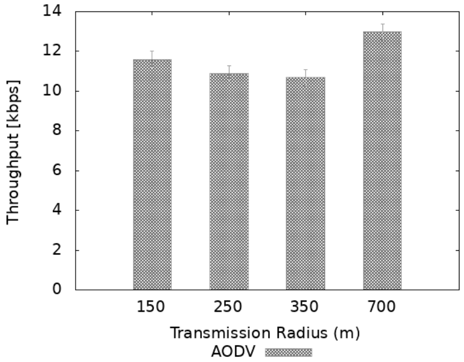


(b) Throughput

Fig. 14: AED and THR for Pinheirinho Terminal (morning).



(a) Average End-to-End Delay



(b) Throughput

Fig. 15: AED and THR for Pinheirinho Terminal (afternoon).

C. Scenario 3: Behavior Around the Terminals

The behavior of the network around the terminals, considering an area with a radius of 2 or 4 km is analyzed in this item. Because it is a denser region with a high number of favorable nodes (vehicles) for route formation, the PDR tends to offer better results compared to the other scenarios analyzed. That is illustrated in Table II, where a comparative study between the present case and the previous scenarios is presented (Figs. 10a and 14a, respectively). The PDR (in the vicinity of terminals and with a large transmission range) can be as high as 80%, favoring communication between buses and terminals.

TABLE II: PDR comparison (morning).

Transmission Radius	**Rui Barbosa** 2 km	**Rui Barbosa** 4 km
150 m	12.94% (increase 5.5%)	9.44% (increase 2.0%)
250 m	55.37% (increase 27.7%)	38.06% (increase 10.4%)
350 m	59.55% (increase 31.8%)	40.31% (increase 12.6%)
700 m	88.07% (increase 49.6%)	57.61% (increase 19.1%)
Transmission Radius	**Pinheirinho** 2 km	**Pinheirinho** 4 km
150 m	13.32% (increase 6.7%)	8.84% (increase 2.2%)
250 m	31.72% (increase 16.3%)	20.90% (increase 5.5%)
350 m	36.97% (increase 19.2%)	23.97% (increase 6.2%)
700 m	77.40% (increase 40.7%)	50.95% (increase 14.2%)

VI. CONCLUSION

This work proposed a methodology for modeling and analyzing bus-based networks. Making use of EMMS, this work conducted an exploratory study involving bus lines belonging to the PTS of Curitiba, based in the central region (Rui Barbosa Square) and in the southern region of the city (Pinheirinho Terminal). Comparison between simulated and actual data showed that EMMS was efficient in the process of map-matching. The high degree of compatibility in varied cases ensures that the results obtained from the simulation environment are reliable.

The network simulations performed in *NS*-2 showed that the formation of small temporary networks is possible, allowing data exchange between buses. However, the packet delivery rate presented low values (on the order of 10 to 30%). The throughput found is quite low, about 10 to 12 kbps. The average delay presented high values, on the order of 0.5 to 1.0 s. Finally, the routing overhead showed high values because many routing packets were sent, and a few data packets were received. These results are valid for both city regions (central and southern) and in both periods evaluated (morning and afternoon). All these results are explained by the high time taken to re-establish communications after a link disruption, a characteristic of bus-based networks.

Given the dynamic nature of the system studied in this paper, it is possible to state that the potential types of communication services allowed in such a network would be classified as opportunistic and delay-tolerant. Accordingly, as subject of future work, our model can be further considered to analyze the performance of specific routing protocols for opportunistic and delay-tolerant networks.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. We also thank one of the reviewer for suggesting the inclusion of the comparative table (Tab. 1) and the EMMS flowchart (Fig. 1).

REFERENCES

[1] A. Founoun and A. Hayar, “Evaluation of the concept of the smart city through local regulation and the importance of local initiative,” in *2018 IEEE International Smart Cities Conference (ISC2)*. Kansas City, Missouri, USA: IEEE Smart Cities, 2018. doi: 10.1109/ISC2.2018.8656933 pp. 1–6.

[2] D. Zandbergen, “”We Are Sensemakers”: The (Anti-)politics of Smart City Co-creation,” in *Public Culture, Durham*, vol. 29, no. 3, Sept. 2017. doi: 10.1215/08992363-3869596 pp. 539–562.

[3] J. Andersen and S. Sutcliffe, “Intelligent Transport Systems (ITS) - An Overview,” in *IFAC Proceedings Volumes*, vol. 33, no. 18, 2000. doi: 10.1016/S1474-6670(17)37129-X. ISSN 1474-6670 pp. 99–106, IFAC Conference on Technology Transfer in Developing Countries: Automation in Infrastructure Creation (DECOM-TT 2000), Pretoria, South Africa, 5-7 July 2000.

[4] J. Alves Junior and E. C. G. Wille, “Increasing connectivity in VANETs using public transport backbones,” in *IEEE Latin America Transactions*, vol. 13, no. 10, Oct 2015. doi: 10.1109/TLA.2015.7387250 pp. 3421–3431.

[5] H. Maowad and E. Shaaban, “Efficient routing protocol for vehicular ad hoc networks,” in *1st International Conference Pervasive Computing Signal Processing and Applications (PCSPA)*, 2011. doi: 10.1109/ICNSC.2012.6204918

[6] A. Mahajan, N. Potnis, K. Gopalan, and A. Wang, “Urban mobility models for VANET,” in *2nd Workshop on Next Generation Wireless Networks*, 2007.

[7] J. Alves Junior and E. C. G. Wille, “P-AOMDV: An improved routing protocol for V2V communication based on public transport backbones,” in *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 12, 2016. doi: 10.1002/ett.3104 pp. 1653–1663.

[8] D. R. de Assis, J. Alves Junior, and E. C. G. Wille, “Strategies to increase connectivity and performance of VANETs: Analysis of route lifetime and routing protocol proposal,” in *Transactions on Emerging Telecommunications Technologies*, vol. 34, no. 1, 2023. doi: https://doi.org/10.1002/ett.4665 p. e4665.

[9] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, “Microscopic traffic simulation using SUMO,” in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. doi: 10.1109/ITSC.2018.8569938

[10] J. Xie, Y. Marco Nie, and X. Liu, “Testing the proportionality condition with taxi trajectory data,” in *Transportation Research Part B: Methodological*, vol. 104, no. C, 2017. doi: 10.1016/j.trb.2017.05.003 pp. 583–601.

[11] S. Ahmed and S. S. Kanere, “SKVR: scalable knowledge-based routing architecture for public transport networks,” in *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, ACM*, 2006. doi: 10.1145/1161064.1161082 pp. 92–93.

[12] M. Kezia and K. V. Anusuya, “Mobility models for internet of vehicles: A survey,” in *Wireless Personal Communications*, vol. 125, no. 2, 2022. doi: 10.1007/s11277-022-09637-7 p. 1857–1881.

[13] M. Fiore, J. Harri, F. Filali, and C. Bonnet, “Understanding vehicular mobility in network simulation,” in *IEEE International Conference on Mobile Adhoc and Sensor Systems*, 2007. doi: 10.1109/MOBHOC.2007.4428738 pp. 1–6.

[14] S. Batabyal and P. Bhaumik, “Mobility models, traces and impact of mobility on opportunistic routing algorithms: A survey,” in *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, 2015. doi: 10.1109/COMST.2015.2419819 pp. 1679–1707.

[15] C. Kumar, B. Bhushan, and S. Gupta, “Evaluation of MANET Performance in Presence of Obstacles,” in *International Journal of Ad hoc, Sensor & Ubiquitous Computing (IJASUC)*, vol. 3, no. 3, 06 2012. doi: 10.5121/ijasuc.2012.3303 pp. 37–46.

[16] *OpenStreetMap*, OpenStreetMap, 12 2022. [Online]. Available: http://www.openstreetmap.org

[17] M. Quddus, W. Ochieng, and R. Noland, "Current map-matching algorithms for transport applications: state-of-the art and future research directions," in *Transportation Research Part C: Emerging Technologies, 15 (5), pp. 312-328.*, 2007. doi: 10.1016/j.trc.2007.05.002

[18] F. C. Pereira, H. Costa, and N. M. Pereira, "An off-line map-matching algorithm for incomplete map databases," in *European Conference of Transport Research Institutes (ECTRI)*, 2009. doi: 10.1007/s12544-009-0013-6

[19] M. Hashemi and H. A. Karimi, "A critical review of real-time map-matching algorithms: current issues and future directions," in *Computers, Environment and Urban Systems 48: 153-165*, 2014. doi: 10.1016/j.compenvurbsys.2014.07.009

[20] S. Gaito, D. Maggiorini, E. Pagani, and G. P. Rossi, "Distance vector routing for public transportation vehicular networks: Performance evaluation on a real topology," in *IEEE XPLORE*, 2010. doi: 10.5555/1834116.1834178

[21] T. Abdeldjalil, F. Li, R. Li, and X. Li, "Table-driven bus-based routing protocol for urban vehicular ad hoc networks," in *Wireless Algorithms, Systems, and Applications*, Z. Cai, C. Wang, S. Cheng, H. Wang, and H. Gao, Eds. Cham: Springer International Publishing, 2014. ISBN 978-3-319-07782-6 pp. 90–101.

[22] L. Bedogni, M. Gramaglia, A. Vesco, M. Fiore, J. Härri, and F. Ferrero, "The Bologna ringway dataset: Improving road network conversion in SUMO and validating urban mobility via navigation services," in *IEEE Trans. Veh. Technol.*, vol. 64, no. 12, 2015. doi: 10.1109/TVT.2015.2475608 pp. 5464–5476.

[23] F. Zhang, B. Jin, Z. Wang, H. Liu, J. Hu, and L. Zhang, "On geocasting over urban bus-based networks by mining trajectories," in *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 6, 2016. doi: 10.1109/TITS.2015.2504513 pp. 1734–1747.

[24] M. Rios, "GeOpps-N: Opportunistic routing for VANET in a public transit system," in *IEEE Latin America Transactions*, vol. 14, no. 4, 2016. doi: 10.1109/TLA.2016.7483493 pp. 1630–1637.

[25] J. Alves Junior and E. C. G. Wille, "Exploiting the inherent connectivity of urban mobile backbones using the P-DSDV routing protocol," in *Advances in Electrical and Computer Engineering*, vol. 20, no. 1, 2020. doi: 10.4316/AECE.2020.01011 pp. 83–90.

[26] N. Chaib, O. S. Oubbati, M. L. Bensaad, A. Lakas, P. Lorenz, and A. Jamalipour, "BRT: Bus-based routing technique in urban vehicular networks," in *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, 2020. doi: 10.1109/TITS.2019.2938871 pp. 4550–4562.

[27] S. Rashidibajgan and T. Hupperich, "Improving the performance of opportunistic networks in real-world applications using machine learning techniques," in *Journal of Sensor and Actuator Networks*, vol. 11, no. 4, 2022. doi: 10.3390/jsan11040061. ISSN 2224-2708

[28] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE Simulator for DTN Protocol Evaluation," in *2nd International Conference on Simulation Tools and Techniques*. New York, NY, USA: ICST, 2009. ISBN 978-963-9799-45-5

[29] "Python," Python Software Foundation, 12 2022. [Online]. Available: https://www.python.org/

[30] *URBS – Integrated Network Public Data*, Urbanização de Curitiba S/A, Curitiba, Paraná, Jan. 2022. [Online]. Available: https://www.curitiba.pr.gov.br/dadosabertos/

[31] R.-I. Ciobanu, C. Dobre, and V. Cristea, "A data dissemination algorithm for opportunistic networks," in *13th International Symposium on Symbolic and Numeric Algorithms for Scientific Compuitng*, 2011. doi: 10.1109/SYNASC.2011.56 pp. 299–305.

[32] "The network simulator," NS-2, 12 2022. [Online]. Available: https://www.isi.edu/nsnam/ns/

[33] T. T. Almeida, L. de C. Gomes, F. M. Ortiz, J. G. R. Junior, and L. H. M. K. Costa, "IEEE 802.11p performance evaluation: Simulations vs. real experiments," in *21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018. doi: 10.1109/ITSC.2018.8569676 pp. 3840–3845.

[34] E. E. Akkari, G. Dos Santos, M. Alves, and M. M. Santos, "Performance analysis and comparison of the DSDV, AODV and OLSR routing protocols under VANETs," in *16th International Conference on Intelligent Transportation Systems Telecommunications (ITST).*, 2018. doi: 10.1109/ITST.2018.8566825

[35] B. V. Coutinho, E. C. G. Wille, and H. I. Del Monego, "Performance of routing protocols for VANETs: A realistic analysis format," in *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, ser. IMCOM'15, 2015. doi: 10.1145/2701126.2701141

Eduardo Renan Manika received the B.S.E.E degree in Telecommunications Systems Technology and M.Sc. degree in Electronic Engineering from Federal University of Technology of Paraná (UTFPR), Curitiba, Brazil in 2009 and 2022, respectively. He is currently at UTFPR, Curitiba, Brazil, as IT solutions architect. His research interests include wireless systems and performance evaluation in communication networks.

Emilio Carlos Gomes Wille was born in Lapa (PR) - Brazil. He received his degree in Electrical Engineering in February 1989, and a M.Sc. in Electrical Engineering in July 1991, both from Universidade Tecnológica Federal do Paraná - UTFPR (Curitiba - Brazil). He received his Ph.D. degree in Electronic and Telecommunications Engineering from Politecnico di Torino (Italy) in February 2004. During his stay in Italy he was supported by a CAPES Foundation scholarship from the Ministry of Education of Brazil. He is a Full Professor at the UTFPR, and since October 1991 he is with the Electronics Department. His teaching duties at UTFPR comprise graduate and undergraduate-level courses on electronic and telecommunication theory. He has co-authored over 100 papers published in international journals and presented in leading conferences, all of them in the area of communication networks. He served as the head of the Graduate Program in Electrical and Computer Engineering (CPGEI) at the UTFPR from 2014 to 2016. His research interests include communication networks, wireless systems, stochastic processes, computer simulation and optimization algorithms.

Joilson Alves Junior received the B.S.E.E degree in Informatics Technology from Federal University of Technology of Paraná (UTFPR), Curitiba, Brazil, in 2005, and M.Sc. degree in Informatics from Federal University of Paraná (UFPR), Curitiba, Brazil, in 2012. He received his Ph.D. degree in Electronic Engineering from UTFPR (Curitiba) in 2016. Since 2018 he is with the Electronics Department at UTFPR - Curitiba. His research interests include wireless systems and performance evaluation in communication networks.