Journal of Internet Services and Applications, 2024, 15:1, doi: 10.5753/jisa.2024.3812

Towards spatiotemporal integration of bus transit with data-driven approaches

Júlio C. Borges [**Universidade Tecnológica Federal do Paraná** | ***julio.2018@alunos.utfpr.edu.br***]
Altieris M. Peixoto [**Universidade Tecnológica Federal do Paraná** | ***altieris.marcelino@gmail.com***]
Thiago H. Silva [**Universidade Tecnológica Federal do Paraná** | ***thiagoh@utfpr.edu.br***]
Anelise Munaretto [**Universidade Tecnológica Federal do Paraná** | ***anelise@utfpr.edu.br***]
Ricardo Lüders [**Universidade Tecnológica Federal do Paraná** | ***luders@utfpr.edu.br***]

CPGEI, Universidade Tecnológica Federal do Paraná (UTFPR), Av. Sete de Setembro, 3165, Rebouças, Curitiba, PR, 80230-901, Brazil.

Received: 01 November 2023 • **Accepted:** 24 April 2024 • **Published:** 05 June 2024

Abstract This study aims to propose an approach for spatiotemporal integration of bus transit, which enables users to change bus lines by paying a single fare. This could increase bus transit efficiency and, consequently, help to make this mode of transit more attractive. Usually, this strategy is allowed for a few hours in a non-restricted area; thus, certain walking distance areas behave like "virtual terminals." For that, two data-driven algorithms are proposed in this work. First, a new algorithm for detecting itineraries based on bus GPS data and the bus stop location. The proposed algorithm's results show that 90% of the database detected valid itineraries by excluding invalid markings and adding times at missing bus stops through temporal interpolation. Second, this study proposes a bus stop clustering algorithm to define suitable areas for these virtual terminals where it would be possible to make bus transfers outside the physical terminals. Using real-world origin-destination trips, the bus network, including clusters, can reduce traveled distances by up to 50%, making twice as many connections on average.

Keywords: bus transit network, spatiotemporal integration, data-driven model, urban computing, smart city

1 Introduction

The collective public transit service has been little expanded and modernized over the last few decades, compared to private individual transit, which currently consumes the largest available urban road space. Motta *et al*. [2013] show historical conflicts and contradictions at the origin of the problems affecting collective public transit systems throughout Brazil.

Aiming to revert this, attract new users, and improve the quality of the service, several companies that manage collective public transit and researchers in this area are using vehicle GPS data more frequently by providing an efficient and accurate way to track the position of vehicles at a given time [Bona *et al*., 2016; Curzel *et al*., 2019; Santin *et al*., 2020; Gubert *et al*., 2023]. In this way, public transit can be investigated from different perspectives, examining the system's dynamic behavior, measuring the efficiency of services, identifying movement patterns, integration capacity, peak hours, etc.

In analyzing the operation of a public transit network, it is generally necessary to identify the instants of time in which the bus passes at the stops, whether or not the bus stops. In general, this identification is done by a map-matching algorithm crossing the geolocation information of the bus with the location of the bus stop. A suitable method for this task was employed in the work of Martins *et al*. [2022]. The method presents a solution for detecting bus stops, even on two-way roads, correcting GPS inaccuracies, and identifying the exact time of passage at a bus stop.

However, the practical application of working with real GPS data can pose significant challenges due to intermittent communication failures, leading to data gaps that may adversely impact analysis outcomes if not appropriately managed. Another problem is the need to cross-reference scheduled timetables with GPS information, as outdated timetables generate inconsistencies, such as the cases reported by Martins *et al*. [2022]. Thus, researchers are often forced to look for periods in which there is consistency in logs and tables. Nevertheless, this limitation confines the analysis to specific time intervals and conditions during which the data is optimal.

The itinerary detection problem, as defined in this article, goes further, as it consists of matching the GPS logs of the movement of a bus with its respective itinerary and schedule, which is defined in the bus operation planning. This problem is affected by communication failures in sending GPS data, making it impossible to detect the bus itinerary correctly. An approach for detecting itineraries was employed in the work of Peixoto *et al*. [2020] using the schedule table for a bus line. In this case, both the starting point departure time and the scheduled arrival time at the endpoint must be known. In addition, logs of bus movements are needed. Although this algorithm identifies most itineraries, several logs from bus monitoring are discarded, as they do not appear associated with any itinerary. Unlike the previous work, the algorithm proposed in this article does not use the scheduled bus line schedule.

In our previous work [Borges *et al*., 2023], an algorithm was introduced capable of detecting the itinerary of a bus in operation using data from its geolocation without the need to use information from the scheduled timetables of each bus line. Furthermore, the proposed algorithm adds missing data due to communication failures by interpolating known time values.

In this context, this article aims to develop a spatiotemporal integration of bus transit using data-driven strategies. There is no single definition for public transit service integration as it depends on the context and the transit modes examined. This work focuses on the integration of the public bus transit, which can be understood as a strategy to increase the network connectivity, offering users greater route possibilities without the need to create new bus stops [Wang *et al.*, 2021].

The main contributions of this paper are as follows:

- Bus itinerary detection algorithm for building itineraries with spatiotemporal information based on GPS bus monitoring;
- Analysis of the impact of the interpolation of GPS data on the different types of lines of the transit system;
- Full database assessment using real GPS bus data and temporal evaluation of bus service;
- Bus stop clustering algorithm for grouping nearby stops where connections can be made between bus lines with a single fare;
- Evaluation of the bus stop clustering algorithm and the bus service synchronization;
- Evaluation of the spatiotemporal integration impact on bus trips in terms of distance traveled and number of transfers between bus lines using real-world origin-destination trips.

The article's structure is as follows: In **Section 2**, an overview of related works is presented. **Section 3** provides a detailed explanation of the proposed algorithms. The outcomes of these algorithms are discussed in **Section 4**, followed by the conclusion in **Section 5**.

2 Related Works

2.1 Public transit Mobility Understanding

Detailed information about public transit systems, including bus stops, stations, terminals, routes, and timetables, are widely deployed in standard data formats, such as the General Transit Feed Specification (GTFS). In recent decades, new technologies have improved tools for monitoring and verifying schedules and other performance metrics, such as Automatic Vehicle Location (AVL) and other automated data collection to traffic [Wilson *et al.*, 2009]. In particular, GPS equipment allows a wide range of applications with the potential to improve service and efficiency. The studies of Sridevi *et al.* [2017], Hakeem *et al.* [2022], and Desai *et al.* [2022] exemplify recent applications in which onboard equipment collects GPS trajectories of buses and centralizes them in a server. These data can be used in various transit system management applications.

A literature review on the bus trajectory data application can be seen in War *et al.* [2022]. Aspects such as data sources and methods of Big Data and IoT in mass public transit are described in Welch and Widita [2019]. Further discussion of using bus GPS trajectory data is provided in Singla and Bhatia [2015].

Concerning the Integrated transit Network (ITN), several studies such as Bona *et al.* [2016]; Curzel *et al.* [2019]; Santin *et al.* [2020]; Rosa *et al.* [2020]; Gubert *et al.* [2023], used open public transit data, such as vehicle GPS trajectory data, timetables, and itineraries, in the creation of models that allowed the expansion of the understanding of the characteristics and behaviors of the transit network, to promote the improvement of the efficiency of the service. Other studies on urban mobility, transit networks, and computational models that use similar data are performed by Rodrigues *et al.* [2017]; Wehmuth *et al.* [2018]; Maduako *et al.* [2019]; Sadeghian *et al.* [2021]; Li and Rong [2022]. These models provide many efficiency measures for transit services, helping to identify opportunities for important improvements such as cost reduction.

2.2 Public Transit Data Quality

Another important issue is the data quality. Cleaning the raw data must reduce inconsistencies for the models to work properly. For example, Martins *et al.* [2022] pointed out several issues present in real GPS data, and therefore, they developed a model of map matching. This model can be used for: i) vehicle stop detection between nearby bus stops on a two-way street, in which one point serves both directions ("going" and "returning") from the bus line; ii) GPS inaccuracies; and iii) the vehicle's exact time of passage at a bus stop. The problem of detecting bus stops was also addressed in the work of Peixoto *et al.* [2020], where the vehicle's itinerary was also employed using data from timetables. Other authors also faced the challenge of detecting compatibility between the trajectory of buses and their respective itineraries, such as Yin *et al.* [2014]; Queiroz *et al.* [2019]; Chawuthai *et al.* [2023]. In these works, the main objective of detection is to identify whether or not a bus GPS trajectory is in accordance with the planned itinerary to signal any inconsistency. In the work of Gallotti and Barthelemy [2015], inconsistencies in vehicle stop times were corrected using a temporal interpolation method, but no measure of interpolation error was presented.

The algorithm proposed in the present article fills some gaps, identifying and correcting inconsistencies in GPS trajectories according to the itinerary. Furthermore, a method to measure the interpolation error is presented. Therefore, the proposed approach treats the itinerary detection problem and the temporal interpolation by reducing the inconsistencies in the raw data and providing a reliable database for new applications.

2.3 Public Transit Integration

Coordinating several transit routes, schedules, and vehicles while looking to attend to passenger preferences is a complex challenge [Arriagada *et al.*, 2022]. Although bus services have previously defined timetables, current trips often breach schedules due to uncertainties such as congestion, traffic lights, poor weather conditions, boarding and alighting times, and maintenance interruptions [Li *et al.*, 2021; Kumar and Khani, 2023]. Public transit users often avoid long waits, extensive walking distances, or route changes. However, transfers are seen as a strategy to deal with overcrowding in vehicles and bus stations [Pan *et al.*, 2023]. In addition, transfers are necessary to integrate routes without building new stops [Wang *et al.*, 2021] while reducing the total operational cost [Liu *et al.*, 2023]. The distance required to walk to a bus stop and the time spent are important aspects for building more accessible public transit systems [Żochowska *et al.*, 2022]. In this direction, Noichan and Dewancker [2018] employed spatial analysis to assess route connectivity, examining the distances between transit points as a crucial factor. A better synchronization between timetables and vehicle schedules may reduce the wait time [Prathyusha *et al.*, 2021; Kumar and Khani, 2023].

In general, works that present approaches for integrating transit networks focus on route optimization problems. They propose to combine different routes in a transit network to complete a trip. These approaches usually employ optimization methods such as mathematical programming, heuristics, or simulation. In this case, transfer points are previously known or identified by static data such as vehicles' scheduled departure and arrival times provided in the timetables. However, known transfer points or timetable approaches may not be the best option for bus networks because of the uncertainties in travel times that are introduced by several of the previously mentioned factors. Therefore, many works employ AVL data from online bus travel monitoring systems because it offers a more accurate way of evaluating transit networks [Li *et al.*, 2021; Kumar and Khani, 2023].

A summary of related works for public transit integration is presented in Table 1. It presents relevant approaches for bus or multimodal networks using timetables or AVL data and considering temporal, spatial, or spatiotemporal dimensions of the integration. The results are generated and reported using data-driven, optimization, or simulation methodologies. "Virtual Terminal" refers to the ability to identify bus line integration regions as described below.

Our approach of identifying transfer points between bus lines relates to AVL-based (online GPS information of vehicles) and data-driven approaches. However, it is different from previous works because it identifies suitable transfer points using both spatial (bus stop location) and temporal (online frequency of buses) information. Regions of interest are identified using a measure of maximum walking distance and "synchronization" between bus lines at a particular bus stop. They can be used as "virtual terminals" for bus line integration. Moreover, it is general enough to be adapted to other GTFS and AVL data systems and can identify opportunities to integrate bus lines based on historical data.

3 Spatiotemporal Integration of Bus Transit

A spatiotemporal integration of bus transit is a strategy that allows users to change bus lines by paying a single fare. Usually, this strategy is allowed for a few hours in a non-restricted area (terminals are restricted areas, for instance). Certain walking distance areas function similarly to "virtual terminals". Two goals must be considered: i) detection of bus itineraries and ii) bus stop clustering to define suitable areas for these "virtual terminals." Both objectives are achieved by the two data-driven algorithms proposed in this section.

The central issue is to develop an itinerary detection algorithm independent of the bus schedule table. Initially, it is necessary to differentiate the concepts of "static network" and "dynamic network." These concepts were also used in the work of Peixoto *et al.* [2020]. A static network represents the topology of the transit network, that is, the sequencing of bus stops on a specific line covering all itineraries offered by the service, as considered in Bona *et al.* [2016].

Since the static network describes the topology of the bus line and its respective itineraries without including timetables, the dynamic network is formed as a given vehicle reaches the points provided for in its service itinerary. The proposed itinerary detection algorithm is composed of 3 steps:

- **step 1**: mark the time a bus passes at bus stops (map matching algorithm).
- **step 2**: sequence bus stops according to these time marks (temporal sequencing).
- **step 3**: associate a temporal sequence of bus stops to a known itinerary, interpolating and removing marks if necessary (the proposed algorithm).

A map-matching algorithm is used at **step 1**. The algorithm used in this work is based on Martins *et al.* [2022] algorithm. It calculates the Haversine distance from each vehicle position (log) as used in Panigrahi [2014]; Lawhead [2015] to all bus line stops and assigns the log to the closest stop. This way, it is possible to mark the time of passage of a vehicle to each bus line stop. **Step 2** sorts time marks in ascending order to obtain a temporal sequencing of points. **Step 3** is accomplished by the algorithm proposed below.

The itinerary detection algorithm aims to associate a log of events captured by the movement of a specific bus to the sequence of points of their respective bus line registered in the table `LinePoints`. Thus, it is possible to associate the instant time of passage of the bus at all points on the line. This is illustrated in **Figure 1**, where a log of events $log = ((l_1, t_1), (l_2, t_2), (l_3, t_3), (l_4, t_4), (l_5, t_5))$ will be associated with a itinerary $iti = (p_1, p_2, p_3, p_4, p_5)$, where t_i is the moment when the bus passes at coordinate l_i and p_i is a point on the bus itinerary.

For example, according to **Figure 1**, there is no record of a bus passing at point p_3, and there is a record of a bus passing at a position l_4 that does not correspond to any registered point on the line.

The map matching algorithm associates the locations l_i to the respective bus stops by evaluating a measure of spatial proximity between l_i and a point on the itinerary. In the case of **Figure 1**, the result of map matching is the mapping $map = ((p_1, l_1), (p_2, l_2), (p_4, l_3), (-, l_4), (p_5, l_5))$, and no point is associated with location l_4 and there is no record of the passage through point p_3.

Table 1. Related works for public transit integration.

Authors	Network	Timetable-Based	AVL-Based	Virtual Terminal	Dimension	Approach
Our work	Bus	No	Yes	Yes	Spatiotemporal	Data-Driven
Wang *et al.* [2021]	Bus	No	Yes	No	Spatiotemporal	Optimization
Kumar and Khani [2023]	Multimodal	Yes	Yes	No	Temporal	Optimization
Li *et al.* [2021]	Multimodal	Yes	Yes	No	Spatiotemporal	Simulation
Żochowska *et al.* [2022]	Bus	Yes	Yes	No	Spatiotemporal	Data-Driven
Prathyusha *et al.* [2021]	Single bus route	Yes	No	No	Temporal	Optimization
Steiner and Irnich [2020]	Multimodal	Yes	No	No	Spatiotemporal	Optimization
Noichan and Dewancker [2018]	Multimodal	Yes	No	No	Spatial	Data-Driven
Liu *et al.* [2023]	Bus	Yes	No	No	Temporal	Optimization

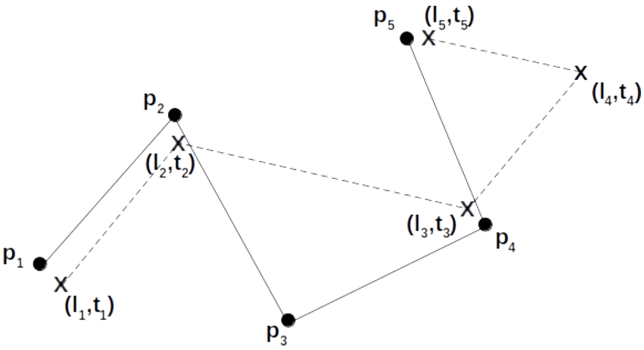


Figure 1. Example of an itinerary $iti = (p_1, p_2, p_3, p_4, p_5)$ in full line and a $log = ((l_1, t_1), (l_2, t_2), (l_3, t_3), (l_4, t_4), (l_5, t_5))$ in dashed line.

Then, the proposed algorithm detects the itinerary with temporal information $det = ((p_1, t_1), (p_2, t_2), (p_3, \hat{t}_3), (p_4, t_3), (p_5, t_5))$, associating the moment of passage to each point of the bus line. In this case, the time instant $\hat{t}_3 = t_2 + (t_3 - t_2)/2$ is estimated by averaging the times t_2 and t_3 of points adjacent to p_3.

The above result for itinerary detection $det = \{(p_i, t_j)\}$ with temporal information can be generalized to (1) for iti of dimension n and log of dimension m.

$$(p_i, t_i) = \begin{cases} (p_i, t_j) & : (p_i, l_j) \in map; i = 1 \text{ to } n; j = 1 \text{ to } m \\ (p_i, \hat{t}_i) & \text{otherwise} \end{cases} \quad (1)$$

where $\hat{t}_i = \hat{t}_{i-1} + (t_{k+w} - t_k)/w$ for $k < i < (k + w)$ and $\hat{t}_k = t_k$, considering the $w - 1$ bus stops that were not mapped by map matching between points (p_k, t_k) and $(p_{k+w}, t_{k+w}) \in map$. The above procedure is summarized in **Algorithm 1**, whose main variables are described in Table 2.

Table 2. List of variables for Algorithm 1.

Sets	
iti	Bus stops $\{p_i\}$ ordered according to the bus line route
log	Time-ordered GPS bus logs $\{(l_j, t_j)\}$ for a single trip
map	GPS log assigned to a bus stop $\{(p_i, l_j, t_j)\}$
det	Time-assigned bus stops $\{(p_i, t_i)\}$ for a single trip
Indexes	
i	Bus stop in a bus line route, $1 \le i \le n$
j	Log number in a GPS log list, $1 \le j \le m$
k	Start of a sequence of $(w - 1)$ bus stops without time log
Variables	
$(w - 1)$	Number of bus stops without time log
Δt	Interval time elapsed between logs of stops p_k and p_{k+w}
$\hat{t}_i$	Time estimate for stop p_i using temporal interpolation

Given iti as a set of bus stops ordered according to the bus route and log as a time-ordered set of GPS bus logs for a single trip, **Algorithm 1** searches for bus stops in the log. For each bus stop p_i (line 3), a set map is built with corresponding locations l_j found in the log (line 7). The corresponding $time$ is updated (line 9) to ensure an increasing time sequence in the map. If a bus stop is not found, a pair $(None, None)$ is included in the map for further processing (line 13).

After the map is built, a new set det is generated by assigning each bus stop to its corresponding log time (line 19). If a bus stop is not previously identified with a corresponding log (line 13), then a time estimate $\hat{t}_i$ should be computed (line 22) according to (1) and included in det (line 23). Time estimates are computed for $(w-1)$ missing stops with a time interval Δt elapsed from stops p_k to $p_{(k+w)}$.

Algorithm 1 Itinerary detection

Input: $iti = \{p_i\}, 1 \le i \le n; log = \{(l_j, t_j)\}, 1 \le j \le m$ // ordered by points and time, respectively
Output: $det = \{(p_i, t_i)\}, 1 \le i \le n$
1: $map \leftarrow \{\}$
2: $time \leftarrow -1$
3: **for each** $p_i \in iti$ **do**
4: $found \leftarrow False$
5: **for each** $(l_j, t_j) \in log$ **do**
6: **if** $(p_i = l_j)$ and $(t_j > time)$ **then**
7: $map \leftarrow map \cup \{(p_i, l_j, t_j)\}$
8: $found \leftarrow True$
9: $time \leftarrow t_j$
10: **end if**
11: **end for**
12: **if** $(found = False)$ **then**
13: $map \leftarrow map \cup \{(p_i, None, None)\}$
14: **end if**
15: **end for**
16: $det \leftarrow \{\}$
17: **for each** $(p_i, l_i, t_i) \in map$ **do**
18: **if** $(l_i \neq None)$ **then**
19: $det \leftarrow det \cup \{(p_i, t_i)\}$
20: **else**
21: computes w, $\Delta t = (t_{k+w} - t_k)$ and $\hat{t}_k = t_k$ // missing points between p_k and p_{k+w} acquaintances
22: $\hat{t}_i = \hat{t}_{i-1} + \Delta t/w$
23: $det \leftarrow det \cup \{(p_i, \hat{t}_i)\}$
24: **end if**
25: **end for**

The proposed algorithm has as requirements the static network and the dynamic network. That is, it is necessary to provide as input both the structure (or topology) of the transit network, according to the file `PontosLinha`, as well as the *logs* of GPS of buses from file `Vehicles`. The advantage over the method proposed by Peixoto *et al.* [2020] is that there is no need for a table of scheduled bus lines (files `TabelaLinhas` and `TabelaVeículos`).

Given a set of markings with known times (p_i, t_i) and estimated times $(p_i, \hat{t}_i)$, temporal interpolation introduces an estimation error given by $err_i = |t_i - \hat{t}_i|$. This work takes known values from the original data set to obtain the estimation error evaluation results.

The second algorithm deals with bus stop clustering that behaves like virtual terminals (not constrained to a particular area), where users can change between bus lines with a single fare. They are defined when nearby bus stops are clustered within a 600 meter radius, which is considered a walking distance suitable for changing bus lines [Peixoto *et al.*, 2020].

Table 3. List of variables for Algorithm 2

Sets	
$candidates$	Bus stops $\{c_i\}$ candidate to cluster centroids sorted in ascending order by the average number of buses
bus_stops	All stops $\{b_i\}$ of the bus network
$cluster$	Bus stops of a cluster
$clusters$	Set of clusters
Indexes	
i	Bus stop in a list of centroid candidates, $1 \leq i \leq n$
j	Bus stop in a list of all bus stops, $1 \leq j \leq m$

Centroids and clusters are computed according to **Algorithm 2**, whose main variables are described in Table 3. It starts with a descending-order list of candidate centroids ordered by the average number of buses. For each centroid in the head of the list (line 3), all neighbor bus stops within a given distance from the centroid are included in the cluster (line 7). The new cluster is then added to the set of clusters (line 10), and all bus stops of the cluster are removed from the candidates list. It means that clustered bus stops are no longer candidates for another cluster. The algorithm ends when the list of candidates is empty (line 12).

Algorithm 2 Bus stop clustering

Input: $candidates = \{c_i\}, 1 \leq i \leq n$; // ordered list of n centroid candidates
$bus\_stops = \{b_j\}, 1 \leq j \leq m$ // list of bus stops
Output: $clusters = \{\{cluster\}_i\}, 1 \leq i \leq l$ // set of l clusters
1: $clusters \leftarrow \{\}$
2: **while** $candidates \neq \{\}$ **do**
3: $centroid \leftarrow head(candidates)$ // first element of the ordered list
4: $cluster \leftarrow \{centroid\}$
5: **for each** $b_j \in bus\_stops$ **do**
6: **if** $haversine(centroid, b_j) \leq 600$ **then** // Haversine distance
7: $cluster \leftarrow cluster \cup \{b_j\}$
8: **end if**
9: **end for**
10: $clusters \leftarrow clusters \cup \{\{cluster\}\}$
11: $candidates \leftarrow candidates - (candidates \cap cluster)$ // remove the clustered bus stops from the candidates list
12: **end while**

4 Results and Discussions

The evaluation of our proposal is accomplished using the real bus GPS data. Data description, itinerary detection, and evaluation of interpolation errors are presented in **Sections 4.1 to 4.3**. Results for spatiotemporal integration of bus transit are presented in **Sections 4.4 to 4.6**.

4.1 Public Transit Data

The C3SL repository[1] is recognized as the main source of data for academic applications on public transit in Curitiba. It has been used in several studies.

[1] `http://dadosabertos.c3sl.ufpr.br/curitibaurbs/`

Geolocated data from bus monitoring are needed to deal with the map matching problem and static information from the transit network and the bus schedule. The Open Data Portal of Curitiba City Hall provides a daily updated database containing data on public transit in Curitiba available via WebService with relevant information such as GTSF Files, Lines, Points, Itineraries, Position of Vehicles, and Tables of Schedules. Data is transferred through files in JSON format through an API. The API data dictionary can be found in technical documentation [URBS, 2022b].

According to the operational data published in URBS [2022a], ITN has a fleet of 1,226 vehicles (disregarding the reserve buses) that serve 250 lines, 22 terminals, and 329 tube stations and make, on average, 1,365,615 trips per day useful. These vehicles periodically send their location according to URBS [2022b], which is stored in a daily log to be consulted via the API. Because the native service does not offer requests by date, C3SL provides JSON files containing a daily and complete history of ITN operations updated on day 1. An extensive exploratory analysis of these data is given in Vila *et al.* [2016].

The following data files of C3SL are used in the experiments:

- `Linhas`: contains code, name, service category, color, and other attributes of all ITN bus lines.
- `PontosLinhas`: stores name, code, type, latitude, and longitude of all ITN bus stops, and describes the correct sequence of stops according to bus line itineraries.
- `Vehicles`: contains the coordinate history of vehicles in operation. The GPS position of a vehicle with date and time is sampled every 20 seconds on average.
- `TabelaLinhas`: stores bus line timetables at stops; most stops do not have timetable information.
- `TabelaVeículos`: stores schedule times of bus itinerary segments.

4.2 Case Study - Bus Line 829

The bus line 829 "Universidade Positivo" ("Alimentador") was chosen for a case study. It is circular, i.e., the same start and end stops, with single-direction trips. In addition, it has few stops that make visualization and interpretation of the data easier. **Table 4** shows the scheduled itinerary of bus line 829, containing bus stop names and sequences.

Table 4. Scheduled itinerary of bus line 829 with bus stops.

Bus stop	Seq.
Terminal Campo Comprido	1
R. Angelo Nebosne, 75	2
R. Prof. Pedro Viriato Parigot de Souza, 4716	3
R. Prof. Pedro Viriato Parigot de Souza, 5136	4
R. Casemiro Augusto Rodacki, 233	5
R. Carlos Müller, 331	6
R. Carlos Müller, 871	7
R. Eduardo Sprada, 5273	8
R. Dep. Heitor Alencar Furtado, 5181	9
R. Dep. Heitor Alencar Furtado, 4900	10
Terminal Campo Comprido	1

The bus route starts at stop 1 in "Terminal Campo Comprido", reaches intermediate stops 2 to 10, and returns to the starting stop 1, as shown in **Figure 2**.

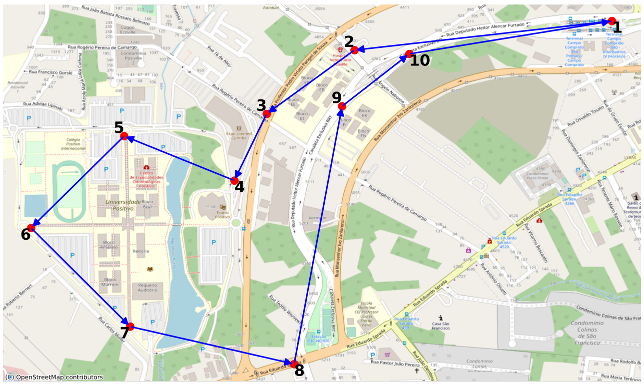


Figure 2. Programmed itinerary of line 829 that starts at point 1, passes through intermediate points 2 to 10, and returns to the starting point.

The scenario was built using real logs from 07/11/2022 of bus BA020. A round trip occurs between 06:04 to 06:32, during which there is no loss of GPS data. Therefore, this scenario is a suitable case to verify the application of the proposed algorithm and evaluate interpolation errors, simulating communication failures. Some logs are then deleted within specific time intervals. In this case, the map-matching algorithm does not detect the vehicle passing at some points, and the proposed algorithm can recover the information using interpolation. Points 3, 5, and 8 of bus line 829 were chosen to be removed from the original data set. The case study parameters are shown in **Table 5**.

Table 5. Case study parameters for line 829.

Line	829 - Universidade Positivo
Bus	BA020
Date	7/11/2022
Return time	06:04 to 06:32
Failure Interval	06:15 to 06:16 at point 3 06:17 to 06:19 at point 5 06:26 to 06:28 at point 8

Table 6 presents the result of **steps 1**, **2** and part of **step 3**. The procedure marks the exact time when bus BA020 passes at stops (map matching), creates a temporal sequencing (**step 2**), locates the itinerary, and assigns a sequence number to each log (part of **step 3**).

Table 6. Results of applying steps 1, 2, and part of step 3 to the case study.

Bus stop	Time	Sequence
Terminal Campo Comprido	06:04:51	1
R. Dep. Heitor Alencar Furtado, 4900	06:14:08	10
R. Angelo Nebosne, 75	06:14:36	2
R. Prof. Pedro Viriato Parigot de Souza, 5136	06:16:43	4
R. Carlos Müller, 331	06:19:30	6
R. Carlos Müller, 871	06:21:06	7
R. Dep. Heitor Alencar Furtado, 5181	06:28:30	9
R. Dep. Heitor Alencar Furtado, 4900	06:29:06	10
Terminal Campo Comprido	06:31:41	1

However, **Table 6** contains some inconsistencies. For example, the stop "R. Dep. Heitor Alencar Furtado, 4900" marked at 06:14:08 is incorrect because it is at the end of bus itinerary. A close examination reveals that the bus route between points 1 and 2 passes very close to point 10, as illustrated in **Figure 3**. In this case, the map-matching algorithm generates a markup error. This algorithm is thus insufficient

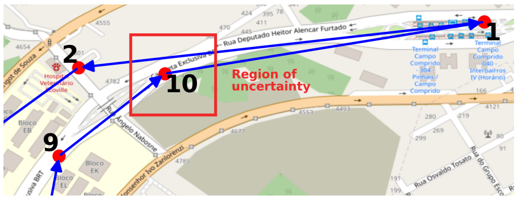


Figure 3. Region of uncertainty in which the map matching algorithm generates a marking error.

to handle logs properly. The marking error is identified by combining the result of map-matching with the proposed **Algorithm 1**.

The proposed algorithm identifies gaps in the sequence after completing **step 3**. Due to the communication failure simulation, it identifies points 3, 5, and 8 are lacking as shown in **Table 7**. The incorrect marking was deleted, and points 3, 5, and 8 were added to the itinerary due to temporal interpolation. This itinerary corresponds to the complete sequence of bus stops registered for line 829 with temporal information.

Table 7. Final result of applying Algorithm 1 to the case study.

Bus stop	Time	Sequence
Terminal Campo Comprido	06:04:51	1
R. Angelo Nebosne, 75	06:14:36	2
R. Prof. Pedro Viriato Parigot de Souza, 4716	06:15:39	3
R. Prof. Pedro Viriato Parigot de Souza, 5136	06:16:43	4
R. Casemiro Augusto Rodacki, 233	06:18:07	5
R. Carlos Müller, 331	06:19:30	6
R. Carlos Müller, 871	06:21:06	7
R. Eduardo Sprada, 5273	06:24:48	8
R. Dep. Heitor Alencar Furtado, 5181	06:28:30	9
R. Dep. Heitor Alencar Furtado, 4900	06:29:06	10
Terminal Campo Comprido	06:31:41	1

In evaluating the interpolation error, data from the movement of buses on line 829 during a whole day from 06:04 to 23:19 were used. Known points were randomly taken from the original data set, simulating communication failures. The estimation error $err_i = |t_i - \hat{t}_i|$ was calculated from the known real values (p_i, t_i) of the points removed and the estimated values $(p_i, \hat{t}_i)$.
Error measures are computed as a function of the number of consecutive bus stops missing. In this experiment, error measures are calculated in seconds for 1 to 7 consecutive missing points, or $w \in \{2, 3, \ldots, 8\}$. For each case, 100 samples without replacement were used to generate the result in **Figure 4**. It is observed that the interpolation error increases with the number of missing points. For most cases, the error ranges from less than 1 min to approximately 2 min (125 seconds).

This result suggests that the uncertainty introduced by interpolation is acceptable. A delay or advance of 2 minutes can be considered tolerable in an urban bus transit system. However, a closer look is needed to understand better which lines are most affected by the interpolation error at which times of day.

4.3 Database Assessment

The proposed algorithm was applied to logs on 07/11/2022 to evaluate the ability to detect itineraries using the entire database. The result is compared with the algorithm of

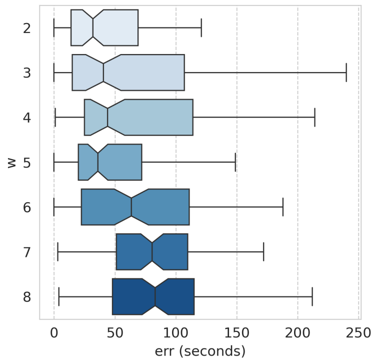


Figure 4. Interpolation error in seconds on line 829 for different values of w (a missing midpoint corresponds to $w = 2$).

Peixoto *et al.* [2020], which uses bus schedule tables. **Table 8** shows the total number of assignments each algorithm makes to a valid itinerary by type of bus line.

It is observed that the proposed algorithm provides a global increase from 68.83% to 99.33% in itinerary traceability gain. The new algorithm presents a result 44.31% better than Peixoto *et al.* [2020]. Except for lines of MADRUGUEIRO that should be further investigated, all lines of other types benefit. This result increases valid data in the database, preventing data from being discarded due to not being associated with any itinerary.

Table 8. Comparison between the number of tags assigned to a valid itinerary according to Peixoto *et al.* [2020] and the proposed algorithm by type of bus line. The percentage values are relative to the number of matches made by the map-matching algorithm.

Line type	**[Peixoto *et al.*, 2020]**		**Proposed**	
	Tags	**%**	**Tags**	**%**
ALIMENTADOR	160,750	70.62%	225,434	99.03%
CONVENCIONAL	52,338	62.02%	84,310	99.90%
EXPRESSO	21,986	62.41%	35,206	99.94%
JARDINEIRA	234	46.89%	499	100.00%
LIGEIRÃO	2,988	63.86%	4,677	99.96%
LINHA DIRETA	8,421	70.37%	11,879	99.26%
MADRUGUEIRO	5,659	98.26%	5,455	94.72%
TRONCAL	26,136	75.84%	34,447	99.96%
TOTAL	**278,512**	**68.83%**	**401,907**	**99.33%**

Table 9. Distribution of the tags of **Table 8** that had two error types: i) out of order and ii) missing bus stops for the proposed algorithm.

Line type	**Tags with errors**			
	i	**ii**	**Total**	**%**
ALIMENTADOR	15,764	21,176	36,940	16.39%
CONVENCIONAL	2,432	7,068	9,500	11.27%
EXPRESSO	487	2,139	2,626	7.46%
JARDINEIRA	0	12	12	2.40%
LIGEIRÃO	83	193	276	5.90%
LINHA DIRETA	126	162	288	2.42%
MADRUGUEIRO	1,470	283	1,753	32.14%
TRONCAL	478	1,896	2,374	6.89%
TOTAL	**20,840**	**32,929**	**53,769**	**13.38%**

Although the tags of **Table 8** are assigned to valid itineraries at the end of the proposed algorithm, some of them had errors in the sequence of stops or missing bus stops. **Table 9** shows the distribution of tags that had two error types: i) out of order and ii) missing bus stops for the proposed algorithm. The percentage of 13.38% of tags is practically all corrected after the adjustment of sequence and missing points performed at the end of the proposed algorithm.

The interpolation error by line type is evaluated similarly to **Section 4.2** but using all bus lines in the database that take only complete paths with real values. **Figure 5** shows the interpolation error in seconds by line type. Bus lines ALIMENTADOR, CONVENCIONAL, EXPRESSO, MADRUGUEIRO, and TRONCAL have an error between 0 and 1 min approximately. On the other hand, JARDINEIRA, LIGEIRÃO, and LINHA DIRETA are more susceptible to interpolation errors. The big errors for JARDINEIRA and LINHA DIRETA might be due to the long distances between bus stops of these lines, whose time interpolation can be significantly affected by road traffic conditions.

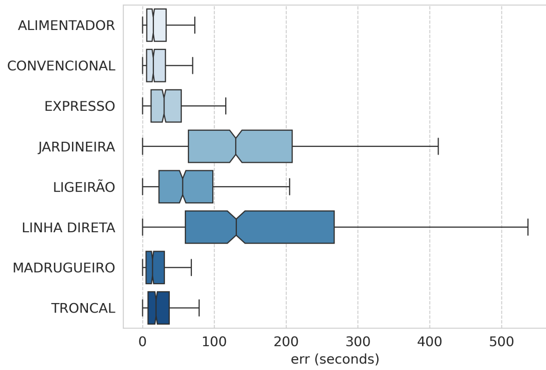


Figure 5. Line type interpolation error.

4.4 Temporal Assessment of Bus Service

A bus service is provided according to the timetables of bus lines. Therefore, users can estimate the time interval between consecutive buses of a given bus line. However, from a user's perspective, at a single bus stop, several buses from different bus lines interact to provide the bus service to that stop. In this case, how often do buses serve a particular bus stop (eventually by different bus lines)? This section provides a temporal assessment of bus services by identifying well-served urban areas with many buses and bus lines. It considers the database from 07/11/2022 to 07/15/2022. The GPS bus trajectories are tracked according to the method described in **Section 3**.

The bus service is evaluated in a time window of 10 minutes, representing an expected waiting time for most users. The number of buses that pass a stop is counted considering consecutive time windows of 10 minutes. This way, buses within a time window of 10 minutes are counted, a shift of 1 minute is then given to the window, allowing buses to be counted within the next 10 minutes, and so on. For each bus stop, a time series represents the number of buses observed in a 10-minute time interval in each minute of the day.

Figure 6 shows the average number of buses observed at a bus stop within a time window of 10 min shifted from 5:00 to 23:00 and aggregated into three categories of bus stops (terminal, street stop, and tube station). All bus stops inside a terminal are considered a single stop for capturing the number of buses available at a given time.

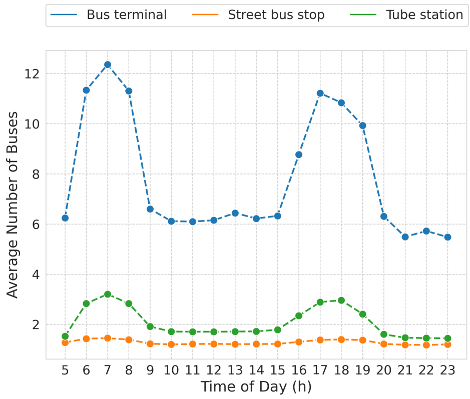


Figure 6. Average number of buses observed at a stop of terminals, street stops, and tube stations in a 10-minute moving window from 5:00 to 23:00.

Figure 6 shows a peak from 6:00 to 8:00, with a maximum of around 7:00, and from 16:00 to 19:00, with a maximum of around 17:00. This behavior is relevant for terminals and tube stations with minor effects on street stops. Moreover, terminals present up to 4 times more buses than tube stations at peak hours, as expected, because they are hubs integrating different bus lines. Terminals and tube stations play an important role in Curitiba's transit system because they offer users the possibility of transferring with a single fare.

The time series of **Figure 6** are then aggregated in a day, obtaining the average number of buses at a stop for terminals, street stops, and tube stations as shown in the boxplots of **Figure 7**. Terminals have the highest average number of buses, as expected. Street stops and tube stations have fewer buses but stand out as outliers, ranging from 2 to 13 buses on average. This means that street bus stops can also act as hubs if some integration between bus lines could be provided.

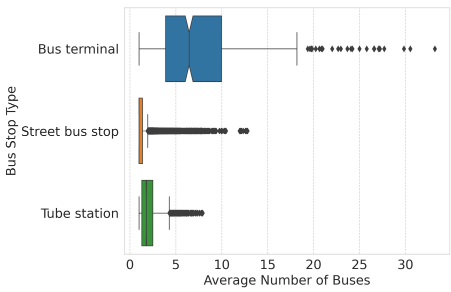


Figure 7. Boxplots showing the average number of buses observed at a stop of terminals, street stops, and tube stations in a 10-minute moving window.

The outliers of **Figure 7** represent an opportunity to improve the bus service because they have bus stops with a high frequency of buses. If the stops are close enough to each other, a hub can be built to allow connections between the respective bus lines. For instance, if temporal integration (with payment of a single fare) is allowed in certain regions of interest, new links between bus lines can be made in the network, eventually shortening distances and trip times. The regions of interest contain stops with a high concentration of buses, as shown in **Figure 8**.

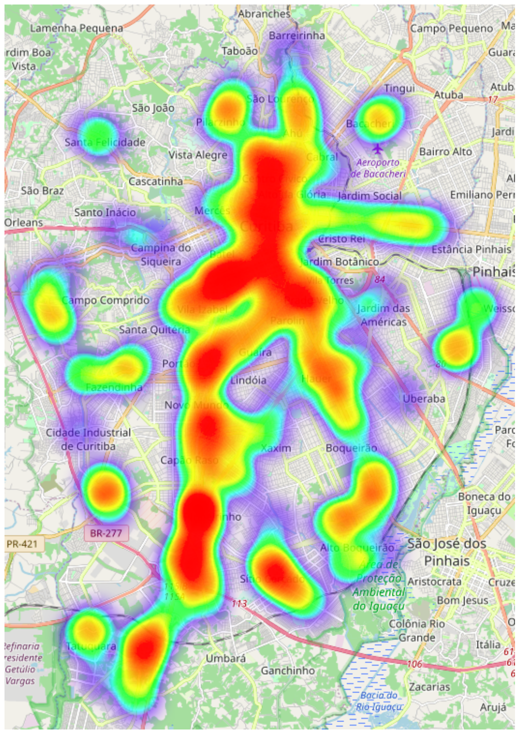


Figure 8. Heat map of regions with a high density of stops and frequency of buses obtained from outliers of street stops and tube stations.

It is a heat map obtained from outliers of street stops and tube stations (terminals are far from each other and usually do not have the potential for integration). The map highlights red regions with a high density of stops and frequency of buses. The most dense regions partially follow the North/South transit corridor. It means that buses running in this corridor have great potential to improve bus line connections in areas other than terminals. Based on this result, we aim to build bus stop clusters to behave like virtual terminals.

4.5 Virtual Terminal Evaluation

The results of **Algorithm 2** are shown in **Figure 9**. It shows centroids of 104 clusters computed with 27 bus stops, each serving 15 bus lines on average and covering 2,472 bus stops.

Figure 10 shows the average number of buses observed at a cluster in a 10-minute moving window from 5:00 to 23:00. Peak hours occur in the morning between 06:00 and 8:00 and between 16:00 and 19:00 in the afternoon. More than 100 buses, on average, are observed between 6:00 and 7:00.

There is a correlation between the average number of buses and the number of bus lines in a cluster according to **Figure 11**. It shows a Pearson's correlation coefficient of 0.78 with p-value < 0.001. In other words, not only do many buses attend a cluster, but also many bus lines.

However, it is necessary to show that some "synchronization" exists between buses passing at cluster stops during the day. It is accomplished by evaluating the correlation between the bus time series of two stops of the same cluster.

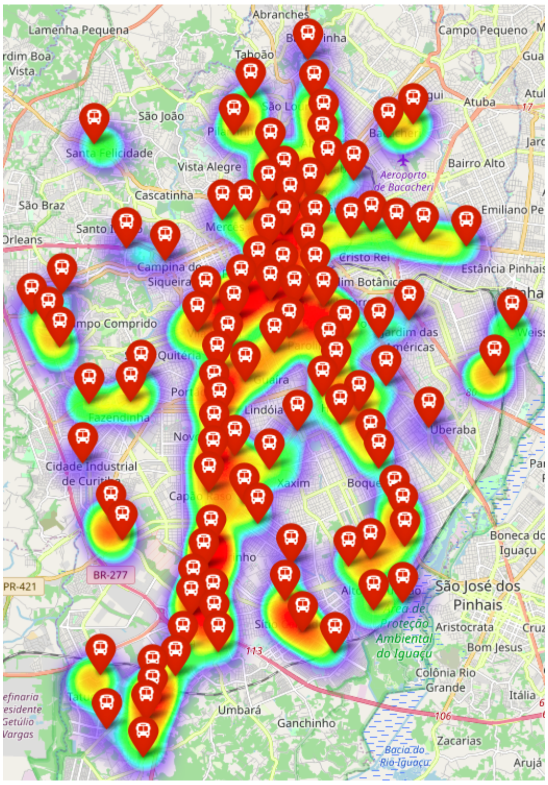


Figure 9. Centroids of 104 clusters.

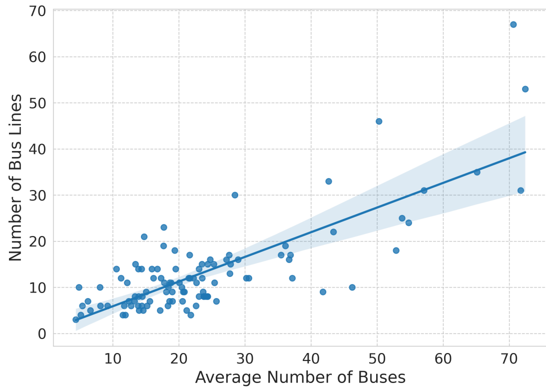


Figure 11. Scatter plot of the average number of buses observed in a 10-minute moving window and the number of bus lines of all 104 clusters.

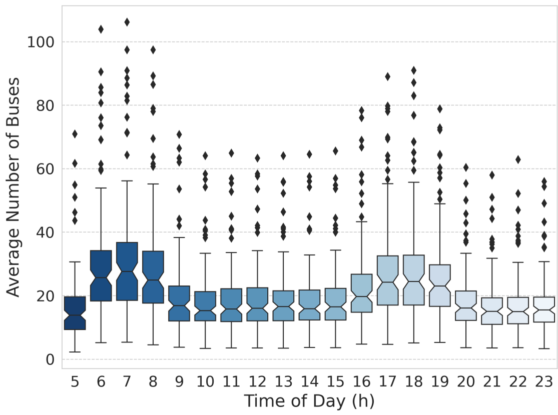


Figure 10. Boxplots showing the average number of buses observed at a cluster in a 10-minute moving window from 5:00 to 23:00.

Pearson's correlation between the time series of two bus stops is computed for all pairs of stops in a cluster. For instance, the correlation matrix between bus stops of the cluster with centroid 170121 is shown in **Figure 12**. A matrix is shown for each period of the day: i) morning from 6:00 to 9:00; ii) midday from 11:00 to 14:00. According to **Figure 12a**, there are pairs of bus stops whose correlation achieves 0.75, which means that buses can meet each other more often in the morning. A similar behavior is observed in the evening from 17:00 to 20:00, as shown in **Figure 12c**. However, this behavior is not observed midday, according to **Figure 12b**. Some pairs of bus stops with a strong correlation in the morning now show a weak correlation in the midday.

The correlation between the time series of two bus stops is averaged on all bus stop pairs of a cluster and then averaged on all 104 clusters for morning, midday, and evening periods. The results are shown in **Figure 13**.

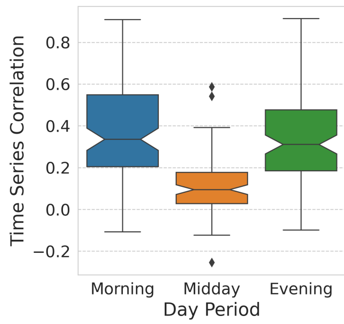
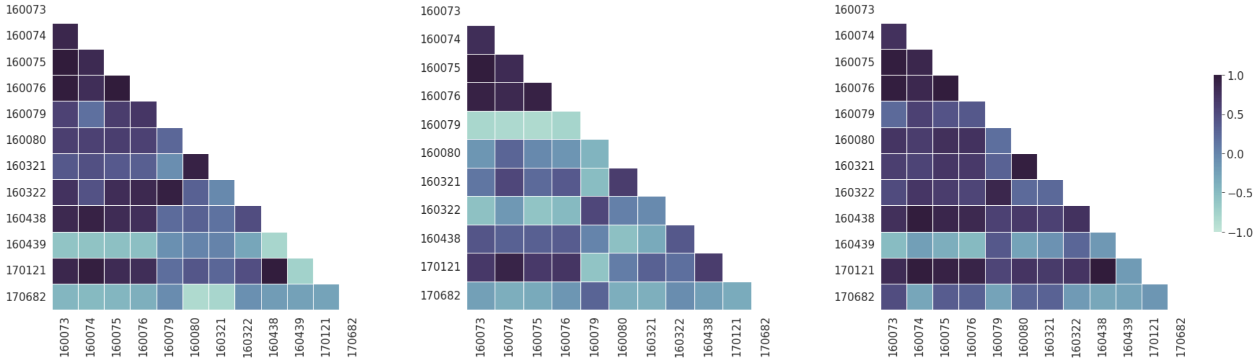


Figure 13. Correlation between time series of two bus stops averaged on all stop pairs of a cluster and all 104 clusters for morning, midday, and evening periods.

It can be seen that the average correlation is more relevant in the morning and evening. It means that buses passing in a cluster are better "synchronized" in the morning or evening on average, i.e., they can meet each other approximately simultaneously considering a time window of 10 min. It is then expected that users at bus stops of the same cluster can change between bus lines within 10 min on average.

When the time window increases, the correlation between the time series tends to increase; in other words, if the passenger is willing to wait longer, they will be more likely to make a bus line transition within the cluster. However, this is not valid for all periods of the day. **Figure 14** shows the results obtained using time windows of $\{10, 15, 20, 25, 30, 35, 40, 45\}$ minutes.



(a) Morning **(b)** Midday **(c)** Evening

Figure 12. Correlation matrix between bus stops in the cluster with centroid 170121 and all its neighbors for morning, midday, and evening.

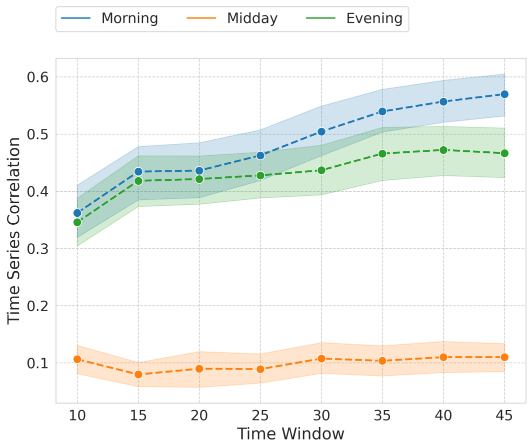


Figure 14. Observed correlation between time series for $\{10, 15, 20, 25, 30, 35, 40, 45\}$-minute moving window.

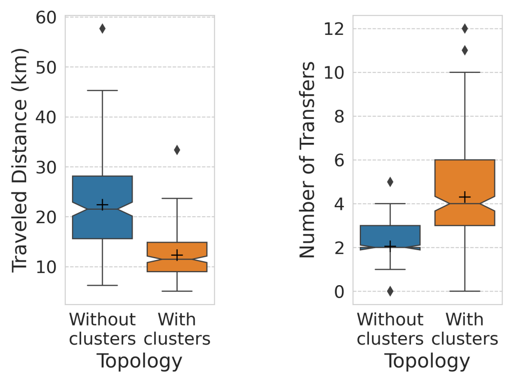
The result suggests an improvement in "synchronization" during the morning and evening, but this does not happen at midday. The explanation is that during this period, as it is not a peak demand, transit companies remove several buses from the streets, negatively impacting the correlation between the time series.

4.6 Impact of Spatiotemporal Integration on Bus Transit

Given an origin and a destination in the bus transit network, the impact of spatiotemporal integration is measured by the distance traveled and the number of transfers made in a trip with and without clusters of bus stops. Each cluster can be seen as a virtual terminal where transfers are made between bus lines with a single fare. The question is how these additional possible connections between bus lines benefit a trip. Networks with and without clusters of bus stops are shown in **Figure 15**.

The results presented in this section are based on an origin-destination (OD) survey by IPPUC in Curitiba [IPPUC, 2017]. Given an OD pair, the closest bus stops from the origin and destination are identified using a search distance of 600 m. Then, a short-path algorithm computes a feasible bus trip in the network and obtains its distance traveled and the number of transfers between bus lines. When a transfer is made in a cluster, an additional walking distance computed between the bus stops is considered. Because several trips can be found for a single OD pair, Yen's algorithm [Yen, 1971] computes K-shortest paths with $K = 30$. In other words, the number of shortest paths is limited to the top 30 alternatives. The results for distance traveled, and the number of transfers are shown in Fig. 16, with and without clusters.

According to **Figure 16a**, the average distance of 22.4 km traveled in the original network (without clusters) is longer than the average distance of 12.3 km using clusters. The opposite is observed concerning the number of transfers as shown in **Figure 16b**. The average value is two transfers without clusters, while the average number of transfers is four with clusters. Therefore, the results suggest that trip distances with bus clusters decrease by almost half at the expense of twice the number of transfers on average.

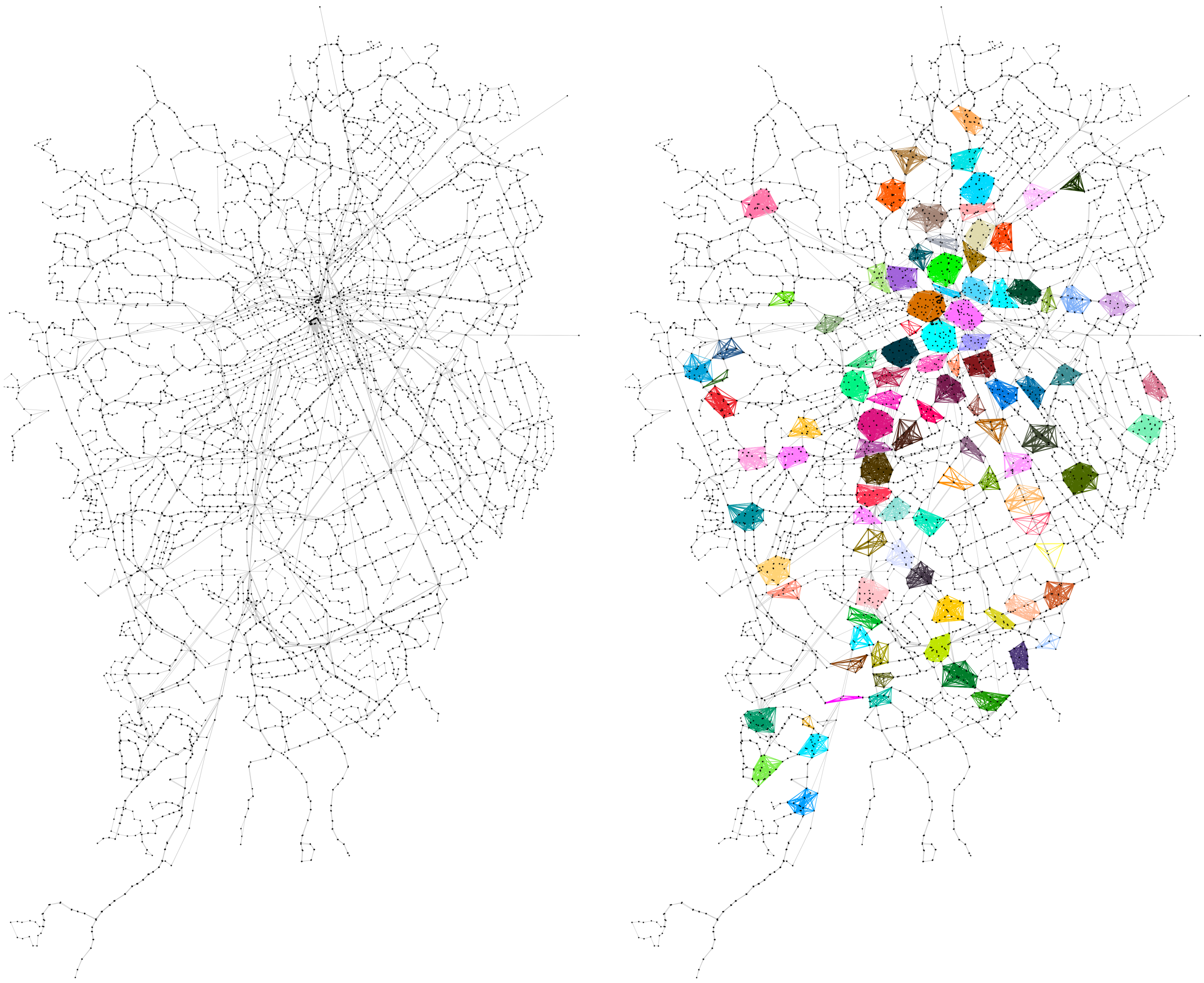


(a) Traveled distance (m) **(b)** Number of transfers between bus lines

Figure 16. Box plots with average values (+) of traveled distances and number of transfers for trips made with and without clusters of bus stops .

5 Conclusion

This work proposed data-driven approaches for detecting bus itineraries from GPS data and integrating bus transit in space and time. This spatiotemporal integration allows passengers



(a) Bus stops without clusters **(b)** Bus stops with clusters

Figure 15. Curitiba bus transit network with and without clusters of bus stops.

to switch bus lines with a single fare by defining "virtual terminals" in specific walking distance areas where transfers can occur during a limited timeframe.

The first algorithm for itinerary detection outcomes valid itineraries in most cases – improving other proposals in the state-of-the-art. The results show an increase from 68.83% to 99.33% in itinerary traceability gain when compared with a method that uses bus timetables. This result increases valid data in the database, preventing them from being discarded due to not being associated with any itinerary.

The second algorithm for bus stop clustering groups bus stops in walking distance areas for establishing "virtual terminals" where bus transfers can occur outside traditional physical terminals. An analysis using real-world origin-destination trips in Curitiba revealed that our approach could potentially reduce travel distances significantly. The average distance of 22.4 km traveled in the transit network without clusters is reduced to 12.3 km with clusters. However, it increases the number of transfers by two on average.

The results are limited regarding time estimated at bus stops because road traffic conditions should not affect them significantly when bus stops are located at short distances from each other. Another important limitation is using the correlation between bus time series to measure transfer times. A strong correlation means that buses are more likely to meet each other at bus stops of the same cluster.

Our contribution can enhance the efficiency of bus transit and even attract more people to public transit. Several future works could be done in this direction. For instance, it may be interesting to consider arrival times for evaluating and selecting routes with better synchronization and also allowing travel time to be computed.

Declarations

Acknowledgements

The authors thank Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES), Project Smart City Concepts in Curitiba, IPPUC, and Curitiba City Hall. JB thanks HUB de Inteligência Artificial e Arquiteturas Cognitivas - Brasil (H.IAAC), and Serviço Nacional de Aprendizagem Industrial - Brasil (SENAI).

Funding

JB is funded by the National Council for Scientific and Technological Development - CNPq (grant 142519/2020-0). This research was supported in part by the project SocialNet (grant 2023/00148-0 from Sao Paulo Research Foundation - FAPESP), CNPq (grants 441444/2023-7 and 314603/2023-9).

Authors' Contributions

JB and AP performed the experiments. JB, TS, AM, and RL helped in the conceptualization of the study and writing of the manuscript. JB is the main contributor and writer of this manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare they do not have competing interests.

Availability of data and materials

The datasets generated and/or analyzed during the current study are available in `https://github.com/jcnborges/busanalysis.git`.

References

Arriagada, J., Munizaga, M. A., Guevara, C. A., and Prato, C. (2022). Unveiling route choice strategy heterogeneity from smart card data in a large-scale public transport network. *Transportation Research Part C: Emerging Technologies*, 134. DOI: 10.1016/j.trc.2021.103467.

Bona, A. A. D., Fonseca, K. V., Rosa, M. O., Lüders, R., and Delgado, M. R. (2016). Analysis of public bus transportation of a Brazilian city based on the theory of complex networks using the p-space. *Mathematical Problems in Engineering*, 2016. DOI: 10.1155/2016/3898762.

Borges, J., Lüders, R., Silva, T., and Munaretto, A. (2023). Algoritmo para detecção de itinerários do transporte público usando dados de gps dos Ônibus. In *Anais do VII Workshop de Computação Urbana*, pages 1–14, Porto Alegre, RS, Brasil. SBC. DOI: 10.5753/courb.2023.739.

Chawuthai, R., Sumalee, A., and Threepak, T. (2023). GPS data analytics for the assessment of public city bus transportation service quality in Bangkok. *Sustainability*, 15(7). DOI: 10.3390/su15075618.

Curzel, J. L., Lüders, R., Fonseca, K. V., and Rosa, M. O. (2019). Temporal performance analysis of bus transportation using link streams. *Mathematical Problems in Engineering*, 2019. DOI: 10.1155/2019/6139379.

Desai, S., Suthar, R., Yadav, V., Ankar, V., and Gupta, V. (2022). Smart bus fleet management system using IoT. In *2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, pages 01–06. DOI: 10.1109/ICERECT56837.2022.10059646.

Gallotti, R. and Barthelemy, M. (2015). The multilayer temporal network of public transport in Great Britain. *Scientific Data*, 2:140056. DOI: 10.1038/sdata.2014.56.

Gubert, F. R., Santin, P., Fonseca, M., Munaretto, A., and Silva, T. H. (2023). On strategies to help reduce contamination on public transit: a multilayer network approach. *Applied Network Science*, 8(1):1–22. DOI: 10.1007/s41109-023-00562-7.

Hakeem, M. F. M. A., Sulaiman, N. A., Kassim, M., and Isa, N. M. (2022). IoT bus monitoring system via mobile application. In *2022 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pages 125–130. DOI: 10.1109/I2CACIS54679.2022.9815268.

IPPUC (2017). Consolidação de Dados de Oferta, Demanda, Sistema Viário e Zoneamento: Relatório 5 - Pesquisa Origem-Destino Domiciliar. Available at: `http://admsite2013.ippuc.org.br/arquivos/documentos/D536/D536_002_BR.pdf` (Last accessed: 2023-06-14).

Kumar, P. and Khani, A. (2023). Schedule-based transit assignment with online bus arrival information. *Transportation Research Part C: Emerging Technologies*, 155. DOI: 10.1016/j.trc.2023.104282.

Lawhead, J. (2015). *Learning geospatial analysis with Python*. Packt Publishing Ltd, Birmingham, 2nd edition.

Li, T., Meredith-Karam, P., Kong, H., Stewart, A., Attanucci, J. P., and Zhao, J. (2021). *Comparison of door-to-door transit travel time estimation using schedules, real-time vehicle arrivals, and smartcard inference methods*, volume 2675, pages 1003–1014. SAGE Publications Ltd. DOI: 10.1177/03611981211023768.

Li, T. and Rong, L. (2022). Spatiotemporally complementary effect of high-speed rail network on robustness of aviation network. *Transportation Research Part A: Policy and Practice*, 155:95–114. DOI: 10.1016/j.tra.2021.10.020.

Liu, T., Ji, W., Gkiotsalitis, K., and Cats, O. (2023). Optimizing public transport transfers by integrating timetable coordination and vehicle scheduling. *Computers and Industrial Engineering*, 184. DOI: 10.1016/j.cie.2023.109577.

Maduako, I. D., Wachowicz, M., and Hanson, T. (2019). Transit performance assessment based on graph analytics. *Transportmetrica A: Transport Science*, 15(2):1382–1401. DOI: 10.1080/23249935.2019.1596991.

Martins, T., Kozievitch, N., Gadda, T., Rosa, M., and Gutierrez, M. (2022). Map matching: Uma análise de dados streaming de trajetórias de GPS no transporte público. In *Temas Emergentes: Cidades Inteligentes (XVIII SBSI)*, pages 294–301. SBC. DOI: 10.5753/sbsi_estendido.2022.221647.

Motta, R. A., Silva, P. C. M. D., and Santos, M. P. D. S. (2013). Crisis of public transport by bus in developing countries: a case study from brazil. *International Journal of Sustainable Development and Planning*, 8:348–361. DOI: 10.2495/SDP-V8-N3-348-361.

Noichan, R. and Dewancker, B. (2018). Analysis of accessibility in an urban mass transit node: A case study in a bangkok transit station. *Sustainability (Switzerland)*, 10. DOI: 10.3390/su10124819.

Pan, L., Waygood, E. O., and Patterson, Z. (2023). Public transit itinerary choice analysis considering various incentives. *Transportation Research Record*, 2677:722–733. DOI: 10.1177/03611981231166682.

Panigrahi, N. (2014). *Computing in geographic information systems*. CRC Press, Boca Raton, Florida, 1st edition.

Peixoto, A., Rosa, M., Lüders, R., and Fonseca, K. (2020). Plataforma computacional para construção de um banco de dados de grafo do sistema de transporte de Curitiba. In *IV Workshop de Computação Urbana*, pages 125–137. SBC. DOI: 10.5753/courb.2020.12358.

Prathyusha, C., Singh, S., and Shivananda, P. (2021). Strategies for sustainable, efficient, and economic integration of public transportation systems. volume 121 LNCE, pages 157–169. Springer Science and Business Media Deutschland GmbH. DOI: 10.1007/978-981-33-4114-$2_1$3.

Queiroz, A. R., Santos, V., Nascimento, D., and Pires, C. E. (2019). Conformity analysis of GTFS routes and bus trajectories. In *XXXIV Simpósio Brasileiro de Banco de Dados*, pages 199–204. SBC. DOI: 10.5753/sbbd.2019.8823.

Rodrigues, D. O., Boukerche, A., Silva, T. H., Loureiro, A. A., and Villas, L. A. (2017). SMAFramework: Urban Data Integration Framework for Mobility Analysis in Smart Cities. In *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM '17, page 227–236, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3127540.3127569.

Rosa, M. O., Fonseca, K. V. O., Kozievitch, N. P., De-Bona, A. A., Curzel, J. L., Pando, L. U., Prestes, O. M., and Lüders, R. (2020). *Advances on Urban Mobility Using Innovative Data-Driven Models*, pages 1–38. Springer International Publishing, Cham. DOI: 10.1007/978-3-030-15145-4_57-1.

Sadeghian, P., Håkansson, J., and Zhao, X. (2021). Review and evaluation of methods in transport mode detection based on GPS tracking data. *Journal of Traffic and Transportation Engineering (English Edition)*, 8(4):467–482. DOI: 10.1016/j.jtte.2021.04.004.

Santin, P., Gubert, F. R., Fonseca, M., Munaretto, A., and Silva, T. H. (2020). Characterization of public transit mobility patterns of different economic classes. *Sustainability*, 12(22). DOI: 10.3390/su12229603.

Singla, L. and Bhatia, P. (2015). GPS based bus tracking system. In *2015 International Conference on Computer, Communication and Control (IC4)*, pages 1–6. DOI: 10.1109/IC4.2015.7375712.

Sridevi, K., Jeevitha, A., Kavitha, K., Sathya, K., and Narmadha, K. (2017). Smart bus tracking and management system using IoT. *Asian Journal of Applied Science and Technology (AJAST)*, 1(2). Available at SSRN: `https://ssrn.com/abstract=2941150`.

Steiner, K. and Irnich, S. (2020). Strategic planning for integrated mobility-on-demand and urban public bus networks. *Transportation Science*, 54:1616–1639. DOI: 10.1287/trsc.2020.0987.

URBS (2022a). Características da rede integrada de transporte. URL: `https://www.urbs.curitiba.pr.gov.br/transporte/rede-integrada-de-transporte` (Last accessed: 2023-03-27).

URBS (2022b). Web-service: Dados públicos da rede integrada do transporte coletivo de Curitiba. URL: `https://www.curitiba.pr.gov.br/dadosabertos/busca/?grupo=8` (Last accessed: 2023-03-27).

Vila, J. J. R., Kozievitch, N. P., Gadda, T. M., Fonseca, K., Rosa, M. O., Gomes-Jr, L. C., and Akbar, M. (2016). Urban mobility challenges–an exploratory analysis of public transportation data in Curitiba. *Revista de Informática Aplicada*, 12(1). DOI: 10.13037/ria.vol12n1.145.

Wang, S., Sun, Y., Musco, C., and Bao, Z. (2021). Public transport planning: When transit network connectivity meets commuting demand. pages 1906–1919. Association for Computing Machinery. DOI: 10.1145/3448016.3457247.

War, M. M., Rakhra, M., and Singh, D. (2022). Review on application based bus tracking system. In *2022 5th International Conference on Contemporary Computing and Informatics (IC3I)*, pages 876–880. DOI: 10.1109/IC3I56241.2022.10072449.

Wehmuth, K., Costa, B., Bechara, J. V., and Ziviani, A. (2018). A multilayer and time-varying structural analysis of the Brazilian air transportation network. In *Latin America Data Science Workshop*, volume 2170 of *CEUR Workshop Proceedings*, pages 57–64. DOI: 10.48550/arXiv.1709.03360.

Welch, T. F. and Widita, A. (2019). Big data in public transportation: a review of sources and methods. *Transport Reviews*, 39(6):795–818. DOI: 10.1080/01441647.2019.1616849.

Wilson, N. H., Zhao, J., and Rahbee, A. (2009). *The potential impact of automated data collection systems on urban public transport planning.*, pages 1–25. DOI: 10.1007/978-0-387-84812-9_5.

Yen, J. Y. (1971). Finding the k shortest loopless paths in a network. *Management Science*, 17:712–716. DOI: 10.1287/mnsc.17.11.712.

Yin, L., Hu, J., Huang, L., Zhang, F., and Ren, P. (2014). Detecting illegal pickups of intercity buses from their GPS traces. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2162–2167. DOI: 10.1109/ITSC.2014.6958023.

Żochowska, R., Kłos, M. J., Soczówka, P., and Pilch, M. (2022). Assessment of accessibility of public transport by using temporal and spatial analysis. *Sustainability (Switzerland)*, 14. DOI: 10.3390/su142316127.