Validating the coverage of bus schedules: A Machine Learning approach

João Mendes-Moreira\textsuperscript{a,b}, Luís Moreira-Matias\textsuperscript{c,a,b,*}, João Gama\textsuperscript{a,d}, Jorge Freire de Sousa\textsuperscript{e,f}

\textsuperscript{a} LIAAD-INESC TEC, 4200-465 Porto, Portugal
\textsuperscript{b} Dep. de Eng. Informática, Fac. de Engenharia, U. Porto, 4200–465 Porto, Portugal
\textsuperscript{c} Instituto de Telecomunicações, 4200–465 Porto, Portugal
\textsuperscript{d} Faculdade de Economia, U. Porto, 4200–465 Porto, Portugal
\textsuperscript{e} UGEI-INESC TEC, 4200–465 Porto, Portugal
\textsuperscript{f} DGEI, Fac. de Engenharia, U. Porto, 4200-465 Porto, Portugal

**A B S T R A C T**

Nowadays, every public transportation company uses Automatic Vehicle Location (AVL) systems to track the services provided by each vehicle. Such information can be used to improve operational planning. This paper describes an AVL-based evaluation framework to test whether the actual Schedule Plan fits, in terms of days covered by each schedule, the network’s operational conditions.

Firstly, clustering is employed to group days with similar profiles in terms of travel times (this is done for each different route). Secondly, consensus clustering is used to obtain a unique set of clusters for all routes. Finally, a set of rules about the groups content is drawn based on appropriate decision variables. Each group will correspond to a different schedule and the rules identify the days covered by each schedule.

This methodology is simultaneously an evaluator of the schedules that are offered by the company (regarding its coverage) and an advisor on possible changes to such offer. It was tested by using data collected for one year in a company running in Porto, Portugal. The results are sound.

The main contribution of this paper is that it proposes a way to combine Machine Learning techniques to add a novel dimension to the Schedule Plan evaluation methods: the day coverage. Such approach meets no parallel in the current literature.

© 2014 Elsevier Inc. All rights reserved.

### 1. Introduction

The bus has become a key player in highly populated urban areas. Inner-city transportation networks are becoming larger, and thus good operational planning is crucial. However, many vehicles are still failing to meet their schedules even when the defined plan is suitable. The *arriving as planned* factor is the most important achievement of both transit planners and passengers [21,27].

Throughout the 1990s, many mass transit companies started to install new computer-aided Bus Dispatch Systems. A vital part of these systems are the Automatic Vehicle Location (AVL) technologies. The installation consisted of equipping...
fleets with GPS-based communication systems capable of transmitting information on the status and the location of each vehicle to a data server with a certain but short periodicity. Many researchers understood the hidden potential of the stored AVL data to provide insights on how to evaluate (and then improve) the Operational Planning (OP) [26,12,27,2]. As a consequence, many research projects to collect and mine this massive amount of data arose in many cities around the world. Some examples include New Jersey, Chicago, Minneapolis, Seattle (United States); Ottawa and Montreal (Canada) or Eindhoven and The Hague (Netherlands). Promising insights on those case studies were reported in the survey presented by Furth et al. [10]. This paper focuses on improving the operational planning by mining AVL historical data.

Typically, a Schedule Planning process for a given route relies on three distinct steps: (1) the first step is defining the number \(k\) of schedules and their individual coverage, \(C_i\). Secondly, (2) the schedule time points are chosen among all the route bus stops, and finally, (3) timetables \(t_i\) for each route schedule \(S_i\) are defined containing the time the buses pass at each schedule time point (per trip). ¹ This process is done for all routes. It should be guaranteed that he number \(k\) of schedules and the coverage \(C_i\) are the same for all routes to facilitate the SP memorization by the passengers.

1.1. Literature review

The traditional Schedule Plan (SP) evaluation is highly focused on the time-tabling task. Specifically, this evaluation uses three key indicators: (1) on-time arrival performance, (2) headway adherence and (3) cruising time adherence [21,29,2]. A statistical framework to evaluate the schedule reliability levels – such as the running time adherence to the scheduled time or the headway regularity – is presented in Lin et al. [14]. In Patnaik et al. [23], the authors proposed a method to evaluate the defined timetable by clustering AVL data. The trips within the same cluster should have the same headway regularity defined throughout the day. One of the most common planning-related problems is the Travel Time Prediction (TTP) (or Arrival Time Estimation) and relevant research has been conducted to solve this problem [1,6,4,22]. Frameworks to propose changes on the timetables are using such TTP models. A good example is the work proposed by El-Geneidy et al. [9] – it uses multiple regression models to measure the variation between the scheduled times and (1) the actual headway and/or (2) the round-trip times (that is, bus cruising time, bus travel time, time elapsed between a vehicle departure and its arrival at the destination stop). The predictive models were employed to (re) define the round-trip times in the schedule points along the route. Studies focused on headway deviation effects – such as Bus Bunching² – were also conducted to evaluate schedule reliability. The research presented in Moreira-Matias et al. [20] introduces a novel framework to find frequent headway deviation sequences which explain this phenomenon. Despite the framework’s relevance to evaluate schedule reliability, it is clear that every timetable definition is largely dependent on the \(k\) number of existing schedules, and moreover, on its day coverage \(C_i\). However, at the best of our knowledge, there

¹ The step 2 just changes the way the schedules are presented to the public. It does not change neither the definition of trips nor the assignment of duties.
² This phenomenon occurs when two or more buses of the same line are running in a platoon or with a very short headway [20,15,5].
is no work in the literature regarding the evaluation of the suitability of the current number of schedules $k$ and respective day coverages $C$ to the current mobility needs.

1.2. Scope and goals

The purpose of this study is to create a methodology to validate the day coverage of a SP by mining the AVL historical data of a given set of routes. For that, we start with the following definition: let a day profile of a given route be a time series containing information on the trips where the time dimension (x-axis) represents a trip start time and the value dimension (y-axis) represents its round-trip time (see the light yellow curves of Fig. 1 to see examples of daily profiles).

This methodology relies on an interesting application of Machine Learning techniques such as consensus clustering [19] and rule induction [8] to discover relevant information on a massive amount of data (such as the AVL one). The main contribution of this methodology is that it proposes a way to combine such techniques to add a novel dimension to the SP evaluation methods: its daily coverage. Such approach meets no parallel in the current state-of-the-art.

1.3. Paper structure

The remainder of the paper is structured as follows: Section 2 presents some preliminary concepts and assumptions relevant to understand this problem. Section 3 describes the data acquisition process and its preparation in detail. Section 4 formally describes the approach to this problem and its main contributions to the existing literature. The fifth Section describes the experimental setup used and the results obtained. Such results are discussed in detail along Section 6, firstly (1) by highlighting the most relevant patterns and (2) by suggesting a possible Schedule Coverage to meet such constraints. Then, (3) by discussing the possibilities of deploying such methodology on a real world company and by quantifying its impact in our case study. Finally, conclusions from the work hereby described are drawn.

2. Preliminary concepts

Throughout this section, some additional details on improving the OP are provided. It starts by a brief description of the OP steps. Secondly, the SP is formally defined. Then, the concept of Schedule Coverage is formally introduced. Finally, the results obtained by our framework and their potential impact are discussed, along with some assumptions followed along this paper.

A typical OP process is executed by sequentially following four main steps [3,17]:

1. **Network definition**: It consists of defining the lines, routes and subsequent bus stops. Here, a route is considered as a road path between an origin and a destination which passes by multiple bus stops. A line is defined as a set of routes (which typically comprises two routes with very similar paths, but inversely ordered).

2. **Schedule planning**: Typically, the trips are defined by firstly identifying the set of bus stops to which schedule time points will be set (where the origin/destination stops are always part of this set). Secondly, timestamps are assigned to previously defined schedule time points. However, in highly frequent routes, this timetabling can also be defined by setting the time between two consecutive trips in the same route (i.e. headway-based) [29]. The set of resulting trips is often defined as the Schedule Plan.

3. **Definition of duties**: A duty is a task that a driver and/or a bus must perform. The definition of the drivers’ duties is subjected to many more constraints than the bus (for instance, a driver must go on breaks throughout the day; governmental legislation). Typically, the logical definition of bus duties is performed before the drivers’ duties.

4. **Assignment of Duties**: It consists of physically assigning the previously defined logical duties to the companies’ drivers and buses.

A Schedule Plan (SP) consists of a set $S = \{S_1, S_2, \ldots, S_n\}$ of $k$ schedules which provide detailed information about every trips running on the previously defined routes. Each schedule contains a timetable $\tau_i : i \in \{1, \ldots, k\}$. Different routes may have different timetables. Nevertheless, they share the number $k$ of schedules and the day coverage of each schedule (this should be common to every bus line to help the customers to easily memorize the SP). A definition of the day coverage $C_i$ in a given schedule $S_i$ is presented below.

Let $D = \{d_1, d_2, \ldots, d_i\}$ be a set of $s$ days of interest to include in a Schedule Plan (typically, $s = 365$ is used – it corresponds to a one year period). The day coverage $C_i$ of a given schedule $S_i$ is represented by the set of days where its corresponding timetable $t_i$ will be followed. It is possible to define it as:

$$C_i = \{d_1, d_2, \ldots, d_i\} : \bigcup_{i=1}^{k} C_i \neq D \land \theta_i > 0$$

(1)

where $\theta_i$ is the number of days covered by the timetable $t_i$. The set of every schedule day coverages $C = \{C_1, C_2, \ldots, C_k\}$ is called Schedule Coverage. An illustrative example of that is displayed in Fig. 3.
Once established, it is expected that an SP meets the passengers’ demand by following their mobility needs (namely, their mobility routines). However, today’s urban areas are characterized by a constant evolution of road networks, services provided and location (for instance, new commercial and/or leisure areas/facilities). Therefore, it is highly important to automatically assess how the SP suits the needs of an urban area. An efficient evaluation can lead to important changes on a SP. These changes will lead to: (1) a reduction in operational costs (for instance, by reducing the number of daily trips in a given route) and/or (2) a reliability improvement in the entire transportation network, which will increase the quality of the passengers’ experience and, therefore, the number of costumers [30]. Going from the previous definition of the steps required to build an SP, it is possible to divide the evaluation into two different dimensions: (1) the suitability of the number of schedules \( k \) and of the set of their day coverages \( C \) and (2) the reliability of their timetables \( \{t_1, \ldots, t_k\} \) (to test whether the real arrival times of each vehicle at each bus stop are meeting the previously defined timetable).

Then, two relevant assumptions are stated:

**Assumption 2.1.** Days with similar profiles should be covered by the same timetable, which means that they must be included in the same schedule.

**Assumption 2.2.** The number of schedules to use \( (k) \) is already known\(^3\);

Theoretically, all the days covered by the same timetable have exactly the same daily profile due to the fact that they share the same departing/arrival times. However, the real values of such times (given by the historical AVL data) may differ from the original ones. This paper, proposes a framework that explores such differences by grouping each one of the days available \( d_{ij} \) in one of the possible coverage sets, \( C_i, i \in \{1, \ldots, k\} \). This grouping is made according to a distance measured between each pair of days where the criteria rely on their profiles. As output, rules about which days should be covered by the same timetables are provided. Such rules can be used by the operational transportation planners to evaluate whether the current coverage is still meeting the network behavior (that is, the real departure and round-trip times). It also provides insights on how can the current coverage be changed in order to achieve that.

### 3. Data preparation

The case study in this work was the STCP (Sociedade de Transportes Colectivos do Porto), the main mass public transportation company in Porto, Portugal. The STCP has a total of 51 lines operating with their own resources. Their AVL system collects information on the location of each vehicle running every 30 s. Then the data is sent to the main server.

#### 3.1. Data collection

This study was conducted using a heterogeneous group of four lines – corresponding to six routes – that are representative of the entire network behavior by including all the three possible route types: circular, urban and non-urban routes. The data were collected during a one year-period from January to December 2007 (365 days). The selected bus lines were the 300, 301, 205 and 505. All four lines pass by the Hospital São João (HSJ), an important bus/light train interface in the city. Lines 300 and 301 are arterial urban circular lines, each one corresponding to one route. These lines are quite similar, but with opposite directions and they connect the city center to the HSJ, passing by another important bus/light train/train interface, which is the São Bento train station. Lines 205 and 505 both have two routes each: outward and return. Line 205 follows almost the entire peripheral road that marks the city limits, crossing several entrances to the city and several mass transport interfaces, such as Campanhã, which is the main train station. Finally, line 505 serves a suburban area, connecting Porto to a neighboring town, where there is a sea port. The line ends at the HSJ.

An illustration of these routes on the road network in the urban area of Porto is displayed in Fig. 2. The orange dots represent the bus stops of each route.

#### 3.2. The schedule plan in place

In any SP studied, it is necessary to detail particular seasons that are important to the framework due to their impact on the passengers’ behavior. In this case study, these seasons are (1) Easter time (ET), (2) Christmas Time (CT) and (3) School Holidays (NSP). The ET represents the period contained in the first eight days of April and the CT corresponds to the last nine days of December. The NSP was set as the period between 15 July and 15 September (including these two boundary days).

The SP at the STCP had a total of four schedules (i.e. \( k = 4 \)) during the year of 2007. Their Schedule Coverage was arranged as follows: Schedule 1: Saturdays; Schedule 2: Sundays and Holidays; Schedule 3: working days during school holidays; Schedule 4: working days outside school holidays. Fig. 3 illustrates the Schedule Coverage.

---

\(^3\) The selection of the number of schedules is not within the scope of this paper due to its connection with other variables, such as bus frequency, passenger arrival pattern and the traffic volume over road segments [23].
Fig. 2. Illustration of some routes (one per line) considered over a geographical representation of the road network in Porto, Portugal. Image obtained from [25].

Fig. 3. Schedule Coverage in the case study during the year 2007. The H-symbols represent holidays.
3.3. Preprocessing

The data were firstly collected for a PhD study and extensively treated and prepared. This is described in detail for a specific route (78-1-1) in Sections 2.5.2 and 5.1 of thesis [17]. A similar process was conducted to obtain the present data and it is briefly described below.

The fleet is equipped with differential GPS devices able to communicate each vehicle’s position to the AVL data server. This information is automatically sent to the data server in real-time using General Packet Radio Service (GPRS). The relevant trip data is stored in two different tables from the AVL data server: trip starting time and trip ending time. Obtaining the trip data is not a direct process due to the lack of a primary key identifying each trip individually on the server’s database. It is necessary to (1) sort the data and then (2) match pairs of trips starting/ending times, thus making it possible to obtain the round-trip times. The data (1) sorted using the timestamps of vehicle’s location associated to each trip. The pairs were matched by identifying the records containing each trip’s beginning/ending – consequently, it is possible to compute the respective round trip times. Using these times, it is possible to build route datasets. Each dataset has one entry for each trip containing the following information: the starting date of the trip, the departure time, the bus model, the code of the driver, the code of the route service, the day of the year, the type of day (normal day, holiday and floating holidays) and duration of the trip.

As part of the preprocessing task, new datasets were constructed based on the original set. The authors did so because the original database has some missing values and also an excessive amount of information regarding this specific task. The new dataset contains only the day, the week day, its type and an ordered sequence of round-trip times for the trips completed during the day. The first three variables are used to address the coverage details, while the ordered sequence of round-trip times is used to define groups of days with similar profiles of round-trip times.

Some route values are missing (64 days in 365 × 6 days possible – see Table 1) due to the lack of pair matching and/or other communication failures. To overcome this issue, the expected round-trip time profiles were calculated. An example on these profiles are the light yellow curves in Fig. 1. Such curves represent round-trip time profiles of multiple days calculated from a route of interest using the data of the remaining days.

The computation of expected round-trip time profiles for the days with missing data consisted of firstly (1) selecting data from the same route but from other days with the same type (for instance, if there were missing data about a certain Tuesday, the information about other Tuesdays would replace it). Then, (2) an expected round-trip time profile is built by using both (2a) the number of trips of the most recent similar day (that is the last Tuesday) and (2b) the round-trip times of every past similar days. This preprocessing method forms an expected profile for a day with missing data by calculating averages of (2a) these round-trip times into a number of bins equal to (2b) such number of trips. The error introduced by such interpolation method is not significant since the percentage of missing days in every route (2.9% per route on average) is not sufficient to change the output rules that defines the Schedule Coverage in place (which would need, in general, a larger support in the input dataset).

3.4. Summary

Table 1 presents a summary of the data used. The columns are the six routes denominated by a XXX_Y mask, where the XXX corresponds to each line and the Y corresponds to the direction considered. The table rows correspond to (1) the total number of trips considered in each route and (2) the number of days with missing data – all in the period considered; (3 and 4) the maximum/minimum number of daily trips (i.e. DT, in number of trips) in the same period; (5 and 6) the maximum/minimum travel time (TT – round trip time, in seconds) ever registered for a trip on such period; (7) the median, (8) the mean and, finally, (9) the coefficient of variation of the travel time.

It is possible to observe that line 205 presents a higher number of trips than any other route considered. Lines 300 and 301 present higher round-trip times than the other lines. All the lines present approximate Coefficients of Variation (i.e. the std. dev. of such coefficient from route to route is only σ = 0.0049). This index can be faced as a relative Standard Deviation which exhibits the TT relative variability on each route. These results suggest that such variability is similar from route to route. Yet, it is not possible to infer more than this based only on such coefficients.

<table>
<thead>
<tr>
<th></th>
<th>205_1</th>
<th>205_2</th>
<th>505_1</th>
<th>505_2</th>
<th>300_1</th>
<th>301_1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of trips</td>
<td>21640</td>
<td>20813</td>
<td>9277</td>
<td>5198</td>
<td>13906</td>
<td>14042</td>
</tr>
<tr>
<td>Missing days</td>
<td>16</td>
<td>14</td>
<td>1</td>
<td>3</td>
<td>26</td>
<td>4</td>
</tr>
<tr>
<td>Maximum DT</td>
<td>80</td>
<td>78</td>
<td>37</td>
<td>25</td>
<td>58</td>
<td>59</td>
</tr>
<tr>
<td>Minimum DT</td>
<td>6</td>
<td>11</td>
<td>7</td>
<td>4</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>Maximum TT</td>
<td>4799</td>
<td>4800</td>
<td>4493</td>
<td>4500</td>
<td>5299</td>
<td>5797</td>
</tr>
<tr>
<td>Minimum TT</td>
<td>1842</td>
<td>1828</td>
<td>1602</td>
<td>2085</td>
<td>2165</td>
<td>2278</td>
</tr>
<tr>
<td>Median TT</td>
<td>3413</td>
<td>3299</td>
<td>3049</td>
<td>3503</td>
<td>4218</td>
<td>4242</td>
</tr>
<tr>
<td>Mean TT</td>
<td>3416.04</td>
<td>3313.06</td>
<td>3130.75</td>
<td>3495.10</td>
<td>4203.55</td>
<td>4344.07</td>
</tr>
<tr>
<td>Coef. variation TT</td>
<td>0.1285</td>
<td>0.1349</td>
<td>0.1427</td>
<td>0.1316</td>
<td>0.1279</td>
<td>0.1326</td>
</tr>
</tbody>
</table>
4. Methodology

4.1. Framework analysis

The validation framework presented in this paper is divided into three simple steps: firstly, (1) the running times are extracted from the AVL data of just one route and clustered to obtain the optimal day coverage for this specific route (each cluster will correspond to a possible schedule). This step is repeated by every route of interest. Secondly, (2) the Schedule Coverage of each route is assembled to create a consensual cluster that is common to every route in the network, using consensual clustering techniques. Finally, (3) rules are extracted, obtaining a new SP day coverage (a feasible and readable coverage plan for the entire network). These steps are described below.

Step 1 starts by clustering the day profiles (extracted from a given route) into a predefined number of \( k \) schedules/clusters. The days in each group will then indicate the coverages \( C_i : i \in \{1, \ldots, k\} \) for an SP running on a specific route. A methodology for that has already been proposed in our previous work [16].

Since each route data will produce different partitions (for instance, different day coverages), this framework is only able to produce an individual analysis to one route at a time, which will correspond to the optimal coverage for that single route of interest. Nevertheless, it is not acceptable that each route has its own Schedule Coverage. Therefore, the work in [16] is just a proof of concept about what can be done by mining the AVL data focused on this specific topic. Therefore, it is not applicable just per se.

Hereby, a novel framework is proposed that by using this previous work as a building block is able to extend its functionality to multiple routes. As a result, it provides a consensual schedule day coverage to as many routes as necessary. Besides the aforementioned first step, the validation framework presented here consists of two novel steps: in Step 2, a consensual day coverage for the schedule network is mined from the partitions extracted from distinct routes. This is done using a well-known consensual clustering technique [19]; finally, in Step 3, rules are extracted from the consensual clusters obtained and compared to the existing plan. The aim of this step is to turn the resulting clusters into rules which are easily understandable by a larger audience. To do so, a rule induction algorithm was used: the RIPPER [8].

The proposed framework can be seen as a hypothesis test having as null hypothesis the fact that the current SP fits the network behavior (however, it is not possible to state which is its significance). The identified changes are the most critical because the network behavior is already conditioned by the previously defined SP. In the same line, important planning variables, such as passenger demand and the timetable arrival times, are not directly considered (even when the coverage in place changes are already affecting the round-trip times and, therefore, the profiles obtained). The evaluation of the timetables and of the number of schedules is not addressed in this paper. In fact, it is necessary to assume a predefined number of schedules to evaluate the Schedule Coverage using this methodology (go to Assumption 2.2 for more details on this matter).

4.2. Main contributions

The major contribution of the work presented here relies on steps (2) and (3) of this framework. Step (2) overcomes the limitations of the work in [16] by extending the number of possible routes to analyze simultaneously from one to many. Without this step, the previous methodology is not usable by a company as a method to evaluate the existing Schedule Coverage. In order to be useful, a unique schedule day coverage should be considered by all routes. The third step makes it easier to interpret the framework outputs by providing comprehensible rules which do not require any previous Machine Learning knowledge to be fully understood. This is highly relevant since the main users of this methodology will not be data miners or

Fig. 4. Generic representation of the framework. The left red dotted square delimits the basic part of this framework proposed in [16]. The right blue square highlights the contributions of this paper. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
Machine Learning experts, but be transport operational planners who want to extract information from the historical AVL data about if and how they may change the Schedule Coverage to improve the SP reliability.

An illustration of this methodology is presented in Fig. 4. The datasets contain the trip information on each different route. The samples of each route are then clustered based on a distance measurement of interest (i.e. Dynamic Time Warping (DTW)) and a consensus between all routes is found. Rules are then extracted from such consensual partitioning. The work in [16] is represented by the building block surrounded by a red dotted square, on the left part of the figure, which corresponds to step 1 of this framework, while the right blue square highlights the work contribution of this paper, i.e., steps 2 and 3.

A formal definition of the methodology presented here is enunciated below.

4.3. Step 1: How can we find the optimal schedule for a single route using AVL data?

Let \( X = \{x_1, x_2, \ldots, x_n\} \) be a set of \( n \) datasets (for instance, AVL historical data from \( n \) routes) of interest with the same number of samples \( s \) (the objects that need to be clustered consensually). The initial datasets \( X \) were firstly turned into new datasets, having each one an entry for each day present in the initial dataset. The information stored per day is a sequence of pairs with the departure time and round trip time. This forms irregularly spaced data sequences (ISDS) of round trip times (i.e. each day has a different number of trips).

The idea is to build a quadratic matrix of distances \( s \times s \) which maintains the distance between each day based on the ordered series of round trip times (i.e. by the trip departure times). Then, this matrix is the input of a \( k \)-dependent clustering algorithm of interest that proposes an ideal SP for that specific route (\( P_n = \{C_{1n}, C_{2n}, \ldots, C_{kn}\} \)). However, common distance measures – such as the Euclidean – are very sensitive to variations in both depth and in granularity of the time axis, such as the ISDS used here. To overcome this problem, the use of the Dynamic Time Warping (DTW) distance algorithm is proposed. This was firstly proposed by Chu et al. [7]. This algorithm is formally presented below according to its original definition.

### 4.3.1. Dynamic Time Warping (DTW)

Let \( Z_n \) and \( Q_n \) be two sequences having the lengths \( n \) and \( m \), respectively, where \( n \) may not be equal to \( m \). If the aim is to align them using DTW, it is necessary to construct an \( n \)-by-\( m \) matrix containing the distances between all points in the two series. Then, a warping path is defined. This warping path is a contiguous set of matrix elements that defines a possible and optimized mapping between \( Z \) and \( Q \). The \( u_{im} \) element of the warping path is defined as \( w_u = (i,j) \), and therefore we have \( W = \{w_1, w_2, \ldots, w_{s_1}, w_{s_2}, \ldots, w_{s_{m_1}}\} \), which requires the validity of the following in-equation:

\[
\text{max}(m, n) \leq U \leq m + n - 1
\]

This path is subjected to three major constraints:

1. **Boundary conditions**: \( w_1 = (1, 1) \) and \( w_u = (m, n) \). This requires that the warping starts and ends in the diagonally opposite cells of the matrix.
2. **Continuity**: Let \( w_u = (a, b) \). Then \( w_{u-1} = (a', b') \), where \( a - a' \leq 1 \) and \( b - b' \leq 1 \). This restricts the possible steps in the warping path to adjacent cells (including diagonally adjacent cells).
3. **Monotonicity**: Let \( w_u = (a, b) \). Then \( w_{u-1} = (a', b') \), where \( a - a' \geq 0 \) and \( b - b' \geq 0 \). This forces the points in \( W \) to be monotonically spaced in time.

In order to build an optimized path satisfying the conditions above, it is necessary to minimize the warping cost:

\[
\text{DTW}(Z, Q) = \min \left\{ \frac{\sqrt{\sum_{u}^n w_u}}{U} \right\}
\]

### 4.4. Steps 2, 3: finding consensual rules to build a schedule plan

By partitioning the datasets (the AVL data) into \( k \) clusters (for instance, schedules forming a possible SP), it is possible to define the partitions of \( X \), called \( P \), according to the following definition:

\[
P = \{P_{11}, P_{12}, \ldots, P_{1k}, P_{n1}, P_{n2}, \ldots, P_{nk}\}, k \geq 2 \land k \in \mathbb{N}
\]

where \( \bigcup_{m=1}^{k} P_{mn} = X \). \( P_{ij} \cap P_{il} = \emptyset, \forall i \neq j, k \) represents the optimal day coverage for a given route \( i \) (i.e. the schedule day coverage set \( \{C_{1i}, C_{2i}, \ldots, C_{ki}\} \)). By defining \( P \) as the \( k \) partitions formed from the \( n \) input datasets, it is necessary to establish a **new distance measure** between each possible pair of days \( d_i \) based on the agreement between the partitions (i.e. a consensual clustering of the data provided by every input routes).

Let \( M_i(s \times s) \) (for instance, a quadratic matrix with the number of days considered for each route where each position is set as 1 if the days are in the same schedule and 0 if they are not) be the co-association matrix (or connectivity matrix)
representing the clustering membership for the samples in the $X_i$ data set and a given number of partitions $k$. It can be obtained as follows:

$$M(r,j) = \begin{cases} 1 & \text{if } r \in P_i \land j \in P_m, l = m \\ 0 & \text{if } r \in P_i \land j \in P_m, l \neq m \end{cases}, l, m \in \{1, \ldots, k\} \land l, m \in \mathbb{N}$$

(6)

Then, it is possible to calculate the agreement matrix $\mathcal{M}$ (the consensus between every SP found) and the distance consensus matrix $\mathcal{D}$ using the following equation:

$$\mathcal{M} = \sum_{m=1}^{k} \frac{M_m}{n}, \mathcal{D} = 1 - \mathcal{M}$$

(7)

The resulting matrix $\mathcal{D}$ is a quadratic $s \times s$ distance matrix related to all samples (the distances between all days considered). By applying a $k$-dependent clustering algorithm of interest to $\mathcal{D}$, it is possible to obtain the dataset $\mathcal{P}$ of $k$ consensual partitions from the datasets in $X$:

$$\mathcal{P} \equiv \text{clusteringAlgorithm}(\mathcal{P}, k) \equiv \{P_1, P_2, \ldots, P_k\}$$

(8)

where each $P_i : i \in \{1, \ldots, n\}$ will contain a set of days $\{d_1, \ldots, d_z\} : z > 0$. Using the consensus function definition described in Eqs. (6) and (7), it is possible to obtain the consensus clustering for the input datasets. Using these new partitions, logical rules can be extracted using a rule induction algorithm such as the RIPPER [8].

5. Experimental results

This section starts by describing the experimental setup used in the experiments. Then, the results obtained with the setup are presented.

5.1. Experimental setup

Firstly, the $k$-Means was chosen as a clustering algorithm since it was employed in the experimental setup of our previous work [16]. To reduce the $k$-Means random start effects, a deterministic divisive hierarchical clustering was employed, as proposed in [28].

Secondly, both the individual and the consensual clustering experiments were carried out using the R language [24]. The $k$ parameter values varied from 2 to 7. This was done because it is not acceptable and/or common to have a number of schedules outside this range.\(^4\)

Finally, the J-RIP algorithm – the JAVA implementation of the RIPPER algorithm – was applied to the consensual partitions using the WEKA software [11]. A set of seven intuitive decision variables (i.e., features) was used to characterize each day: (1) WEEKDAY: the day of the week (Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday); (2) DAYTYPE: the type of the day (holiday, normal, non-working day, weekend holiday) where a non working day represents a working day where the public sector does not work – even if it is not an official holiday; (3) MONTH: \{1, \ldots, 12\}; (4) EASTERTIME: boolean, (5) CHRISTMASTIME: boolean and (6) NONSCHOOLPERIOD: boolean; (7) SCHEDULE: the schedule proposed for each day \{1, \ldots, k\}. Such variables are then used by the RIPPER to output rules that can meet as much as possible the coverage proposed for each schedule using the SCHEDULE variable as target.

The RIPPER outputs a set of rules in a hierarchically divisive form (e.g., like a decision tree based on rules). An accuracy evaluation metric was defined as

$$\text{Accuracy} = \frac{\text{Number of Days Classified Correctly}}{\text{Total of Days}}$$

(9)

Typically, an accuracy metric is employed in classification problems – which is not our case. Nevertheless, it is employed here to evaluate how representative the rule set is of the coverage proposed by the consensual clustering process. This was done by measuring a possible accuracy as if the obtained rule set was considered as a classifier (which is the core of the J-RIP algorithm). Comparative tests using the same partitions (i.e., training sets) as test sets were then performed (i.e., each Schedule is considered a possible class (SCHEDULE) and each day is seen as a sample defined by the values of the remaining six features).

The J-RIP algorithm takes four parameters: (1) FOLDS: it determines the amount of data used for the pruning stage\(^5\); (2) WEIGHT: the minimum total weight of instances in a rule (i.e., it works like a minimum support threshold to consider a rule as meaningful); (3) OPTIMIZATIONS: the number of runs in the optimization process and (4) SEED: a numerical seed used to randomize the data. The following default values were used to this parameter set: 3, 2, 2 and 1, respectively. In this paper, the purpose on employing RIPPER is to demonstrate that it is able to extract rules (which highlight the patterns underlying on our data) for those who are not familiar with Machine Learning techniques. Consequently, no sensitivity analysis was carried out on such parameter value combination and this value set was used in every experiment conducted.

\(^4\) It is desirable to have a number of schedules as low as possible to make it easier for the passengers to memorize the SP [10].

\(^5\) The data is divided into multiple folds; typically one of the folds is used on pruning while the others are used to grow the rules.
It is relevant to highlight that this methodology is not an automatic classifier to assign a Schedule to each day. The primary goal of the SP is to meet a certain expected demand minimizing the quantity of resources employed \[3\]. However, changes on the Schedule Coverage (e.g. to force the Saturdays to have the same timetable as the Sundays and Holidays) may not be possible due to these previous definitions (e.g. number of drivers and/or vehicles available on Saturdays). This framework should be seen as a decision support tool that should be used together with other information, namely, the resources available in each scenario.

5.2. Results

The results are displayed in three distinct dimensions: (1) the resulting distribution of days along the clusters is presented in Table 2; (2) an illustration of the distribution of days among the \( k = 4 \) clusters (i.e. the Schedule Coverage) is shown in Fig. 5; (3) Fig. 6 presents a decision tree exhibiting the rules learned from the consensus clustering using 2–4 schedules.

The acronyms used in Table 2 can be defined as follows: TOT is the total number of days within the cluster; MON (Monday) to SUN (Sunday) corresponds to the number of days in the cluster by weekdays; the HOL column represents the holidays (including the ones during the weekend), the SHO is the sum of the floating holidays and non-working days; the NSW and the NSH are, respectively, the working days and the weekends during school holidays; the CHR represents the days in Christmas Time and the EAS is the Easter period.

Fig. 5 displays the Schedule Coverage provided by the framework presented for a scenario with four Schedules. The \( x \)-axis represents the days of the year where the first day of each month is highlighted with an axis caption. The colored points correspond to the days and the colors represent different months of the year. The \( y \)-axis are the possible schedules where the days can be grouped. This figure shows seasonalities (i.e. a day of the same type that is grouped in different schedules depending on the month) that are not observable in Table 2.

In Fig. 6, the circles represent the schedule found in each tree leaf, while the rectangles contain the conditions in each tree node. The left branches should be followed when the condition is satisfied. The accuracy achieved by each set of rules in the three Schedule Coverages considered \( k = \{2, 3, 4\} \) were 0.97, 0.78 and 0.77, respectively.

<table>
<thead>
<tr>
<th></th>
<th>MON</th>
<th>TUE</th>
<th>WED</th>
<th>THU</th>
<th>FRI</th>
<th>SAT</th>
<th>SUN</th>
<th>TOT</th>
<th>HOL</th>
<th>SHO</th>
<th>NSW</th>
<th>NSH</th>
<th>CHR</th>
<th>EAS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k = 2 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>48</td>
<td>46</td>
<td>47</td>
<td>47</td>
<td>47</td>
<td>1</td>
<td>0</td>
<td>236</td>
<td>0</td>
<td>44</td>
<td>0</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>48</td>
<td>48</td>
<td>106</td>
<td>12</td>
<td>2</td>
<td>1</td>
<td>17</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>( k = 3 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>24</td>
<td>21</td>
<td>22</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td>117</td>
<td>0</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>19</td>
<td>19</td>
<td>21</td>
<td>22</td>
<td>18</td>
<td>1</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>1</td>
<td>11</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>29</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>29</td>
<td>0</td>
<td>0</td>
<td>136</td>
<td>0</td>
<td>2</td>
<td>33</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>( k = 4 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>25</td>
<td>24</td>
<td>21</td>
<td>22</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td>117</td>
<td>0</td>
<td>1</td>
<td>13</td>
<td>0</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>17</td>
<td>15</td>
<td>13</td>
<td>15</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>68</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>7</td>
<td>12</td>
<td>11</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>1</td>
<td>28</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( k = 5 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>0</td>
<td>0</td>
<td>65</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>7</td>
<td>14</td>
<td>13</td>
<td>17</td>
<td>0</td>
<td>0</td>
<td>66</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td>0</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>9</td>
<td>47</td>
<td>11</td>
<td>107</td>
<td>11</td>
<td>2</td>
<td>1</td>
<td>17</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>( k = 6 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>14</td>
<td>14</td>
<td>12</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>63</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>11</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>57</td>
<td>0</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>47</td>
<td>46</td>
<td>57</td>
<td>3</td>
<td>0</td>
<td>3</td>
<td>9</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>( k = 7 )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>3</td>
<td>7</td>
<td>9</td>
<td>9</td>
<td>0</td>
<td>0</td>
<td>34</td>
<td>0</td>
<td>0</td>
<td>26</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>7</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>38</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>26</td>
<td>19</td>
<td>50</td>
<td>4</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>23</td>
<td>29</td>
<td>60</td>
<td>8</td>
<td>2</td>
<td>0</td>
<td>10</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>6</td>
<td>9</td>
<td>5</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>32</td>
<td>0</td>
<td>1</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>11</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>8</td>
<td>0</td>
<td>0</td>
<td>55</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>13</td>
<td>17</td>
<td>13</td>
<td>15</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>73</td>
<td>0</td>
<td>1</td>
<td>6</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>
6. Discussion

Departing from the previous work on this topic [16], it is clear that the present methodology uncovers a path to employ Machine Learning techniques to improve the Schedule Coverage. The framework supports the employment of data from multiple routes in simultaneous by employing a consensual clustering technique [19]. This ability is key because it allows to express the network’s behavior – which would be hardly done using just a single route.

The resulting clusters helped to evaluate whether the current day coverage is the most suitable according to the daily profile of the routes in this case study. This methodology does not depend on the number of \( k \) schedules previously defined,

![Consensual Schedule Coverage](image)

**Fig. 5.** Consensual Schedule Coverage proposed for \( k = 4 \) along the months of the year. The point colors represent their months. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

**Fig. 6.** Illustration of the rule sets obtained from three consensual Schedule Coverages \( k = \{2, 3, 4\} \) as decision trees.
as can easily be observed by the variation of this parameter in Table 2. It can be applied to any public transportation network, even if the study presented here is considering a case study where only one company is running. For that, it is only necessary to deploy a bus dispatch system whose fleet is equipped with a communication system capable of automatically transmitting (with a certain but short periodicity) the vehicle’s position (in GPS coordinates) associated with a timestamp (also known as AVL system).

The coverage in place is already constraining the daily profiles – as they force the trips to have the same timetables in the days covered by each schedule and yet, different timetables from schedule to schedule (i.e. longer travel times between schedule points, distinct slack times, etc.). Consequently, it would be expectable to obtain slight differences on such profiles – namely, just on a few days. However, by observing Table 2, it is possible to state that the results suggest large-scale differences from the Schedule Coverage in place (please see Fig. 3). Such differences are not isolated, as they usually correspond to a given pattern (i.e. Saturdays are similar to Sundays). Such patterns highlight how valuable this Machine Learning framework can be. Moreover, their meaning is strengthened by the surprising differences discovered between the uncovered Schedule Coverage and its initial definition.

The amount of data used to conduct this study – one year – may not appear to be sufficient to consider all extracted patterns meaningful. The setting of each Schedule Coverage, like many other transportation variables, concern some seasonalties which size may vary from Schedule to Schedule. A good example on this can be a Schedule covering the Saturdays/Sundays – which will follow weekly patterns. On the other hand, a Schedule containing all the workdays in October–December will result from trimester-based pattern. An year contain many weeks but just a few trimesters (i.e. four). While such amount of data may be sufficient to validate an weekly pattern, it may overfit patterns over larger time spans (such as the trimester ones). Consequently, it would be desirable to test this framework using a larger amount of data (e.g. two years or more). Nevertheless, the abovementioned findings represent a fair number of opportunities to reduce the company’s costs by, for instance, reducing the number of trips in Saturdays – as we will set the same timetable for all the weekends. Such opportunities are worthy to be explored per se. Consequently, the authors consider the amount of data employed as sufficient to this particular end – even if we state that a larger training dataset would be desirable.

Its ability to illustrate the similarities between the daily profiles in different routes (even where each route provides heterogeneous insights) is key, especially if we consider that such results already depend highly on the SP (number, coverage and timetables) already in place.

There is a main pattern common to almost every number of schedule k considered, which is depicted in Table 2: Saturdays and Sundays should use the same timetable. All the obtained Schedule Coverages (but one, i.e. k = 5) contain a schedule/cluster similar to the cluster 2 in Fig. 5 (note the red border surrounding it). The existence of this very same cluster on each one of the proposed Schedule Coverage highlight how consensual are the different partitionings on the patterns followed by the daily profiles during the weekends. Such rule may reduce the number of necessary resources since the Saturdays used their own timetable in our case study (see Fig. 5 – namely the number of driver shifts, driving hours and/or the necessary vehicles.

A relevant but distinct pattern is observable in Fig. 6: the working days in the School Period during the months of September, October, November and December must be put on an individual schedule. This difference is even more visible when we consider the same k = 4 schedules that are currently in place in Porto: it is possible to observe a clear difference between clusters one and three in Fig. 5. Such difference occurs due to a change in the variability of the round-trip times observed on the days belonging to each one of these clusters. This variability corresponds to an absolute variance – which is not numerically comparable between clusters since it may refers to different populations (i.e. routes).

However, the authors claim that the days in distinct clusters will have completely distinct daily profiles. Such differences correspond to round-trip times larger than the usual in some periods of the day. This phenomenon may be explained by the weather conditions in the city of Porto during this period, where storms are frequent, or by some unexpected event, such as long term work on an important city road. However, it is not possible to determine that for sure and the reasons behind this difference are not addressed in this work.

The rules learned can cover the majority of the days considered (≥ 77%), thus demonstrating its capacity to turn the Schedule Coverage obtained into easy-to-read information that cover almost the entire partitioning found by the consensual clustering previously applied. The results using the default parameter values for RIPPER were satisfactory on this dataset but its optimal setting may vary to each case study.

To use 2 folds is to determine that 50% of the data will be used on the pruning stage. In this specific application, the pruning stage may be important to reduce sample overfitting – which would be manifested by misclassification rules which are just covered by a few number of days (i.e. to enlarge the Non Scholar Period Coverage or the Christmas for more time than it would be desirable). It is advisable to maintain this stage whenever we are expecting to obtain Schedules with a low coverage (< 15 days). Otherwise, it may be better to omit the pruning stage altogether. Another option could be to increase this number and, consequently, to reduce the data used in this stage. Yet, the reduced number of attributes employed (six) disallows such option as the branches removed based on a shorter amount of data may be highly relevant to the remain folds of this stage. Yet, the determination of an optimal number of folds to this problem is still an open research question and it may be necessary to employ some method (i.e. wrappers [13]) to automatize such parameter tuning. The optimization stage is not that impactful but it should not be disregarded. It consists into trying to find alternatives to the existent rule set by repeating the rule grow/prune stages. It may comprise slight improvements on the accuracy in exchange on some computational effort (which will grow along with the parameter value). Yet, if there are no alternatives to the discovered rules, it may be useless to increase such number.
By the abovementioned reasons, the 23% of information lost by the rule induction process from the clustering one could be partially recovered by (a) omitting the pruning stage or by (b) increasing the number of optimizations. However, it is not possible to confirm such hypothesis using the present experimental set. Nonetheless, the authors want to highlight that the optimization of the \textit{J-RIP} classifier accuracy was not in the scope of this paper – the emphasis is on the rules produced by the RIPPER. The goal is to prove that this rule learner is able to extract rules (which highlight the patterns underlying on our data) for those who are not familiar with Machine Learning techniques.

Departing by the conclusions obtained along this section, it is possible to confirm the importance of this tool as it provides useful insights on the coverage of the bus schedule. In fact, this framework can find rules that cannot be discovered using the evaluation methods already described in the literature. Such insights can be used to produce a new Schedule Coverage capable of reducing the variability observed between the real and the scheduled round-trip times. A possible proposal to do that in this specific case study is described below.

6.1. A schedule coverage proposal

From the analysis of the consensus clusters, five main constraints to our Schedule Coverage can be drawn:

1. The working days should be in a schedule separated from the remaining days (as suggested by Table 2 where this type of days is commonly grouped in schedules that are different from the weekends and/or the holidays one).
2. The working days in a school holiday period should be in an individual schedule (check the values in bold in the NSW column in Table 2 for \( k \geq 3 \) to see some examples of this pattern).
3. The weekends and the holidays should be in an individual schedule (a good illustration of these is made by cluster 2 in Fig. 5 or in Table 2 – especially for \( k < 5 \)).
4. The CT could be in the same schedule as the weekends and the holidays (typically Christmas Time days, represented by the CHR column in Table 2 are grouped with the weekend days or holidays).
5. There is a clear difference between the working days in the last four months and those in the remaining months (visible in clusters 1 and 3 of Fig. 5).

Following these constraints, several hypotheses can be made to re-arrange the Schedule Coverage on this case study. However, they must meet other operational planning constraints such as the number of drivers/vehicles available and their shifts [3]. For more information on this topic, the reader can consult the following survey on urban planning for public transportation companies [29].

A possible new Schedule Coverage – according to the current number of schedules – could be the following:

- **Schedule 1** working days from January 1st to July 15th (beginning of the school holiday period).
- **Schedule 2** working days from July 15th to September 15th (school holiday period).
- **Schedule 3** working days from September 15th to December 31st.
- **Schedule 4** all non-working days including all holidays and weekends.

6.2. Potential infusion and impact

The use of the proposed framework depends on the perception that transportation planners have about its usefulness. In this section, the main issues on evaluating the changes to the existing SP coverage are described. Then, an approach to measure the usefulness of such framework in a way that could be understood by the planners is presented.

The main obstacle to perform such evaluation is the inexistence of data obtained with both the current and the new Schedule Plans. In our case study, evaluating a new SP coverage is hardly done before deployment [18]. The main reason for that is that by using a different coverage, various schedules should be used in order to better adjust the schedules to traffic in different days. Despite this difficulty, the proposed approach must be evaluated prior to deployment.

Reducing the variance of round-trip times originated by the same scheduled trip has a potential impact on three different components of revenue and costs for a bus company: (1) the revenue can be increased, (2) and the budgeted costs and (3) non-budgeted costs can be reduced.

The first component can happen when there is an increase in client satisfaction as a consequence of the perceived increase in service quality. Measuring such impact is very difficult without generating data based on the new SP. However, it is easier to define a scheduled round-trip time (TT) that is more adjusted to the actual TT when the variance is lower. The method used reduces the variance inside the groups. For this reason, it is expected that the new schedules will improve the users’ perception of service quality.

The second component is probably the easiest to estimate. In fact, when the variance of TT inside the groups reduces, it is possible to reduce slack times, which have an impact on the definition of crew services, increasing the percentage of driving time in these services. The average cost of the drivers per minute is an important key performance indicator for a public transport company and can be used to estimate the reduction of budgeted costs caused by reducing time \( \times \) drivers. This cost is the most important of the budgeted operational costs. It should be emphasized that a small reduction in slack times can cause an important decrease in operational costs due to the increase of the averaged travel time per driver duty.
The third component occurs when it is necessary to adopt extra measures. This happens when there are disruptions to the actual and the scheduled service. The operational planners can create a tighter or wider schedule. In the first case, slack times will be shorter but the probability of disruption increases, thus increasing the non-budgeted operational costs. In the second case, the slack time will be larger, reducing the probability of disruption and, consequently, reducing the non-budgeted operational costs, and yet increasing the budgeted component of the cost. By reducing the variance inside the groups, and maintaining the same probability of disruption, budgeted costs must necessarily fall.

In this case, the method proposed was evaluated by estimating the variance of the trips. This was performed by grouping the trips by route, schedule and scheduled trip start time, and by calculating the sample variance for each group. Then, the global variation was calculated for both the current coverage and the coverage proposed here using the weighted average of the variances in all groups previously described. This weighted average reduces the degrees of freedom in each variance used, as explained in any introductory book on statistics. Comparing the sample standard deviations, the results are inconclusive (present coverage: 594.3 s; proposed coverage: 607.8 s). It is important to emphasize that these results are necessarily biased by the use of trips generated using the current coverage. Indeed, in such conditions, it is natural that the generated trips are more adapted to the current schedule. This is particularly true for circular routes, as it is the case of lines 300 and 301. However, this result does not invalidate the reasonableness of such approach. Its evaluation after deployment, even if in a controlled way (for instance using only a small number of routes) would be particularly important. This work is now ready to be used. However, its usefulness for a bus company depends on its ability to cover, at least, all functionalities when creating and maintaining timetables. Moreover, since the definition of timetables has an impact on the remaining steps of operational planning (as described in Section 1), this kind of software will be especially interesting when included in Decision Support Systems that cover all steps of operational planning.

7. Conclusions

Most classical approaches to schedule evaluation rely only on how to change the defined timetables and driver shifts. However, these definitions are based on previous definitions which could not be evaluated using an automatic algorithm. Additionally, such changes usually represent an increase in operational costs, for instance, due to increases in the number of running vehicles, slack times and/or driver shifts.

To the authors’ best knowledge, this is the very first framework capable of evaluating whether the current coverage fits the network needs. The insights hereby discovered will enhance the operational planning tasks by providing novel decision variables to the planners. These variables carry information that can cause an impact on planning: by optimizing the Schedule Coverage, the planners will be able to take full advantage of the existing resources, or even reduce related costs, while improving the passengers’ perception of service reliability and providing SP day coverage according to their mobility needs.

This problem was addressed using a reasonably complex Machine Learning system. The steps used were: (1) \( k \)-Means with the DTW distance per route to find an optimal Schedule Coverage for each route based on its trip daily profiles, (2) a consensual clustering to find a consensual day partition between all the considered routes, and (3) rule induction using the RIPPER algorithm to extract rules. The use of consensual clustering is emphasized to address an important real problem in the transportation area. The employment of the rule induction system broadens the target audience for this methodology by removing the need for a solid background on Machine Learning techniques.

The experiments were conducted in a specific case study, a public transportation operator in Porto, Portugal, which highlighted the usefulness of this framework: it is capable of extracting important information regarding the Schedule Coverage from a vast amount of data. It is independent of the number of schedules \( k \) and, more importantly, of the company where the framework will be deployed (only an AVL communication system is required). The authors of this paper believe that the work presented in this paper is unique due to the type of patterns it can reveal about the Schedule Coverage. Moreover, it opens new research lines for evaluating SP by broadening its scope to the coverage dimension.

Acknowledgements

This work was fully supported by project KDUS – Knowledge Discovery from Ubiquitous Data Streams (PTDC/EIA–EIA/098355/2008) and by the ERDF – European Regional Development Fund, through the COMPETE Program (operational program for competitiveness), as well as by Portuguese Funds provided by the FCT (Portuguese Foundation for Science and Technology) as part of project FCOMP-01-0124-FEDER-037281.

We would also like to thank the STCP (Sociedade de Transportes Coletivos do Porto) for the data provided for this study. Finally, the authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this work.

References