Bus Bunching Detection by Mining Sequences of Headway Deviations

Luís Moreira-Matias[1,2], Carlos Ferreira[2,3], João Gama[2,5], João Mendes-Moreira[1,2], [4]Jorge Freire de Sousa

[1] Departamento de Engenharia Informática, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, s/n 4200-465 Porto – Portugal

[2] LIAAD-INESC Porto L.A. Rua de Ceuta, 118, 6º; 4050-190 Porto – Portugal

[3] Instituto Superior de Engenharia do Porto, Instituto Politécnico do Porto, Rua Dr. António Bernardino de Almeida, 431, 4200-072 Porto

[4] Departamento de Engenharia Industrial e Gestão, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, s/n 4200-465 Porto – Portugal

[5] Faculdade de Economia, Universidade do Porto

Rua Dr. Roberto Frias, s/n 4200-465 Porto - Portugal

[1,4] [luis.matias; jmoreira; jfsousa @fe.up.pt]; [3] cgf@isep.ipp.pt; [5] jgama@fep.up.pt;

Abstract. In highly populated urban zones, it is common to notice headway deviations (HD) between pairs of buses. When these events occur in a bus stop, they often cause bus bunching (BB) in the following bus stops. Several proposals have been suggested to mitigate this problem. In this paper, we propose to find BBS (Bunching Black Spots) – sequences of bus stops where systematic HD events cause the formation of BB. We run a sequence mining algorithm, named PrefixSpan, to find interesting events available in time series. We prove that we can accurately model the BB trip usual pattern like a frequent sequence mining problem. The subsequences proved to be a promising way of identify the route' schedule points to adjust in order to mitigate such events.

Keywords: Sequence Mining, Bus Bunching, Headway Irregularities

1 Introduction

In highly populated urban zones, it is well known that there is some schedule instability, especially in highly frequent routes (10 minutes or less) [1-5]. In this kind of routes it is more important the headway (time separation between vehicle arrivals or departures) regularity than the fulfillment of the arrival time at the bus stops [4]. Due to this high frequency, this kind of situations may force a bus platoon running over the same route. In fact, a small delay of a bus provokes the raising of the number of passengers in the next stop. This number increases the dwell time (time period

where the bus is stopped at a bus stop) and obviously also increases the bus's delay. On the other hand, the next bus will have fewer passengers, shorter dwell times with no delays. This will continue as a snow ball effect and, at a further point of that route, the two buses will meet at a bus stop, forming a platoon as it is illustrated in Fig. 1. This phenomenon has several denominations: the Bangkok effect [6], Bus Platooning [7], Vehicle Pairing [8], Headway Instability [1], Bus Clumping or Bus Bunching (BB) [9], [2]. From now on, we will use the last one.

The occurrence of BB forces the controllers to take actions in order to avoid this headway instability, forcing the adherence to the schedule. BB situations can cause several problems like: further buses delays, full buses, decreased comfort in the buses, larger waiting times at the bus stops, growing number of passengers waiting, greater resources demand and a decrease of schedule reliability. All this can cause the loss of passengers to other transportation means and/or companies.

Our goal is to identify the causes of BB occurrences using AVL (Automatic Vehicle Location) historical data. The BB phenomenon always starts by a headway deviation (HD) at a bus stop [10]. We intend to find frequent and systematic HD event sequences in the trips of a given route: bus stops where the bus activities - like the passenger boarding - will propagate the headway irregularities further and further. These bus stops sequences **highlights problematic route regions**: from now on we will refer to it as **Bunching Black Spots** (BBS - bus stops sequences where a HD will, with a high probability, start a BB in one of the following bus stops of the trip).

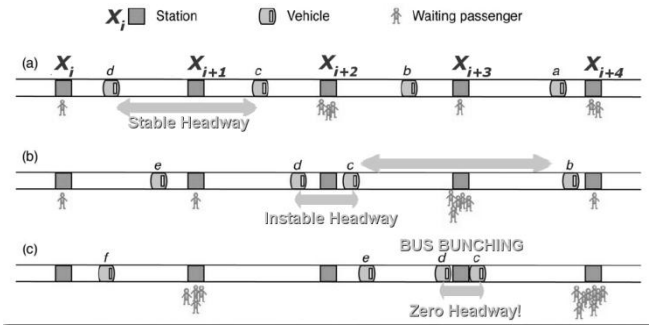

Fig. 1. Bus Bunching problem illustration. Figure based on Fig.1 from [1].

We use the PrefixSpan algorithm (presented in Section 3) to mine frequent sequences in the HD sequences extracted from this dataset. We apply this methodology to data from two urban lines of a public transport operator of Porto. It proved to be efficient in the detection of HD patterns in the bus stops of the studied routes.

The results from this framework can be highly useful to the public transport planners. One of the most known ways to mitigate the bus bunching is to adjust the slack time introduced in each schedule point (bus stops selected along the route for which the arrival time is defined) [11]. By using this framework, the planners can use the information about the BBS along the routes to select which schedule points should be changed (increasing or decreasing the slack time) to mitigate BB effectively.

The main results are: the observation that the BB phenomenon starts at the initial bus stops; and the existence of high correlation between HD that occurs at a given bus stop and the HD detected in the next ones.
This paper is structured as follows. Section 2 states a brief description of the problem we want to solve, the related work, our motivation and a clear definition of our approach. Section 3 presents the methodology proposed. Section 4 presents summarily the dataset used, its main characteristics and some statistics about it. Section 5 presents the results obtained through the application of the PrefixSpan algorithm to our dataset and a discussion about those results. Section 6 concludes and describes the future work we intend to carry on.

2 Problem Overview

Nowadays, the road public transportation (PT) companies face a huge competition of other companies or even of other transportation means like the trains, the light trams or the private ones. The service reliability is a fundamental metric to win this *race* [12]: if a passenger *knows* that a bus of a selected company will arrive *certainly* on the schedule on his bus stop, he will probably pick it often. The reverse effect is also demonstrated and a BB event forming a visual bus pair is a strong bad reliability signal to the passengers' perception of the service quality, which can lead to important profit losses [9, 13]. This tendency to form platoons is usual for urban vehicles (specially the PT ones) and arises for the specific and complex characteristics of transit service perturbations. Those are mainly related with changes in three key factors [8]: the dwell time and the loading time (highly correlated) and the non-casual passenger arriving (passengers that, for an unexpected reason – like a soccer match or a local holiday - try to board in a specific bus stop distinct from the usual one). However, the study of these changes impact on the service reliability is not in our current scope. Our goal is to find persistent and frequent headway irregularities which will *probably* provoke, in a short time horizon, a BB event.
There are two distinct approaches found in the literature to handle the BB events: the first one defines the bunching problem as a secondary effect of a traffic system malfunction like a traffic/logistic problem (signal priority handling, adaptation of bus stops/hubs logistics to the needs, adjustments of the bus routes to the passengers demand, etc.). The second one defines the BB problem like a main one that must be treated and solved *per se* (adjust the timetables and the schedule plans to improve schedules' reliability or set live actions to the irregular bus pairs, for instance).
In this work, we are just focused on the second approach which related work, motivation and scope we present along this section.

2.1 Related Work

There are two distinct approaches to mitigate BB: (1) the PT planning one, where they try to adjust the schedule plans somehow and the control one, where the BB is avoided by actions suggested live by the controllers and (2) the real-time approaches, which use streaming data to evaluate the network and to choose some actions to keep the system stable. To do so, it is suggested one or more actions to the irregular (i.e. schedule behind or ahead) buses. There are four types of actions that can be proposed

to avoid BB in real time: the change in bus holding time, the stop-skipping, the preplanning deadheading (the scheduling of some vehicles to run empty through a number of stations at the beginning or the end of their routes) and the change in the bus cruise speed.
We can split the existing experimental setups to test and evaluate such approaches in two big groups: the first one uses simulation models and the newer one's uses AVL historical data to test their approaches. A brief state-of-art on both is presented below.

Simulation Models

Newell *et. al* presented one of the first known models to reduce BB [14]: an optimization framework to control the headway deviation effects. Basically, it consists in the simulation of two buses and one control point. The simulation was run assuming ideal conditions and it consists in the introduction of delay in one of the buses using stochastic variables. The simulation tested control metrics to force the headway to remain stable.
Public transportation companies use slack times in the building of their schedule plans in order to avoid that delays in a given trip force delays in the departure of the next trip. This is a common practice in order to guarantee passengers' satisfaction by increasing schedules reliability. An important definition is presented by Zhao *et al.* in [11]: "*an optimal slack time will correspond to the best schedule plan possible. This plan should avoid BB situations*". They present a method to obtain the optimal slack times for a given number of vehicles on highly frequent routes.
One of the first probabilistic model to predict BB [15] defines a distribution along a given line to evaluate the tendency of buses to form pairs as they progress down their route. Other works present models like this one. One of them [16] uses the Monte Carlo theorem to introduce stochastic variations to the traffic conditions, namely, the bus speed between stops. Usually these works consider classical variables of public transportation planning like the bus speed between bus stops, passengers boarding time, headway, among others, to suggest forced actions to detect BB in a simulation. These two works suggest one or two types of forced actions to maintain stability in the simulation after the launch of a BB trigger.
Gershenson *et. al.* presented a model adapted from a metro-like system and implemented a multi-agent simulation [1]. To achieve stability, they implemented adaptive strategies where the parameters are decided by the system itself, depending on the passenger density. As a result, the system puts a restriction to the vehicle holding time (it sets a maximum dwell time), negotiating this value for each bus stop with the other vehicles.

Real Data (AVL) Models

The introduction of AVL systems changed the research point-of-view on bus bunching, in the last ten years, from planning to control. There are several techniques in PT to improve the schedule plans on time tables based on AVL data. An useful review on those is presented by Peter Furth in [17].
C. Daganzo presents a dynamic holding time formulae based on real time AVL data in order to adaptively compensate the headway instability introduced in the system [2].

There are as well bus cruising speed approaches. In [3] it is presented a model allowing the buses to negotiate an ideal cruising speed to avoid potential BB situations.

Headway Irregularities on AVL-based models

The relations between the irregularities in the headway sequences and the BB events have been recently explored: in [8] is presented a study identifying the headway distributions representing service perturbations based on probability density functions (p.d.f.). This study was done using a stochastic simulation model for a one-way transit line accounting several characteristics like the dwell time or the arrivals during the dwell time (which values for each bus stops were calculated using the pre-calculated p.d.f.). Despite their useful conclusions, their model had two main disadvantages: 1) is not based in real AVL data and 2) it does not present a probability density function to represent the pattern of consecutive headways irregularities. We do believe that this specific issue can be rather addressed mining frequent sequences on real AVL data, as we present here.

2.2 Motivation and Scope

We can define the headway irregularities as events that occur in a bus stop of a given trip. Those events consist in a large variation (1 for positive or -1 for negative) on the headway: Headway Deviation events (HD).
These are usually correlated in a snowball effect that may occur (or not) in a given (straight or spaced) sequence of bus stops. Despite the analysis of the state-of-art work on the mitigation of BB events, the authors found no work on systematizing real HD patterns that seem to be in the genesis of a BB event.
An unreliable timetable is one of the main causes of many HD events. Usually, a timetable is defined using schedule points: stops for which there is an arriving or departing time defined. One of the most well-known PT planning ways to mitigate HD events is to add/reduce slack time in these defined timestamps to increase schedule plan overall reliability. However, only a small percentage of the bus stops served by a given timetable are used as schedule points. This is exemplified in the upper part of Fig. 2 (the reader can obtain further details on schedule plan building in chapter 1 from [18]). Usually, PT planners easily identify which lines present more HD and BB events. However, three questions still remain open:

1) Which should be the schedule points affected?
2) Which action (increase/decrease slack time) should be applied to these schedule points in order to reduce the occurrence probability of BB events?
3) Which day periods should have the timestamps in these schedule points changed?

In this work, we address the first and third questions by mining frequent HD event sequences in the trips of a given route: bus stops that systematically propagate the headway irregularities further and further. The second issue is out of our scope but it is well addressed in the literature [11].

Our intention is to point out a route region where an HD event fast and systematically propagates itself along the route, forming a Bunching Black Spot (BBS). The BBS can be specific of a period of the day or continuous along the day. In the bottom part of Fig. 2 we present an example of a BBS. In the next section we present our methodology to mine BBS.

3 Methodology

Our methodology consists in finding consistent patterns of frequent HD events occurring in the same bus stops whenever a BB occurs – BBS. To do so we compare, at each bus stop, the round-trip times of every consecutive bus pairs. With the HD series thus obtained, we mine frequent sequence patterns. Firstly, we introduce the algorithm we used and finally we describe how we use it to create and mine our HD series for a given route.

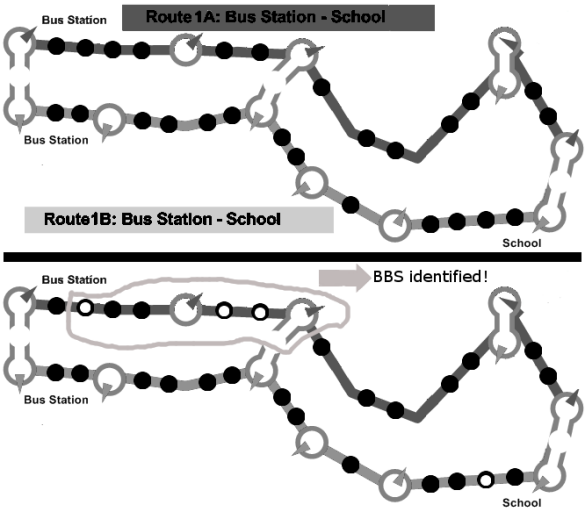

Fig. 2. Example of Schedule Points and BBS. The two schemas exemplify two routes of a line running between an arbitrary school and a main bus station. In top part, route 1A has 19 bus stops represented by 13 small black circles and 6 big grey circles (the single one's are just bus stops, the double are hubs/interfaces). The last ones are the schedule points in the route's timetables. In the bottom part, the stops belonging to frequent HD sequences are identified (even if the BB itself occurs later in the route) with a small white circle inside them. The highlighted stops form a route region (Bunching Black Spot) where the schedule points need to be time-adjusted.

3.1 Mining Time Series Sequences

There is a wide range of algorithms that can explore sequential data efficiently. To the best of our knowledge, Agrawal and Srikant introduced the sequential data mining problem in [19]. Let $\mathbf{I} = \{\mathbf{i}_1, \mathbf{i}_2, .., \mathbf{i}_n\}$ be a set of items and $\mathbf{e}$ an event such that $\mathbf{e} \subseteq \mathbf{I}$. A sequence is an ordered list of events $\mathbf{e}_1\mathbf{e}_2 \ldots \mathbf{e}_m$ where each $\mathbf{e}_i \subseteq \mathbf{I}$.

Given two sequences $\boldsymbol{\alpha}=\mathbf{a_1a}_2\ldots\mathbf{a}_r$ and $\boldsymbol{\beta}=\mathbf{b}_1\ \mathbf{b}_2\ \ldots.\ \mathbf{b}_s$, sequence $\boldsymbol{\alpha}$ is called a subsequence of $\boldsymbol{\beta}$ if there exists integers $1 \leq\ \mathbf{j}_1 < \mathbf{j}_2 < \ldots < \mathbf{j}_r \leq s$ such that $\mathbf{a}_1 \subseteq \mathbf{b}_{j1}$, $\mathbf{a}_2 \subseteq \mathbf{b}_{j2}$, … ,$\mathbf{a}_r \subseteq \mathbf{b}_{jr}$. A sequence database is a set of tuples (**sid**, $\boldsymbol{\alpha}$) where **sid** is the sequence identification and $\boldsymbol{\alpha}$ is a sequence. The count of a sequence $\boldsymbol{\alpha}$ in **D**, denoted **count**($\boldsymbol{\alpha}$, **D**), is the number of sequences in **D** containing the $\boldsymbol{\alpha}$ subsequence.

The support of a sequence $\boldsymbol{\alpha}$ is the ratio between **count**($\boldsymbol{\alpha}$, **D**) and the number of sequences in **D.** We denote sequence support as **support**($\boldsymbol{\alpha}$, **D**). Given a sequence database **D** and a minimum support value $\boldsymbol{\lambda}$**,** the problem of sequence mining is to find all subsequences in **D** having a support value equal or higher than the $\boldsymbol{\lambda}$ value. Each one of the obtained sequences is also known as a frequent sequence.

In [20] the GSP algorithm, an algorithm that generalizes the original sequential pattern mining problem, is introduced. The search procedure of this algorithm is inspired by the well-known APRIORI algorithm [21]. GSP uses a candidate-generation strategy to find all frequent sequences, and uses a lattice to generate all candidate sequences. We observe that GSP has limitations when dealing with large datasets because candidate generation may require multiple database queries.

Several approaches have been proposed to address the above mentioned issue. One of the most interesting and efficient proposals is PrefixSpan algorithm [22]. This algorithm makes use of pattern-growth strategies to efficiently find the complete set of frequent sequences. The algorithm starts by finding all frequent items (length one sequences). Then, for each one of these frequent items (the prefix) PrefixSpan partitions the current database into *prefix projections*. Each projection database contains all the sequences with the given prefix. This procedure runs recursively until all frequent sequences are found.

In this work we run PrefixSpan algorithm to solve our problem due to its popularity and efficiency.

3.2 Methodology

Firstly we constructed headway sequences based in the AVL historic data for every bus pairs in a given route. Then we identified the headway profiles where BB events occurred based on the bus service reliability metrics presented in [23] and we extracted HD sequences from them.

Let $X = x_1x_2\ldots x_n$ be a headway sequence measured between a bus pair in a given route through n bus stops running with a frequency f $(f = 1/x_1)$. We identify a BB if exists a x_i satisfying the inequality $x_i \leq (0.25 * 1/f)$ for at least one $i \in \{1, \ldots, n\}$. An example of this analysis is shown in Fig. 3 and in Fig. 4, where we identified 4 BB events. Based on this headway profiles, we formed a HD sequence as follows. Let $H = h_1h_2\ldots h_n$ be the HD sequences based on X. We compute the value of each h_i (the headway between a bus pair in the bus stop x_i), for each $i \in \{2, \ldots, n\}$, using the expression 1.

$$\boldsymbol{h_i} = \begin{cases} \mathbf{0}\ \boldsymbol{if} |\boldsymbol{x_i} - \boldsymbol{x_{i-1}}| < \left(\frac{\mathbf{1}}{\boldsymbol{f}}\right) * \boldsymbol{ht} \\ \mathbf{1}\ \boldsymbol{if}\ \boldsymbol{x_i} - \boldsymbol{x_{i-1}} \geq \left(\frac{\mathbf{1}}{\boldsymbol{f}}\right) * \boldsymbol{ht} \\ -\mathbf{1}\ \boldsymbol{if}\ \boldsymbol{x_i} - \boldsymbol{x_{i-1}} \leq -\left(\frac{\mathbf{1}}{\boldsymbol{f}}\right) * \boldsymbol{ht} \end{cases} \quad (1)$$

where ht is a threshold parameter given by the user for the HD definition. For the first bus stop is considered an HD of 0. Basically, a -1 event corresponds to a negative HD (delay) in a bus stop (i.e.: the two buses become closer), the 1 event is a positive HD (ahead of schedule) and the 0 occurs when the headway remains stable.

The x_n represents a headway deviation in a bus stop *n*. The HD sequences are ordered according to the bus stop order defined for a given route. Our goal is to find sequences of bus stops with frequent HD by exploring a set of trips, in a given route, where BB occurrences were identified.

To do so, we collected the HD sequences of trips in work days where a BB event occurred and we mined them using the PrefixSpan algorithm by setting a (user-defined) minimum support value in order to identify HD patterns in the bus stops. Fig. 5 illustrates our methodology. We applied this methodology to four routes in a given period. This data is summarily described in Section 4.

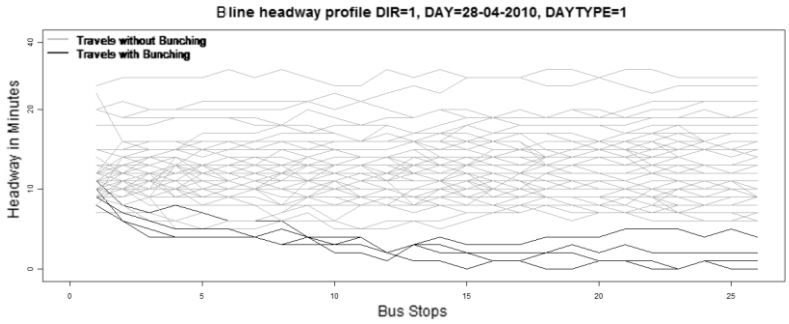

Fig. 3. Headway profiles of the route B1 for a given day. There were four BB events identified.

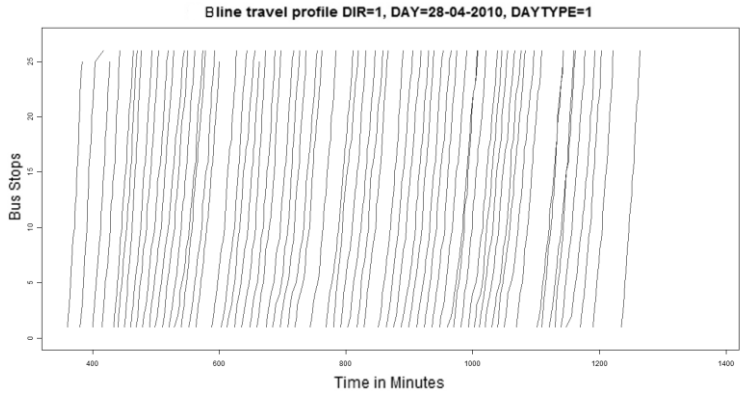

Fig. 4. Travel time profiles for the same day of Fig. 3. It is possible to identify the bunching situations directly.

4 Dataset

The source of this data was STCP, the Public Transport Operator of Porto, Portugal. The dataset was obtained through a bus dispatch system that integrates an Automatic Vehicle Location (AVL) system. The data captured through this system contains data of the trips from two lines (A and B) in the working days for the first ten months of

2010. Each line has two routes – one for each way {A1, A2, B1, B2}. Line B is a common urban line between *Viso* (an important neighborhood in Porto) passing by 26 bus stops (BS1_B1 to BS26_B1 and BS1_B2 to BS26_B2, respectively), and ending at *Sá da Bandeira*, a downtown bus hub. Line A is also an urban line between another downtown bus hub (*Cordoaria*) and *Hospital São João* - an important bus/light train interface in the city – using 22 bus stops (same schema than line B).This dataset has one entry for each stop made by a bus running in the route during that period. It has associated a timestamp and a day type (1 for work days, 2-6 for other day types i.e.: holidays and weekends). Table 1 presents some statistics about the set of trips per route considered and the BB events identified. The *Nr. of Trips* is the total number of trips considered in the given route, TT is the round-trip time, expressed in minutes, and DT is the number of daily trips occurred. Finally, trips with BB are the trips where at least one BB situation occurs and HD events are the positive or negative events ($h_i = 1$ or $h_i = -1$, respectively) measured in every bus stops along every trip for a given line.

Table 1. Descriptive statistics for each route considered. These times are in minutes. TT means round-trip times. DT means daily trips. Based in our HD event definition, the maximum number of events for a time period is given as $Nr.of\ Bus\ Stops * Nr.Of\ Trips$.

	B1	B2	A1	A2
Nr. of Trips	9391	10675	13802	12753
Nr. of Bus Stops	26	26	22	22
Minimum TT	11	11	11	11
Maximum TT	78	82	70	65
Minimum of DT	39	39	33	36
Maximum of DT	74	74	89	88
Median TT	29	21	21	38
Nr. of Bus Stops	26	26	22	22
Nr. of Trips w/ BB	332	378	559	630
Nr. of HD events detected	26905	29911	42803	43525

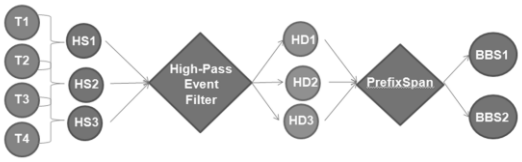

Fig. 5. Bunching Black Spot Detection Methodology illustration. *Tn* is the time series measured in each bus stop of a given trip. *HS* are the corresponding Headway Sequences and *HD* the Headway Deviation event subsequences.

5 Results

We did our experiments only for the trips occurred during the peak periods (08:00 to 11:00 and 16:00 to 19:00). We did so because BB mainly occurred – as expected – during those periods, as can be seen in Fig. 6. The routes A1 and A2 *suffer* more BB events and they are time-dispersed along the day. This happens because this line is an urban one between two important bus/metro interfaces (the downtown and the University Campus) with regular high frequencies during the entire day. So, they are highly frequent routes with many passengers during the entire day, which are well

known factors to provoke BB occurrences. We mined sequences just in the bunching partition (trips with BB events). Moreover, we use the two partitions to compute the confidence of each sequence to be specific on the BB one. Our goal was to find patterns (i.e. frequent HD sequences) describing the headway irregular behavior of a typical BB trip in a given route.

We did two different experiments: the first one mined sequences in both peak hours simultaneously; the second one mined each peak hour considered individually (the morning and the evening ones). We did so to mine BBS peak-dependent (just occur in one of the two peaks), discovering whether the schedule points should be adjusted for the entire day or just in a specific period.

The results presented in Table 2 are for frequent subsequences of the HD sequences. We set PrefixSpan minimum support to 40% (sequences of length=1) and 20% (sequences with a length greater than 1) in the selected data partition, and a *ht*=0.15. We did so because the significance of the second case is higher than the first one. The second case demonstrates high correlations between distinct HD events in distinct bus stops that explain better the origin of the BB events.

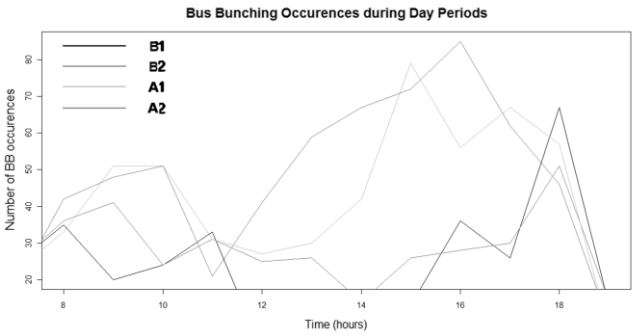

Fig. 6. Bus Bunching Occurences during Day Periods. The trips with occurences starting during the defined peak hours: 08:00-11:00 and 16:00-19:00.

Table 2. The values presented are the Support of the sequences (number of trips where those events occur / total number of BB trips considered) as well as the confidence between the occurrences of those in the trips with BB and the total trips occurred in the period.

ID	Route	Peaks Considered	Sequence (possible BBS)	Support	Confidence
01	**B1**	**Both**	**BS3_B1 = -1 BS4_B1=-1**	**0,2619**	**0,75**
04	**B1**	Both	BS2_B1 = -1	0,4206	0,80
05	**A1**	Both	BS2_A1 = -1	0,5095	0,72
06	**A2**	Both	BS2_A2 = -1	0,5706	0,61
07	**B1**	8h to 11h	BS5_B1 = -1	0,4000	0,91
08	**B1**	8h to 11h	BS2_B1 = -1	0,4308	0,85
09	**A1**	8h to 11h	BS6_A1 = -1	0,4064	0,88
10	**A1**	8h to 11h	BS3_A1 = -1	0,4225	0,87
11	**A1**	8h to 11h	BS2_A1 = -1	0,5669	0,72
12	**A2**	8h to 11h	BS2_A2 = -1	0,6237	0,74
13	**B1**	16h to 19h	BS2_B1 = -1	0,4099	0,82
14	**A1**	16h to 19h	BS2_A1 = -1	0,4500	0,81
15	**A2**	16h to 19h	BS2_A2 = -1	0,6237	0,78

5.1 Discussion

Firstly, we want to highlight that **only frequent HD subsequences (BBS) with events of type -1 (headway reductions) were detected**. All the sequences presents high confidence, demonstrating their specific validity in the bunching partition.
In route B1 two BBS were identified: BS2_B1 and the pair BS3_B1 and BS4_B1. Both are located at the beginning of the route: the gap verified in these points may become larger in successive stops. The pair is deeply analyzed in Table 3: the isolated events in BS3_B1 and BS4_B1 have the same support than the events occurred in both bus stops. We can also set an association rule like BS3_B1= -1 -> BS4_B1= -1 (with a confidence of 97%) identifying a solid BBS in those two bus stops and an expected BB behavior. In Fig. 7, we illustrate one example of the pattern extracted on a morning peak hour of a typical working day. Assuming casual and regular passengers arriving, we describe two cases: (1 - Non-Bunching) ideal case: bus pair running with a short but regular headway; (2 - Bunching) real case: another pair running in the route with an irregular headway, having a BB event in BS10_B1.
In line A, BS2_A1 and BS2_A2 were identified as BBS. Additionally, they are - as well as the BBS identified in line B – located in the beginning of the route. The causes for this behavior are, probably, the large affluence of passengers in peak hours but the authors cannot sustain this with the available data.
Summarily, just BBS for the first bus stops were found. Based on this, we can conclude that **the BB in those routes were largely provoked by successive bus delays in the first bus stops** (the HD -1 events are mainly caused by bus delays [8]) although we cannot sustain whether they are failing the schedule.
In the second study, we analyzed whether the BBS identified were coherent in both peak hours. In route B1, the BS2_B1 is a BBS for both peak hours.
BS2_A1 and BS2_A2 are also persistent BBS in both peak hours. Those two bus stops correspond to an important bus interface (*Sá da Bandeira*) in the city and to a University Campus (*Asprela*), respectively. This happens because both routes maintain a high frequency and a large number of passengers during the day, being always busy.
In our opinion, the short lengths of the frequent subsequences mined (1 and 2) are not relevant compared with the relevance of the identified patterns. Those lengths will always depend on the routes analyzed, so they can be larger when applied to other datasets. The achieved patterns demonstrate that the BB patterns can be modeled like a frequent sequence mining problem. The results achieved demonstrate the utility of our framework to identify the exact schedule points to change in the timetables.

6 Conclusions and Future Work

In public transportation planning, it is crucial to maintain the passengers' satisfaction as high as possible. A good way to do so is to prevent the phenomenon known as Bus Bunching.
There are two main approaches to handle this problem: the PT planning one, anticipating and identifying the origin of the problem, and a real time one, which tries to reduce the problem online (during the network function).

Table 3. Detailed analysis of the mined sequence BS3_B1 = -1, BS4_B1=-1. The support of the highlighted sequences 01a and 01b are the same of the sequence 01: this can demonstrate an implication between the bus delays in the BS3_B1 and BS4_B1, an usual BB behavior. The confidence for a possible association rule BS3_B1 = -1 -> BS4_B1=-1 is 97%.

ID	Route	Peaks Considered	Sequence (possible BBS)	Support
01	**B1**	**Both**	**BS3_B1 = -1 BS4_B1=-1**	**0,2619**
01a	**B1**	**Both**	**BS3_B1 = -1**	**0,2619**
01b	**B1**	**Both**	**BS4_B1 = -1**	**0,2619**

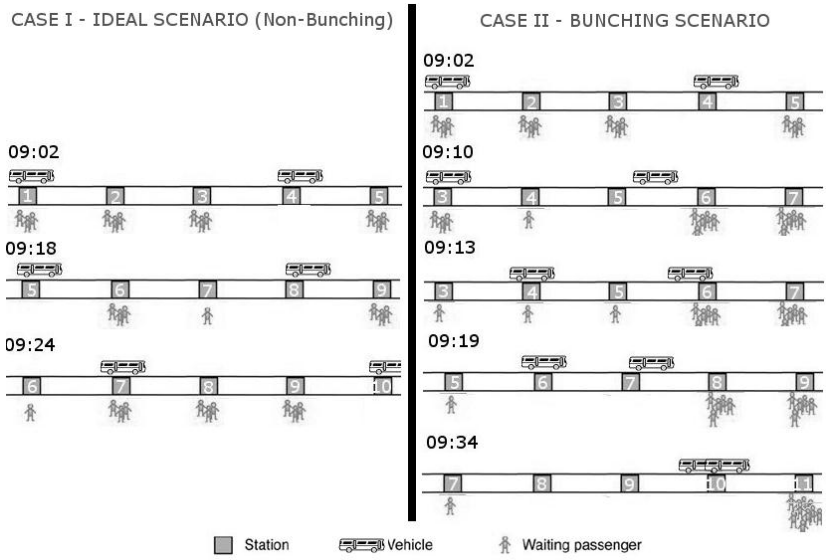

Fig. 7. Two possible cases in a Tuesday morning: one with BB and other without it. The numbers inside the squares are the bus stops' identifiers. The case II is one of the 28,6% of BB trips with the frequent subsequence 01 (see Table 2). The passengers in each stop are an estimation assuming casual passenger arriving [8]. It is possible to observe the strong effect of the first HDs on the number of passengers waiting in the following bus stops and, consequently, in the headways.

Our approach is a contribution to solve the PT planning problem: this framework can help to identify patterns of bus events from historical data to discover the schedule points to be adjusted in the timetables.

In this paper, we presented a methodology to identify BB events that use headway deviations from AVL trips data. We ran a sequence mining algorithm, the PrefixSpan, to explore such data.

The results are promising. We clearly demonstrated the existence of relevant patterns in the HD events of the travels with bunching. There were some bus stops sequences along the routes identified as BBS - Bunching Black Spots, forming regions within the schedule points that should be adjusted. We want to highlight the following findings:

- The high correlation between HD in distinct bus stops – one event in a given bus stop provoke an event on another one with a regularity sustained by a reasonable support and confidence;
- The detection of BBS in the beginning of the routes demonstrated that HD that occurs in the beginning of the trips can have a higher impact into the occurrence of BB compared with events occurred in bus stops further.

The main contributions of this work are: 1) to model the BB trip usual pattern like a frequent sequence mining problem; 2) to provide the operator the possibility to mitigate the BB in a given line by adjusting the timetables, instead of suggesting forced actions that can decrease schedule reliability and, consequently, reduce passengers' satisfaction.

The identified patterns are no more than alerts that suggest a systematic cause for the BB in the studied routes. This information can be used to improve the schedule. The goal is not to eliminate those events but just to mitigate them. Our future work consists in forecasting BB in a data stream environment based on AVL data. By using this approach, the BSS will be identified online as the data arrive in a continuous manner [24]. This possibility will allow the use of control actions to avoid BB events that can occur even when the timetables are well adjusted, in order to prevent the majority of the potential BB occurrences.

Acknowledgements

We would like to thank STCP (Sociedade de Transportes Colectivos do Porto, S.A.) for the AVL historical data supplied to this work. We would also like to thank the support of the project Knowledge Discovery from Ubiquitous Data Streams (PTDC /EIA-EIA/098355/2008).

References

1. Gershenson, C., Pineda, L.: Why Does Public Transport Not Arrive on Time? The Pervasiveness of Equal Headway Instability. PLoS ONE 4, (2009)
2. Daganzo, C.: A Headway-Based approach to eliminate Bus Bunching. Transportation Research Part B 43, 913-921 (2009)
3. Pilachowski, J.: An approach to reducing bus bunching., vol. PhD. Univ. of California, Berkeley, California (2009)
4. Lin, J., Ruan, M.: Probability-based bus headway regularity measure. IET intelligent transport systems 3, 400-408 (2009)
5. Matias, L., Gama, J., Mendes-Moreira, J., Sousa, J. F.: Validation of both number and coverage of bus Schedules using AVL data. . 13th International IEEE Annual Conference on Intelligent Transportation Systems, pp. 131-136, Funchal, Portugal (2010)
6. Newman, P.: Transit-Oriented Development: An Australian Overview. Transit Oriented Development – Making it Happen (2005)
7. Strathman, J., Kimpel, T., Callas, S.: Headway Deviation Effects on Bus Passenger Loads:Analysis of Tri-Met's Archived AVL-APC Data. (2003)

8. Bellei, G., Gkoumas, K.: Transit vehicles' headway distribution and service irregularity. Public Transport 2, 269-289 (2010)
9. Wang, F.: Toward Intelligent Transportation Systems for the 2008 Olympics. IEEE Intelligent Systems 18, 8-11 (2003)
10. Newell, G., Potts, R.: Maintaining a bus schedule. In: 2nd Australian Road Research Board, pp. 388-393. (Year)
11. Zhao, J., Dessouky, M., Bukkapatnam, S.: Optimal Slack Time for Schedule-Based Transit Operations. Transportation Science 40, 529-539 (2006)
12. Strathman, J., Kimpel, T., Dueker, K.: Automated bus dispatching, operations control and service reliability. Transportation Resesearch Record 1666, 28–36 (1999)
13. Mishalani, R.: Passenger Wait Time Perceptions at Bus Stops: Empirical Results and Impact on Evaluating Real-Time Bus Arrival Information. Journal of Public Transportation 2, (2006)
14. Newell, G.: Control of pairing of vehicles on a public transportation route, two vehicles, one control point. Transportation Science 8, 248-264 (1974)
15. Powell, W., Sheffi, Y.:: A Probabilistic Model of Bus Route Performance. Transportation Science 17, 376-404 (1983)
16. Nicholson, A., Mei, K.: Assessing the effect of congestion on bus service reliability. 2nd International Symposium on Transport Network Reliability, Christchurch, NZ (2004)
17. Furth, P., Hemily, B., Muller, T., Strathman, J.: Uses of Archived AVL-APC Data to Improve Transit Performance and Management: Review and Potential. Transportation Research Board (2003)
18. Vuchic, V.: Transit Systems, Operations and Networks. Urban Transit. Wiley, New York (2005)
19. Agrawal, R., Srikant, R.: Mining Sequential Patterns. Eleventh International Conference on Data Engineering, pp. 3-14 Taipei, Taiwan (1995)
20. Srikant, R., Agrawal, R.: Mining Sequential Patterns: Generalizations and Performance Improvements. 5th International Conference on Extending Database Technology, pp. 3-17, Avignon, France (1996)
21. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. 20th International Conference on Very Large Data Bases, pp. 487-499, Santiago de Chile, Chile (1994)
22. Jian, P., Han, J., Mortazavi-asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. 17th International Conference on Data Engineering, pp. 215-224, Heidelberg, Germany (2001)
23. TRB: Transit Capacity and Quality of Service Manual. Transit Cooperative Research Program Web Document No. 6. Transportation Research Board - National Research Council, Washington, D.C. (1999)
24. Gama, J., Gaber, M.: Learning from Data Streams, New York (2007)