Event-Based Passive Tracking of Public Transportation Passengers

Yuval Hadas · Boaz Ben Moshe

Abstract The quality and availability of data collection systems are essential for a high-quality planning and operations of public transportation (PT) systems. Passengers' demand (both spatial and temporal) is a key component in those phases. Acquiring accurate and complete passengers' demand is a complex and time-consuming process. Various information technologies are used to assist in the planning and operations phases. Unfortunately, those systems are not capable of completely tracking passengers' movements throughout the system, i.e., the identification of the major events: arrival at a stop, boarding/alighting (unless swipe-in/out is performed on all trip legs), in-vehicle positioning, and transferring. Global Navigation Satellite System (GNSS) technology can overcome some of the disadvantages. The main drawback of such an approach is the high-energy consumption. Other wireless options exist, such as WLAN and Bluetooth, which are more energy efficient. In this work, a Bluetooth Low Energy based system for tracking PT passengers is introduced. The advantages are: low-energy consumption, acceptable positioning accuracy, passive and anonymous data collection. A framework is introduced, along with algorithms for the automatic identification of these major events. A case study provides insights on the capabilities of the model.

Keywords: Public transportation · Data collection · Tracking · Positioning · Bluetooth
1 Introduction

The quality and availability of data collection systems are essential for a high-quality planning and operations of public transportation (PT) systems. Passengers' demand (both spatial and temporal) is a key component in those phases. Unfortunately, acquiring accurate and complete passengers' demand is a complex and time-consuming process. Various information technologies are used to assist in the planning and operations phases. The main systems being used are: a) Automatic Vehicle Location (AVL) systems b) Automatic Passenger counting (APC) systems, and c) AFC (Automatic Fare Collection) systems (Acumen Bulding Enterprise Inc. et al. 2006; Furth et al. 2006; Strathman et al. 2008). Each of those Automatic Data Collection systems (ADC) has a specific contribution: AVL systems provide tracking of buses, APC systems provide passenger loads, and AFC systems assist with ticketing and monetary transactions. The data acquired can be used for analysis, as well as to enhance the performance of PT systems, and the introduction of advanced models. Unfortunately, those systems are not capable of completely tracking passengers' movements throughout the system, i.e., the identification of the major events: arrival at a stop, boarding/alighting (unless swipe-in/out is performed on all trip legs), in-vehicle positioning, and transferring. Global Navigation Satellite System (GNSS) technology can overcome some of the disadvantages of AVL, APC, and AFC technologies (Yilin 2000). The main drawback of such an approach is the high-energy consumption (user-side). Economical GNSS energy consumption management reduces tracking accuracy, hence prohibit precise positioning (Carrel et al. 2012). Other wireless options exist, such as WLAN and Bluetooth, which are more energy efficient. The main advantages of Bluetooth Low Energy (BLE) is low energy consumption (both user and transmitter-side), positioning accuracy in close range to the transmitters (indoor and outdoor), and low-cost (Gomez et al. 2012).

In our work, a BLE based system for tracking PT passengers is introduced. The advantages of such an approach are: 1) passive data collection which does not require any action from the passengers. 2) automatic events’ identification. 3) anonymous data collection.

Section 2 provides an overview of the technology. In section 3, the system’s framework is introduced, followed by the algorithms required for the identification of the trip’s events. A case study is introduced in section 4, which demonstrates the system’s capabilities. Conclusions are drawn in section 5.

2 BLE technology overview

It should first be emphasized that BLE technology is not claimed to be the optimal choice with respect to positioning accuracy, but rather constitutes a good
compromise of: (1) sufficient accuracy potential in diverse indoor and outdoor environments; (2) highly-efficient energy consumption characteristics; (3) cost-effective deployment model, due to BLE sensor availability in modern Smartphones, as well as off-the-shelf non-expensive (5-10 USD) BLE-tags that are very small in size. (4) available commercial products for Apple iOS and Android (Kashevnik and Shchekotov 2012). It should also be noted that BLE technology is somewhat less straightforward for positioning in comparison to Wi-Fi. This is mostly due to the ease to monitor signal strength of nearby Wi-Fi APs, without association and without authentication. In contrast, BLE technology requires a higher degree of authentication and does not allow unassociated monitoring of emitters' signal strength. Several academic works such as (Bandara 2004; Feldmann 2003; Hay 2009; Kotanen 2003; Pei 2012; Zhou 2006) discuss indoor positioning systems that are based on Bluetooth RSSI signal measurements. Notably, the elementary technological requirement of a PT passengers positioning system is to allow end-users' smartphones to determine their positions inside buses, near bus stations, and in complexes such as central bus stations. Principally, these tasks belong to the field of indoor positioning, and there are multiple prior works teaching how equivalent tasks can be implemented using Wi-Fi or BT technologies, by utilizing either RSSI-based weighted averages or Fingerprinting methodologies. Because RSSI based positioning is widely covered, we have chosen to focus on a system configuration, which can operate without access to RSSI values.

In many scenarios of positioning passengers in PT, RSSI values are not mandatory in order to locate passengers with sufficient accuracy. The basic enabler in these situations is the ability of the end-user's Smartphone to determine whether or not it succeeds to receive signals of BLE-tags (emitters) having known locations. Several use-case examples illustrating this concept are: (1) extremely weak BLE tags attached to the front and back doors of a bus, or to the sides of a bus station. (2) BLE tags that are spread in the target location (e.g., central bus station). Each tag is configured to transmit at different signal strength. (3) A BLE-tag configured to transmit consecutive signals at differentiated signal strengths. Fig. 1 illustrates this concept. The signal strength is dropping fast as the object moving away from the BLE tag. Hence it is sufficient to monitor whether a BLE-tag is present or not. Moreover, the tracking model presented in section 3, and demonstrated in section 4, does not use signal strength for trip construction. In order to validate that signal strength in not required, it was collected during the experiment, and found redundant.
BLE Energy Consumption

BLE’s superior energy efficiency is discussed in several prior works such as (Ben Moshe et al. 2014). We have also experienced redundant energy consumption of our BLE-enabled Smartphone app, in comparison to battery depletion while activating sensors such as the Smartphone’s GPS. We have tested several GPS/GNSS applications for PT such as: Moovit (Moovit 2013) and Offi (Schildbach 2013) and found that they usually drain the battery of the mobile device within ~2-3 hours, while our implementation allows a whole day of use. BLE has a similar Tx/Rx component as standard Bluetooth, but it has a very low duty cycle, allowing the BLE device to be in “deep-sleep” in 99.0-99.8% of the time. This leads to significant improvement in the power consumption of BLE, which may be up to 100 times more efficient than standard Bluetooth. Moreover, for positioning application BLE consumes significantly less energy than other sensors such as mems-gyro, e-compass, accelerometer, or barometric pressure sensor. Thus, a BLE-based positioning device may be able to run for months using a coin size battery, and a BLE tag may be able to use a small solar panel of 10cm x 10cm in order to continuously charge the battery with only 10% “sun-time”. Table 1 presents the power consumption of various positioning sensors. It clearly shows the superior energy efficiency of BLE in comparison to other positioning sensors. Furthermore, based on the case study (section 4), batteries were replaced after over a year from installation (2 standard AAA batteries for a BLE tag).
Table 1 power consumption of positioning sensors

<table>
<thead>
<tr>
<th></th>
<th>BLE</th>
<th>BT 2.1</th>
<th>WLAN</th>
<th>INS</th>
<th>GPS (GNSS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current (in data sheet)</td>
<td>0.01-0.1 (mA)</td>
<td>1-3 (mA)</td>
<td>10-20 (mA)</td>
<td>5-10 (mA)</td>
<td>100-250 (mA)</td>
</tr>
<tr>
<td>Expected battery life</td>
<td>100 days</td>
<td>3-10 days</td>
<td>6-12 hours</td>
<td>12-24 hours</td>
<td>1-2.5 hours</td>
</tr>
<tr>
<td>Tested consumption</td>
<td>0.5 (mA)</td>
<td>3 (mA)</td>
<td>20 (mA)</td>
<td>12 (mA)</td>
<td>300 (mA)</td>
</tr>
</tbody>
</table>

3 Event-based tracking system

3.1 The general framework of the system

The system is composed of the following components: a) low-cost, small size, and battery-operated BLE tags attached to PT stops and PT vehicles, b) a smartphone application, and c) data server. Based on the proximity to the tags, it is possible to identify automatically all the above-mentioned events (arrival, boarding, alighting, etc.), as illustrated in Fig. 2.

![Fig. 2 The tracking system’s components and tracked events](image-url)
For the identification of passengers’ trips events, a model was developed which is composed of three components. 1) data collection from the tags, 2) clustering the data into blocks, and 3) events’ identification and trip’s construction.

3.2 Data collection

The data collection model is based on a communication system composed of tags and smartphones. The tags are small-size and low-energy BLE transmitters, capable of transiting Bluetooth signals up to 50 meters. Each tag is associated with a PT stop or a PT vehicle, as illustrated in Fig. 2. The passengers’ smartphones receive signals from the tags and transmit them to the database server. Each transmission contains: 1) the time, 2) the tag ID, 3) the signal strength, and 4) the unique smartphone BT address (which cannot be associated with the phone number or the smartphone account).

The database server links the tag ID with its assigned object and object type (stop or vehicle). Furthermore, transmissions with unidentified tags are discarded. Table 2 presents a sample data from the raw data file: the time, passenger identification, tag ID, RSSI (signal strength in dB), the object code, and object type.

<table>
<thead>
<tr>
<th>Time</th>
<th>Passenger</th>
<th>Tag ID</th>
<th>RSSI</th>
<th>Object</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>10/08/2017 14:25:22</td>
<td>14247</td>
<td>7C:EC:79:FD:5E:12</td>
<td>-96</td>
<td>V1</td>
<td>Bus</td>
</tr>
<tr>
<td>11/08/2017 08:17:21</td>
<td>125625</td>
<td>98:7B:F3:60:83:3E</td>
<td>-88</td>
<td>BS02</td>
<td>Stop</td>
</tr>
<tr>
<td>10/08/2017 08:30:25</td>
<td>127297</td>
<td>98:7B:F3:60:83:3E</td>
<td>-90</td>
<td>BS02</td>
<td>Stop</td>
</tr>
<tr>
<td>10/08/2017 12:16:20</td>
<td>171339</td>
<td>98:7B:F3:60:83:3E</td>
<td>-87</td>
<td>BS02</td>
<td>Stop</td>
</tr>
</tbody>
</table>

As a result, the database accumulates information of the temporal proximity (realized by the signal strength) of passengers to PT stops and vehicles.

3.3 Clustering algorithm

For the data to be easily and efficiently analyzed, it is necessary to cluster the transmissions into data blocks, each associated with a passenger and the proximity duration to a tag. Each cluster contains the passenger identification, the tag ID, and the first and last times a signal was received.

For better understanding, Fig. 3 illustrates the clustering process. The top chart presents the non-clustered data, for a certain passenger, in which the X-axis represents the time, and the Y-axis the different tags. Each dot represents that a signal was detected, at 10-second intervals. The bottom chart illustrates the results
of the clustering process. Each box represents a different cluster, identified by its start and end times.

![Diagram](image)

**Fig. 3** The clustering process: non-clustered data (top) and clustered data (bottom)

### 3.4 Trip construction algorithm

The trip construction algorithm is required to identify each of the trip’s events (arrival, boarding, riding, alighting, and departure) based on the clusters, and to assign the time each event starts and ends. Based on the clusters' data alone it is possible to identify four states in which a passenger is at, according to the proximity to stop and vehicle tags: 1) near a stop, 2) near a vehicle, 3) near a stop and a vehicle, 4) not in the PT system. These states are summarized in Table 2.

Furthermore, any trip can be defined as a sequence of these states, as illustrated in Fig. 4. A passenger’s trip starts with the arrival at a stop (a), a set of at least one state where both vehicle and stop tags are present (b), a state in which the passenger
is on board a vehicle (c), states where both tags are present (a passthrough via a stop) (d), and finally, proximity to the departure stop (e). These ordered states (a-e) are also presented in Table 3.

Table 3 The relation between proximity to the tags and the state

<table>
<thead>
<tr>
<th>Proximity to a stop tag</th>
<th>Proximity to a vehicle tag</th>
<th>Yes</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Near a stop and a vehicle ([b, d])</td>
<td>Near a stop ([a, e])</td>
<td></td>
</tr>
<tr>
<td>No</td>
<td>Near a vehicle ([c])</td>
<td>Not in the PT system</td>
<td></td>
</tr>
</tbody>
</table>

![Diagram showing the sequence of states](image)

Fig. 4 The sequence of states related to a single trip

However, each state by itself cannot be used to identify the trip’s events. State “a” can occur when a passenger is near a stop tag, but has no intention of boarding a vehicle. Similarly, state “b” does not imply that a passenger is boarding.

On the other hand, the partial sequence “a-b\(_i\)-c” can confirm that a passenger arrived at a stop and boarded a vehicle. The same applies to the full sequence “a-b\(_i\)-c-d\(_j\)-e” which further confirms the ride, the alighting, and the departure.

Given that each cluster is associated with a stop or a vehicle, and with start and end times, it is possible to identify these states. Fig. 5 presents a sample sequence of clusters that corresponds to an actual trip taken by a passenger, with the states highlighted.
Fig. 6 presents the pseudo code of the events’ identification algorithm. The algorithm iterates through all clusters (c), ordered by Passenger (U) and start time. S and E represent the start and end times respectively, and T the cluster type (1-stop, 2-vehicle). Two gap thresholds, d₁ and d₂, are defined. The first defines the maximal gap between consecutive clusters to be considered part of the same trip. The second defines the minimal duration for a set of clusters to be considered as a trip. The algorithm assigns to each cluster three variables: 1) the event type (M), 2) the sequence within the trip (N), and 3) the trip termination time (R).

1. For each cluster c
2. \[ g_1 = S_{c+1} - E_c \]
3. \[ g_2 = E_{c+1} - E_c \]
4. If \( g_1 > d_1 \) And \( S_{c+1} > R \) Or \( U_c <> U_{c+1} \) Then
5. \[ R = 0 \]
6. End If
7. If \( M_c = 0 \) And \( T_c = 1 \) Then \( M_c = 1 \) 'walk by stop
8. If \( U_c = U_{c+1} \) Then
9. If \( T_c = 1 \) And \( M_c = 1 \) And \( T_{c+1} = 2 \) And \( E_c > E_{c+1} \) And \( g_2 > d_2 \) Then
10. \[ N = N + 1 \]
11. \( M_c = 2 \) 'boarding
12. \( M_{c+1} = 3 \) 'riding
13. \( R = E_{c+1} \)
14. Else If \( T_{c+1} = 1 \) And \( E_{c+1} < R \) Then
15. \( M_{c+1} = 4 \) 'pass-through
16. Else If \( T_{c+1} = 1 \) And \( S_{c+1} <= R \) And \( E_{c+1} >= R \) Then
17. \( M_{c+1} = 5 \) 'alighting
18. End If
19. End If
20. If \( M_c = 4 \) And \( R = 0 \) Then
21. \( M_c = 5 \)
22. End If
23. If \( M_c > 1 \) Then
24. \( N_c = N \)
25. Else
26. \( N_c = 0 \)
27. End If
28. If \( M_{c+1} > 1 \) Then
29. \( N_{c+1} = N \)
30. Else
31. \( N_{c+1} = 0 \)
32. End If
33. Loop

Fig. 6 Pseudo code for the trip construction algorithm
Based on the algorithm results, each event’s start time or duration can be easily calculated. Table 4 summarizes the exact calculations for each event.

<table>
<thead>
<tr>
<th>Event</th>
<th>Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival</td>
<td>$S(a)$</td>
</tr>
<tr>
<td>Waiting</td>
<td>$S(b_n) - S(a)$</td>
</tr>
<tr>
<td>Boarding</td>
<td>$S(c)$</td>
</tr>
<tr>
<td>Riding</td>
<td>$S(e) - S(b_n)$</td>
</tr>
<tr>
<td>Alighting</td>
<td>$S(e)$</td>
</tr>
<tr>
<td>Departure</td>
<td>$E(e) - S(e)$</td>
</tr>
</tbody>
</table>

### 4 Case study

A free shuttle service was used to demonstrate the model. The service is a loop route serving Bar-Ilan University’s students, faculty, and administrative staff. The route has 17 stops, is 4 km long, and is served by 8 electric-powered vehicles. The headway is 10 minutes during peak hours, and 20 minutes during off-peak. Tags were installed at the stops and vehicles (Fig. 7), and a popular smartphone app was used to collect the data. The app was selected as its being used by most of the students as a kiosk for academic information. A special software module was added to the app for communicating with the tags and the database server.

Unlike GPS based apps, such as Google maps, Moovit, etc., which significantly increase energy consumption while being used, the BLE based app did not increase energy consumption, even though the app was constantly active in the background, for data collection.

A time-space chart for 2:30 hours is presented in Fig. 8, which illustrates the full trips of several passengers. Each bar represents a cluster associated with a passenger and a tag. The stops are at the lower part of the Y-axis, and the buses are at the upper part. Fig. 9 presents statistics regarding the waiting, riding, and departure times. The average times are 3:29, 6:49, and 0:32 minutes respectively (marked with “x”). The average waiting time is in line with the headways, as well as the riding time, which corresponds to a trip half of the round-trip (which is ~15 minutes).

Fig. 7 Tags located at a stop (left), and inside a vehicle (right)
Fig. 8 A time-space chart of several trips

Fig. 9 Box and Whisker chart for the waiting, riding, and depart times (n=50)
The advantages of the model are clear, waiting and departure time can be easily calculated, which current technology cannot. Furthermore, riding times can be calculated without the passengers actively mark boarding and alighting.

A large-scale demonstration has recently started, with the participation of one of Israel’s PT operator. The demonstration is focused at two popular bus routes, served by 50 buses and with over 80 stops.

5 Conclusions

The suggested framework has the following advantages: 1) improving the quality of data collection, 2) enabling the implementation of advanced PT models, such as DRT, 3) the implementation of zone-based ticket validation, 4) alternative to VMS (“next-bus”), 5) multi-language interface for tourists, etc., 6) smart card/e-wallet integration, 7) better accessibility for disabled and mobility impaired passengers, such as the blind, 8) integration with other sensors (temperature, noise, humidity, pollution, radiation) as part of a smart city system.

Acknowledgements: This research was supported by the Ministry of Transport and Road Safety, Israel (Grant No. 204156).

References


