Available online at www.sciencedirect.com

ScienceDirect

Procedia Technology 18 (2014) 120 – 125

Procedia Technology

International Workshop on Innovations in Information and Communication Science and Technology, IICST 2014, 3-5 September 2014, Warsaw, Poland

Measuring bus passenger load by monitoring Wi-Fi transmissions from mobile devices

Thongtat Oransirikul, Ryo Nishide, Ian Piumarta*, and Hideyuki Takada

College of Information Science and Engineering, Ritsumeikan University, Kusatsu, Shiga, Japan

Abstract

Uneven loading of busses degrades passengers' travel experience for a variety of reasons. Load balancing for busses requires information about the load, both the potential passenger load at the bus stop and the actual load currently aboard travelling busses. This paper describes a feasibility study for measuring bus passenger loads by detecting the periodic network probing activity from WiFi devices built into 'smart phones'. Our experimental results show that WiFi activity does correlate with observed passenger flow at bus stops and the load aboard a bus while *en route*.

Peer-review under responsibility of the Scientific Committee of IICST 2014

Keywords: Bus loading; Passenger flow; Public transport congestion; Wireless network monitoring

1. Introduction

800,000 people in Kyoto, more than half of the population, use public transportation every day. 320,000 of them travel by bus [6]. Time spent taking a bus is divided between waiting and traveling. Of the two, studies show that passengers consider travel time to be of far higher value since it is used productively (whereas wait time is generally wasted) [7]. One cause of increased wait time is bunching (or clumping), when an overloaded bus falls behind its schedule. Falling behind schedule also causes unbalanced congestion on the first bus, leading to lower-quality travel time for its passengers. When bunching occurs it tends to be self-perpetuating because buses running to schedules do not overtake each other—and the situation would not improve even if they did [10].

Load balancing for busses is therefore our long-term goal. Passengers are willing to wait a short time to obtain a higher-quality travel experience aboard a lightly-loaded vehicle. Making passengers aware of a lightly-loaded bus following closely behind a heavily-loaded one gives them an opportunity to maximise the quality of their travel time. At the same time, while this will not eliminate bunching, it should help to lessen its severity.

* Corresponding author. Tel: (+81) 077–561–5202 FAX: (+81) 077–561–5203
E-mail address: piumarta@cs.ritsumei.ac.jp

Peer-review under responsibility of the Scientific Committee of IICST 2014
doi:10.1016/j.protcy.2014.11.023

In extreme cases, a driver of a loaded bus who is made aware of a closely-following lightly-loaded bus could choose not to stop to let more passengers board. A bus company, made aware of abnormally high demand at bus stops, could choose to schedule additional busses.

Our immediate short-term goal, and the subject of this paper, is therefore to determine the feasibility of estimating passenger flow at bus stops and passenger congestion aboard busses using a reliable, non-invasive, and non-participatory mechanism.We experimented with two mechanisms for data collection, finally settling on wireless network (WiFi) activity from smart phones probing for access points. Analysis of data collected at the University's bus terminal, and on board a shuttle bus that runs to a local train station, suggests that this is a feasible mechanism for collecting the data that we need to provide useful load-balancing information and choices to both users and operators of the bus service.

In the rest of this paper we present related work in Section 2. Section 3 explains our experiment method. The results are presented in Section 4 and discussed in Section 5. Finally, Section 6 presents our conclusions and ideas for future work.

2. Related Work

Several groups have reported work on bus scheduling systems that take into account road traffic conditions and passenger's waiting time at the bus stop.

Chen [1] simulates the bus services running on a fixed route, analysing the impact on overtaking the previous bus at the bus stop, while considering the bus capacity and passenger congestion on the bus. The goal is to provide efficient bus routes considering road traffic conditions or load balancing for passengers on multiple busses, but little is presented about the detection of passengers. Zhou [22] estimates bus arrival time at the stop by detecting mobile phones carried by passengers waiting for and boarding the bus. Cell tower and audio signal detection are used to detect and track passengers. Cai [16] describes an intelligent transportation system (based on Zigbee wireless communication devices) that exchanges information between busses, bus stops, and the bus company in order to reduce traffic congestion and improve public transport efficiency. The system requires bulky devices to be installed on the bus and at bus stops [16], as well as cooperation from cell phone companies (for cell tower information) and the bus company [16,22].

GSM signal strength recorded at cell towers has used to discover users' locations [17]. This method may be effective to analyse user's activities, but does not directly address the analysis of their surrounding social contexts.

Methods for extracting the flow and social context of pedestrian activity has been discussed in relation to various goals. Some work focuses on scanning, logging and analysing the Bluetooth MAC addresses transmitted from mobile devices [3,9,13,14]. In recent years, however, Bluetooth signals have become less useful because an increasing number of mobile devices automatically disable Bluetooth functionality if it is unused for a certain time [12]. Other methods focus on detecting WiFi MAC addresses but, to our knowledge, they gather information from WiFi hot-spots rather than directly from the mobile stations. Most of these methods can measure the signal strength from several WiFi hot-spots to improve accuracy of user location [8,15]. Nakano [11] has similar goals to ours and installed a WiFi router to communicate with surrounding devices on a train, estimating the passenger congestion by receiving requests to join the network. The major different to our method is that we have conducted our experiment without using a WiFi router and in a manner that is entirely non-disruptive to any genuine WiFi service that is provided aboard the vehicle.

Ubiquitous computing groups have used public transportation as a target for their technology, including Ubibus [19] which is an application designed to provide real-time context information about vehicles, passengers, and any other dynamic factors that affect the transportation. Onebusaway [4] is an application which displays all bus stops closer to user's location on a map, in reference to user's current schedule, bus timetable, and wait time at the bus stop. Other applications analyse pedestrian flow at music events [18], festivals [21], University campuses [20], and so on.

Our goal is to infer passenger flow and congestion information at bus stops and on board busses by monitoring the WiFi packets sent from mobile phones. A log file with collected WiFi MAC addresses and time stamps, is generated, which is then used to analyse the duration, quantity, and flow of the devices and therefore of the passengers carrying them.

3. Experimental Method

Experiments were conducted at the Ritsumeikan University bus terminal. The location has three closely-spaced bus stops with overlapping arrivals and departures, a published bus schedule, large contrasts between quiet and peak times, and busses that leave almost-full every five to ten minutes during peak hours.

Our first attempt to count passengers relied on Bluetooth device detection. An initial experiment detected only eight Bluetooth devices during a one-hour period, demonstrating that this approach is not feasible.

Given the popularity of WiFi-enabled smart phones, each transmitting packets (probing for known networks) even when otherwise idle, we decided to try detecting wireless network activity instead. To identify unique passengers we needed to record the link-layer *medium access controller* (MAC) hardware address of devices transmitting in the area. The wireless network interface controller of a Linux laptop was placed in *monitor mode*[1] and `tcpdump` [5] used to capture frames and record their link-layer headers. The link-layer header includes the MAC addresses of the transmitter and receiver. In combination with the time stamp (added by `tcpdump`) we were able to recover the first and last time that each unique device was seen. This approach looked promising except for the large numbers of IPv6 multicast packets that swamped the data, and the inability of `tcpdump` to capture packets from all available radio channels.

We eventually chose `airodump-ng` [2] for packet capture. It works in a similar manner to `tcpdump` except that it ignores multicast traffic, captures packets from all available channels by 'channel-hopping', and usefully segregates the collected data into that associated with access points and that associated with mobile stations.

For the final experiment, the Atheros AR9285 wireless network interface controller of a Hewlett-Packard ProBook 4320s laptop running Ubuntu Linux was placed in monitor mode and `airodump-ng` used to capture data for a 60-minute period at a bus stop during a busy part of the afternoon, and then during a ten-minute ride aboard a shuttle bus to a nearby main-line train station. During data capture the arrival and departure times of the busses were recorded, as well as the approximate number of passengers travelling on each one. To estimate the number of passengers we observed that each bus has a seating capacity of approximately 45 people and room for approximately 15 more standing. A bus that was almost full of seated passengers was therefore carrying approximately 40 people, and a bus with the standing space fully-occupied had approximately 60 people aboard, etc.

Data collected by `airodump-ng` already includes aggregated information for each mobile station including the first and last times the station was seen, the total number of packets it transmitted, and the signal strength of its radio transmitter. The data file is formatted as comma-separated values and is easily manipulated by shell scripts using the standard Unix tools such as `grep`, `awk` and `sed`. Raw data was visualised with GhostScript and processed data visualised using `gnuplot`.

4. Results

The experiment ran for approximately 70 minutes. The first 60 minutes were spent at the Ritsumeikan bus terminal, and the final ten minutes on board a shuttle bus (which makes no stops) during its journey from the terminal to a nearby train station. The collected data is summarised in Figure 1.

1,390 unique MAC addresses were recorded during the experiment. Many of these were seen exactly once, and several were active for the entire 60 minute period at the terminal. Neither these, nor devices that were far from the receiver, represent useful indicators of passenger populations. The raw data was therefore filtered to remove stations that transmitted only one packet, those that were active for a period exceeding 30 minutes, or those whose maximum signal strength never rose above -74 dB. After filtering, 333 stations remained in the sample.

A total of 12,639 packets were monitored during the experiment, with a median transmission rate of 0.117 packets per second (one packet every 8.5 seconds) per device. The 32 available channels were sampled round-robin at a rate of 3.3. channels per second.

[1] Monitor mode is similar to the *promiscuous mode* of wired Ethernet interfaces. It allows a wireless network interface to capture all received packets without having to be associated with any access point or *ad-hoc* network.

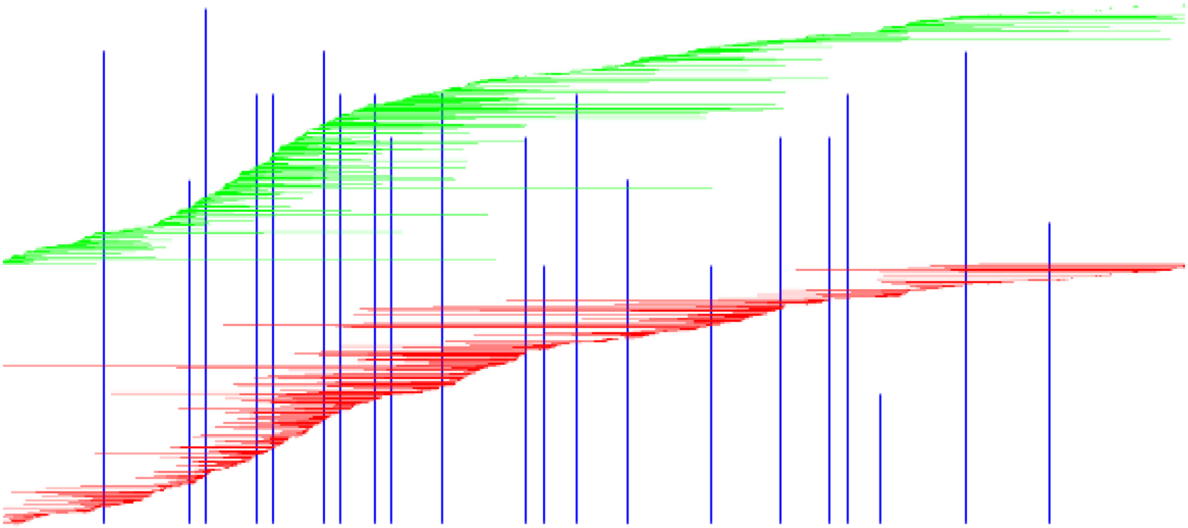

Fig. 1. Summary of the filtered raw data. Each horizontal line represents a recorded mobile station, extending from the first time it was seen until the last time. The same set of 333 stations are plotted twice. In the upper half of the figure the stations are sorted by their arrival time, and in the lower half by their departure time. The left edge of the figure corresponds to the beginning of the experiment, and the right edge to 4200 seconds (70 minutes) of elapsed time. The vertical lines represent the observed bus departure times. The height of each line is proportional to the number of passengers aboard. The shortest line (third from right) corresponds to 15 passengers, and the tallest line (third from left) to 60. The rightmost vertical line is the shuttle bus leaving for the train station, with the experimental apparatus aboard.

During the initial 60-minute period at the bus terminal, 21 busses departed carrying a minimum of 15 passengers and a maximum of 60. A total of 945 passengers departed, for an average of 45 passengers per bus.

5. Discussion

In Figure 1 the long green/red lines show that passengers are waiting for a bus for a relatively long time. This might have occurred from passengers that are waiting for a less frequently-running bus or waiting for another bus to arrive with available seats. There is also a noticeable increase in the number of passengers waiting for a short time to board a bus beginning about ten minutes into the experiment, which coincided with the end of a lecture period.

The final ten-minute ride to the train station can be seen clearly as a thick cluster of lines extending to the end of the experiment, representing many passengers boarding a bus and remaining aboard until the terminus. This shows that the appearance and disappearance of devices resembles the dynamic behaviour of passengers and their social context.

The filtered raw data was analysed to provide approximations for the total population (number of devices that have already arrived but not yet left) and the rate of extinction (number of devices that were last seen) for each one-minute interval of the experiment. The left half of Figure 2 shows the results.

Correlating the total population to bus departures proved difficult, possibly due to complexity in the data caused by the presence of several bus stops servicing both arrivals and departures, and a surge in detected devices after the second teaching period of the afternoon (which ends at 16:10, and generates a clearly-visible increase in population beginning ten minutes into the experiment and lasting approximately 20 minutes).

To better approximate the population information that could be recovered at an isolated bus stop located *en route*, we tried correlating the extinction rate with the number of passengers departing. A visual inspection of the data suggested a correlation containing a slight lag, and so a Pearson correlation was performed after shifting the extinction data two minutes later which gave a value of $r = 0.6215$.

The journey to the train station is illustrated in the right half of Figure 2. The mobile stations that survived the first minute of the trip (and which were therefore aboard the bus) mostly survived to the end of the ten-minute trip. *En route* the number of new appearances approximately equalled the number of extinctions, each having a lifespan of

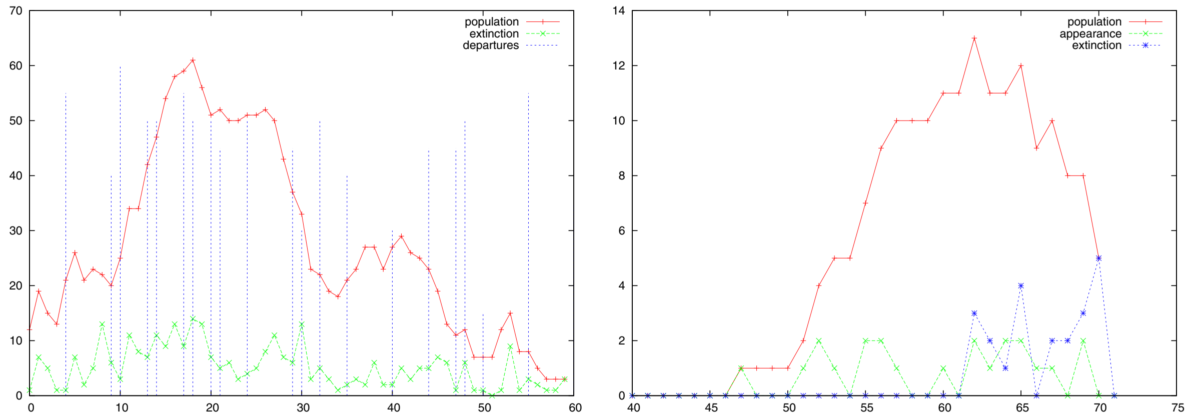

Fig. 2. Total passenger population and rate of extinction plotted against time. The left hand graph covers the time spent at the bus stop, and shows bus departure times and loads as vertical blue lines. The right hand graph covers the time spent aboard the shuttle bus between the University and the train station.

just a few seconds. These were likely stations located outside the bus detected once or during a short period spent at traffic lights. A few of the stations aboard the bus died during the trip, for reasons that are not known. Considering the number of people observed to be aboard the bus and the number of devices whose signals were observed during the entire trip, we can estimate a constant of proportionality between devices and passengers of approximately 5.5.

6. Conclusions

We have examined the detection of passengers' mobile devices by monitoring their WiFi activity and the possibility of extracting the passenger flow information at a bus stop. We collected actual data at the bus stop and aboard a bus during a rush hour when many students leave the campus. The method we employed is simple and convenient, and does not require additional bulky devices, applications, or disruption to passenger WiFi services.

Our experimental results show that there are correlations between the collected data, passenger behaviour, and the surrounding context (e.g., when the experimenter boarded the shuttle bus).

Congestion might also be estimated with our method, but we have insufficient data to state the precise nature of the correlation between the behaviour of detected devices and that of the passengers at a bus stop. The complexity of the data may have been caused by: (i) multiple bus stops for different destinations being too close together, (ii) students gathering with friends around the bus stop and leaving randomly, (iii) students boarding a bus that then waits at the bus stop while blocking their WiFi signal, and so on. Capturing and analysing the WiFi packets might not be enough to extract the number of passengers, and may require enhancement of our method and analysis technique. Further experiments, especially in a less complex environment, will be necessary to answer these questions.

We believe that the information we are trying to collect useful for bus companies (to update and modify the timetable according to the congestion), for bus drivers (to collaborate between other busses to balance the number of passengers aboard), and for passengers (to inform them about the congestion aboard busses about to arrive and help them decide between getting on a heavily-loaded bus or waiting a few minutes for a lightly-unloaded bus). Our methods and results may also be useful for analysing the daily activity or behaviour of passengers, and so be applied to the field of psychological research.

We plan further exploration of efficient methods of detecting passengers and analysing the collected data to better determine the number and behaviour of passengers. Combination with other wireless technologies, including GSM, Bluetooth, etc., may be required to augment our method. Practical or commercial application of our methods to commuter information systems for public transportation, or its applicability to other special events such as festivals or event sites, amusement parks, shopping malls, etc., are also possibly themes for future work.

References

[1] Wei-Ya Chen. Fixed-route high-frequency bus service reliability considering bus overtaking and no-overtaking. *Int'l Conf on Transportation, Mechanical, and Electrical Engineering (TMEE)*, pages 1197–1200, 2011.

[2] Christophe Devine and Thomas d'Otreppe. `aircrack-ng` network software suite. {`http://www.aircrack-ng.org`}, 2004–.

[3] Nathan Eagle and Alex Pentland. Reality mining: sensing complex social systems. *Journal Personal and Ubiquitous Computing Volume 10 Issue 4*, pages 255–268, 2006.

[4] Brian Ferris, Kari Watkins, and Alan Borning. Location-aware tools for improving public transit usability. *IEEE Pervasive Computing*, pages 2–9, 2010.

[5] Van Jacobson, Craig Leres, Steven McCanne, et al. TCPDUMP packet analyser program. {`http://www.tcpdump.org`}, 1987–.

[6] Kyoto City Hall. Topics of City Government, May 2014. `http://www.city.kyoto.jp/koho/eng/topics/2014_5/index.html`.

[7] William Lam and John Morrall. Bus passenger walking distances and waiting times: a summer–winter comparison. *Transportation Quarterly*, 36(3):407–421, 1982.

[8] Locky. {`http://locky.jp/en`}, 2005–.

[9] Bram Van Londersele, Matthias Delafontaine, and Nico Van de Weghe. Bluetooth tracking. *GIM International*, pages 23–25, 2009.

[10] Debora MacKenzie. Why three buses come at once, and how to avoid it. *New Scientist*, October 2009.

[11] Ryusuke Nakano and Masayuki Numao. Congestion level estimation in a train using scan request to wireless LAN access point (in japanese). *DEIM Forum A10-1*, 2012.

[12] Ryo Nishide and Hideyuki Takada. Detecting pedestrian flows on a mobile ad hoc network and issues with trends and feasible applications. *International Journal on Advances in Networks and Services (IARIA Journal)*, 6(1,2):108–117, 2013.

[13] Ryo Nishide, Tatsuya Ushikoshi, Satoshi Nakamura, and Yasuyuki Kono. Detecting social contexts from bluetooth device logs. *Supplemental Proc. Ubicomp 2009 (ACM 11th International Conference on Ubiquitous Computing)*, pages 228–230, 2009.

[14] Eamonn O'Neill, Vassilis Kostakos, Tim Kindberg, Ava Fatah gen. Schiek, Alan Penn, Danaë Stanton Fraser, and Tim Jones. Instrumenting the city: Developing methods for observing and understanding the digital cityscape. *Ubiquitous Computing (UbiComp)*, 2006.

[15] PlaceEngine. {`http://www.placeengine.com/en`}, 2007–.

[16] Chang quig Cai, Zhuo Zhang, and Shi-Dao Ji. The intelligent bus scheduling based on ZIGBEE. *The 7th IEEE International Conference on Computer Science and Education (ICCSE)*, pages 1002–1005, 2012.

[17] Timothy Sohn, Alex Varshavsky, Anthony LaMarca, Mike Y. Chen, Tanzeem Choudhury, Ian Smith, Sunny Consolvo, Jeffrey Hightower, William G. Griswold, and Eyal de Lara. Mobility detection using everyday gsm traces. *Ubiquitous Computing (UbiComp)*, pages 212–224, 2006.

[18] Arkadiusz Stopczynski, Jakob Eg Larsen, Sune Lehmann, Lukasz Dynowski, and Marcos Fuentes. Participatory bluetooth sensing: A method for acquiring spatio-temporal data about participant mobility and interactions at large scale events. *International Workshop on the Impact of Human Mobility in Pervasive Systems and Applications*, pages 242–247, 2013.

[19] Vaninha Vieira, Ana Carolina Salgado, Patricia Tedesco, Valeria Times, Carlos Ferraz, Elisa Huzita, Ana Paula Chaves, and Igor Steinmacher. The UbiBus project: Using context and ubiquitous computing to build advanced public transportation systems to support bus passengers. *Simposio Brasileiro de Sistemas de Informacao (SBSI 2012) Trilhas Tecnicas*, pages 56–60, 2012.

[20] Long Vu, Klara Nahrstedt, Samuel Retika, and Indranil Gupta. Joint bluetooth/wifi scanning framework for characterizing and leveraging people movement in university campus. *Proceedings of the 13th ACM international conference on Modeling, analysis, and simulation of wireless and mobile systems*, pages 257–265, 2010.

[21] Jens Weppner and Paul Lukowicz. Bluetooth based collaborative crowd density estimation with mobile phones. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 193–200, 2013.

[22] Pengfei Zhou, Yuanqing Zheng, and Mo Li. How long to wait?: predicting bus arrival time with mobile phone based participatory sensing. *Proceedings of the 10th international conference on Mobile systems, applications, and services (ACM)*, pages 379–392, 2012.