WiBus: A Wi-Fi based Monitoring System for Public Transportation with Dynamic Route Tracking

Vitor Borges C. da Silva, Tatiana Sciammarella, Miguel Elias M. Campista, and Luís Henrique M. K. Costa
Universidade Federal do Rio de Janeiro - PEE/COPPE/GTA - DEL/POLI
E-mail:{borges,tatiana,miguel,luish}@gta.ufrj.br

Abstract—**A great challenge of today's metropolises is the constant traffic jams, which are a consequence of excessive use of private vehicles. More people would adopt public transportation, e.g. buses, if the time of arrival (ToA) for the next vehicle to each bus stop could be predicted. In this direction, we propose WiBus, which is a system to estimate the buses ToA, based on information from opportunistic IEEE 802.11 contacts. Unlike previous systems, WiBus explicitly takes into account bus route changes, adjusting the routes with an algorithm for dynamic route creation and maintenance. The system is implemented and analyzed via emulation of a real scenario. Experimental results show that WiBus has accumulated error of at most a few minutes.**

I. INTRODUCTION

Today, big cities suffer with constant traffic jams generated by an increasing number of vehicles. In Brazil, in just one year, from September 2012 to 2013, almost 3 million new vehicles started circulating on the streets, causing a significant deterioration of the traffic system [1]. The preference for private transportation is a consequence of the lack of confidence on public transportation commitment with timetables. To change this trend, the idea of the Advanced Public Transportation Systems (APTS) has emerged using, among other technologies, communication networks to provide more information to users [2]. Communication networks in vehicular scenarios predominantly use cellular (3G/4G) or IEEE 802.11 technologies. This work aims at improving the quality of service provided by public transportation systems. The idea is to offer fresh information regarding the ToA of the next bus of every bus line to each possible bus stop. Nevertheless, estimating the ToA of a bus at a given bus stop is a challenge, because traffic and passengers can influence the time a bus takes in its route. Moreover, situations such as an accident or a roadwork may result in changes in a bus route. If these changes were ignored, the estimates would become useless. Therefore, two requirements to calculate the ToA without significant errors are to track buses position; and to be aware of buses route changes in a dynamic fashion, which is a contribution of our work.

To deal with the first requirement, many solutions use GPS, due to its greater accuracy. However, the GPS use requires an additional communication link to inform vehicle position. As an alternative, there are localization techniques that use network technologies [3], which can provide both services of locating a bus and sending the corresponding information to an access point. This potential has been explored even in simple proximity-based localization systems. In such systems, the position of a vehicle is given by the position of the node the vehicle is connected to. Regarding estimates of buses ToA, some works use historical series collected even for months [4]. Then, the estimate is calculated based on averages from the same period of time in the past. Hence, this method does not behave well in case of atypical situations, such as accidents. In opposition to historical-based approaches, in real-time solutions is assumed that the time taken by previous vehicles in a given route is also the time next vehicles will spend [5]. Thus, only information from the same day is used, allowing this method to model unforeseen situations more efficiently. Finally, there are methods that use Kalman filters, which do not perform well when location data are temporally sparse [6].

This paper proposes WiBus, a system to estimate the buses ToA using IEEE 802.11 networks. WiBus tracks the position of buses based on their proximity to IEEE 802.11 access points installed alongside streets. This allows the use of only one type of device to track and communicate, reducing the system complexity. In this work, as localization information is not obtained at the same rate as with a GPS, and because we do not assume the existence of a database with past information, we use a real-time method to estimate ToAs. Based on information from previous buses, WiBus computes the ToA of a bus and dynamically adapts these times to route changes. Therefore, even if a bus deviates from its route, the system can adapt and keep users informed. WiBus estimates incur in errors on the order of a few minutes, as obtained from experiments with real datasets. Our experimental results show the benefits of using WiBus in real environments.

This paper is organized as follows. Section II and III present, respectively, the architecture and the implementation of WiBus. In Section IV, the evaluation setup is described and the corresponding results are shown. Finally, Section V concludes this work and presents future directions.

II. WIBUS ARCHITECTURE

The WiBus system has four types of entities: Central, Roadside Unit (RSU), Bus, and Client. The Central entity plays the most important role in the system. It has all the information needed to compute ToA estimates of buses, their route, location, and the time spent between consecutive stops. The Central also answers customers' ToA requests. Roadside Units (RSUs) are access points installed at bus stops. They provide wireless access to each router within each bus. These

IEEE 802.11 routers run the developed programs and are needed for bus tracking. The developed programs locate the bus in relation to the RSUs and send localization information to the Central through RSUs. A Client is any person that would request estimates to the system.

III. WIBUS IMPLEMENTATION

The WiBus system offers complementary services of localization and arrival time estimates of buses at all stops along their routes. Both services are based on proximity to RSUs. Therefore, upon arriving at a bus stop, the bus connects to the network provided by the RSU. Messages are then exchanged between the bus and the RSU for mutual identification. Then, the bus sends a message informing the Central to which RSU it is connected, the previous RSU it has been connected, the bus, and the bus line identifiers. Although the trivial approach to have the bus route is to assume that they do not change, we consider the possibility of changes. Therefore, from the messages received, the Central can dynamically update the bus line route, be aware of buses location, and estimate buses ToA at any bus stop ahead.

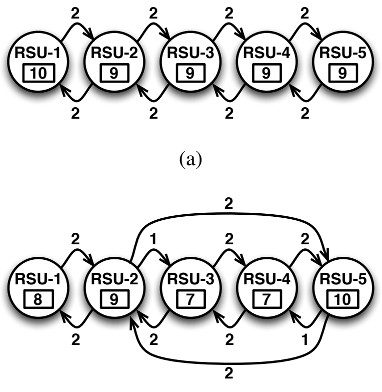
A. Operation of the Central

In WiBus, a bus line is defined by its number and direction (e.g., 913 South). As a consequence, bus routes are not circular for the system. Additionally, a given bus route consists of several segments, where each segment starts in a RSU and terminates in the consecutive one. Note that consecutive RSUs can be known according to the buses movement. Segments are important since at the end of each one, the bus must inform the Central its current position. This information is used to compute the average time between consecutive RSUs. When the Central receives a message from a bus, it updates the bus position using the RSU in the message. Also, it updates the average time needed to traverse that segment. With the average value of the next segments, the system can estimate the time required by a bus to reach any following RSU on its line. This is calculated by adding up the estimated time needed for the bus to traverse each segment, starting from the current and finishing at a target RSU. The problem, therefore, is reduced to estimate the time required for a bus to traverse a given segment. In WiBus, the estimate for a segment is calculated as the average of the time spent by K previous buses to travel the same segment. This approach reduces the complexity in ToA estimation. Upon receiving a message from a bus of a given bus line, the Central entity recalculates for each bus of the same bus line, the average time spent to reach each RSU along that bus line route. To store only the estimate to the next bus ToA at each RSU, only the lowest estimate time is stored.

B. Dynamic Algorithm for Route Creation and Update

The need for dynamic representation of bus routes comes from situations where unforeseen events can change the sequence of segments of a bus line, consequently affecting ToAs. One way of checking if a bus route changed is to observe buses from the same line. The change can be confirmed if they keep repeating the same new route. The algorithm aims at avoiding errors on ToA estimates either because of premature conclusions or excessive delay for bus route adaptation. Hence, the number of times a change must be repeated until it is considered valid is captured by the route computation algorithm proposed for WiBus.

In WiBus, a bus line route is modeled as a sequence of RSUs, where each RSU has directed edges to the next and previous RSUs of the route. To prevent premature changes to bus line routes, a weighted strategy is employed; weights are assigned to edges and to RSUs. Weights assigned to edges govern the dynamics of the bus line route, capturing how fast a change will be considered permanent. On the other hand, weights assigned to RSUs indicate if they are still used in the bus line route. We illustrate the introduced concepts in Figure 1(a), where a fictitious bus line route from RSU-1 to RSU-5 is modeled. Circles represent RSUs and rectangles within circles represent RSU weights. Arrows represent the directed edges linking one RSU to its successor or predecessor in the bus line route, where the number near the edge is its corresponding weight.



(a)

(b)

Figure 1. (a) Original bus line route. (b) Bus line route change.

The weighted strategy works by increments or decrements. The idea is to increment the weight of RSUs and edges currently used by buses along a bus line. In opposition, RSUs and edges currently not used have their weight decremented. The update process of RSU weights consists of decrementing by one the weight of all RSUs in the bus line. Next, the weight of the current RSU, the one informed in the message received from the bus, is incremented by an integer (specified later). Whereas the weights of the RSUs are updated every time a new message arrives to the Central, the weights of the edges are modified only when a bus line route changes. During a possible route change, a RSU may have more than one next or previous RSU. In such cases, the RSU has a list of edges indicating the next or previous RSUs, rather than just a single edge. Thus, the update of the edges to the previous or next RSUs is performed as follows: decrement by one the weight of all edges from the list of previous and next RSUs, and add two units to the weight of the edges used, i.e., previous and next. An example of a bus line route change in progress is shown in Figure 1(b), where RSU-2 has two possible next RSUs as well as RSU-5 has two possible previous RSUs.

The update process of the weighted strategy can culminate on RSU and edge creation or deletion. RSUs and edges are deleted if their weights reach zero, meaning the RSU (or edge) is no longer part of the bus line. In the case of RSU creation, the attributed weight must be large enough so that the RSU will not be mistakenly removed. So, the initial value assigned to a RSU is considered equal to twice the product of the number of RSUs in the bus line by the amount of bus circulating on that bus line, this value is also the maximum value allowed to the RSU weight. As the initial value of the RSU weight depends on the number of RSUs in the bus line, every time a RSU is added, to be fair, the weights of all RSUs in the line are set to the new maximum value. Similarly, the increment given when a RSU is used is equal to twice the number of RSUs in the bus line. Hence, the initial weight assigned to a RSU guarantees that all buses of a given line must travel over a route without a given RSU at least twice, in order to delete this RSU. In the case of the creation of an edge, the initial value of the edge weights can be adjusted, taking into account that its value must be at most two times the number of buses in the same bus line, thus it is possible to define how fast a change in the route will be considered permanent. The initial value chosen for the edges weight is also the maximum value allowed.

IV. EXPERIMENTAL SETUP AND EVALUATION

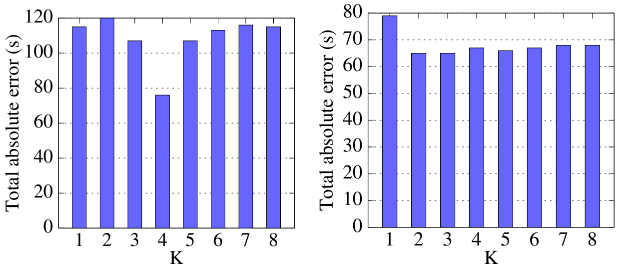
WiBus is evaluated using a prototype assembled in our lab. We have emulated the elements of the architecture using PCs and Wi-Fi routers. The Central entity is a PC configured with Debian, 8 GB of RAM, and Intel Core i7 860 processor. The RSUs and the buses are represented by D-Link DIR-320 Wi-Fi routers. We connect the Central to the RSUs via an Ethernet switch. The D-Link DIR-320 is equipped with a 240 MHz ARM processor and 32 MB of RAM. The operating system used in routers is OpenWRT Backfire 10.03.1.

The experiment, performed with real data of a university scenario, aims at evaluating WiBus estimation errors.

A. Experiments in a University Scenario

For this test, we measured nine times the time spent on all segments of the UFRJ's internal bus lines "COPPEAD" and "Estação UFRJ". These data are used to evaluate the quality of WiBus estimates, which calculate ToAs using simple moving averages of size K. This definition introduces a parameter, K, which is varied from 1 to 8. The value of K is chosen to minimize the average error of the obtained estimates from the first until the last bus stop of the evaluated lines. The values of K are those that produce the lowest absolute error according to Figure 2, which are $K = 4$ for "COPPEAD" and $K = 3$ for "Estação UFRJ".

Finally, from Figure 2, we verify that even for the entire bus route, the average absolute error of the estimates remains below 80s. We conclude that the estimates provided by WiBus have great potential for public transportation users, since the error is on average the time the user must wait at the bus stop until her/his bus arrives.



(a) "COPPEAD" Bus Line.

(b) "Estação UFRJ" Bus Line.

Figure 2. Average estimate error from the first bus stop to the last one for different values of K.

V. CONCLUSIONS AND FUTURE WORK

This paper proposed a system to estimate the ToA of buses at bus stops. WiBus uses information obtained from an IEEE 802.11 network, whose devices are installed inside buses and on the infrastructure along their routes. The idea is to track buses positions in order to estimate their ToAs. To avoid system failures in case of unforeseen route changes, an algorithm for dynamic creation and maintenance of routes was proposed. The ToA estimate is based on a moving average of size K, where examples of K were calculated with real data for two university bus lines. Results have shown that the estimates made by WiBus present errors on the order of minutes, for the entire route followed by the bus lines. As future work, we plan to create user interfaces to facilitate public access to the estimates and to measure the maximum number of buses under which the system is still able to track them and calculate the estimates.

ACKNOWLEDGEMENT

This work was funded by CNPq, CAPES, and FAPERJ.

REFERENCES

[1] DENATRAN, "Frota - DENATRAN - Departamento Nacional de Trânsito," www.denatran.gov.br/frota.htm (in Portuguese), 2013.

[2] R. F. Casey, L. N. Labell, S. P. Prensky, and C. L. Schweiger, "Advanced public transportation systems: the state of the art," U.S. Department of Transportation, Tech. Rep., 1991.

[3] J. Ribeiro Junior, M. Mitre Campista, and L. Costa, "COTraMS: A Collaborative and Opportunistic Traffic Monitoring System," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 949–958, June 2014.

[4] K. Manolis and D. Kwstis, "Intelligent transportation systems - travelers' information systems the case of a medium size city," in *Proceedings of the IEEE International Conference on Mechatronics*, 2004, pp. 200–204.

[5] W.-H. Lin and J. Zeng, "Experimental study of real-time bus arrival time prediction with GPS data," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1666, no. 1, pp. 101–109, 1999.

[6] A. Karbassi and M. Barth, "Vehicle route prediction and time of arrival estimation techniques for improved transportation system management," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2003, pp. 511–516.