The Airplace Indoor Positioning Platform for Android Smartphones

C. Laoudias†, G. Constantinou‡, M. Constantinides‡, S. Nicolaou‡, D. Zeinalipour-Yazti‡ and C. G. Panayiotou†
†KIOS Research Center for Intelligent Systems and Networks, University of Cyprus
‡Department of Computer Science, University of Cyprus
Email: laoudias@ucy.ac.cy, {gconst02, mconst02, snicol02, dzeina}@cs.ucy.ac.cy, christosp@ucy.ac.cy

***Abstract*—In this demonstration paper, we present an indoor positioning system developed for Android smartphones, coined Airplace. To infer the unknown user location we rely on ubiquitous WLANs and exploit Received Signal Strength (RSS) values from neighboring Access Points (AP) that are constantly monitored by the mobile devices under normal operation. Our system follows a mobile-based network-assisted architecture to eliminate the communication overhead and respect user privacy. In a typical scenario, when a user walks inside a building a smartphone client conducts a single communication with our Distribution Server to receive the RSS radiomap and is then able to position itself independently using the observed RSS values. Moreover, we have implemented an Android application to facilitate the collection of RSS values by users that may contribute their data to our system for constructing and updating the radiomap through crowdsourcing[1].**

We will demonstrate the real-time positioning capabilities of the system during the conference by allowing attendees to carry an Android tablet in order to view their position on a floorplan map, while walking around inside the demo area (interactive scenario). Moreover, we will illustrate how to evaluate the performance of different positioning algorithms using profiled data in a trace-driven scenario. Our objective is to highlight the effectiveness and applicability of our system and at the same time the participants will be able to appreciate the potential of indoor location-oriented services and applications.

I. Introduction

According to recent statistics people spend 80-90% of their time in indoors environments, including shopping malls, libraries, airports or university campuses, while 70% of cellular calls and 80% of data connections originate from indoors [1]. This has triggered an increasing interest in indoor Location-Based Services (LBS) and location-aware applications, e.g., in-building guidance and navigation, asset tracking, elderly support for Ambient and Assisted Living (AAL), etc. To enable such applications and facilitate their wide acceptance alternative solutions are required for the provision of accurate and reliable location estimates because satellite-based positioning, e.g., GPS, is unavailable or significantly degraded inside buildings due to the blockage or attenuation of the positioning signals. This has motivated research over the last 15 years and has led to the development of several positioning systems that rely on a variety of technologies, including infrared, Bluetooth, Radio-frequency identification (RFID), Ultra-wideband (UWB), video cameras, ultrasound, Wireless Sensor Networks (WSN) and Wireless Local Area Networks (WLAN); see [2] and references therein for an overview of positioning technologies and commercial systems.

Most of these technologies can deliver a high level of positioning accuracy, however they usually require the deployment and calibration of expensive equipment, such as custom transmitters and antennas, which are dedicated to positioning. This is time consuming and implies high installation costs, while the user needs to carry specialized equipment in order to perform positioning. Therefore, several positioning systems rely on WLANs owing to the wide availability of existing infrastructure, i.e., WLAN Access Points (AP), that provide ubiquitous coverage inside buildings. Location can be inferred at a room level (i.e., the positioning error is in the order of 2-4m) by exploiting Received Signal Strength (RSS) values extracted through passive scanning of the beacon packets transmitted by neighboring APs as part of the standard network functionality. Thus, commercial mobile devices, such as smartphones and tablets, can be used without any hardware modifications because they are usually equipped with WLAN adapters for wireless connectivity and only a software agent is needed for monitoring the RSS values.

To address the challenging signal propagation conditions indoors, due to multipath, reflections and diffractions, RSS fingerprints (i.e., vectors of RSS measurements recorded from APs in the vicinity of the user) have been studied extensively in the literature; see [3] for a detailed survey on fingerprint-based algorithms and systems. This approach has two phases; in the setup phase a number of RSS fingerprints are collected a priori at some predefined reference locations using a commercial device. Each RSS fingerprint is associated with the respective reference location and is stored in the so called radiomap that covers the whole area of interest. Essentially, the radiomap is a mapping from the multi-dimensional RSS space to the physical coordinates (x, y). Later during the positioning phase the currently observed RSS fingerprint is used to find the best match from the reference fingerprints in the radiomap and the unknown user location can be determined as the location associated with the best matching fingerprint.

Even though there are some fingerprint-based positioning systems in the literature, including the *Place Lab* [4], the *Redpin* [5] or the *Molé* [6] systems, the increasing market of Android devices has received limited attention as a development platform. Our contribution is Airplace, an end-to-

[1] Screenshots, a video demonstration and a video tutorial of the Airplace system are available at: http://www2.ucy.ac.cy/~laoudias/pages/platform.html

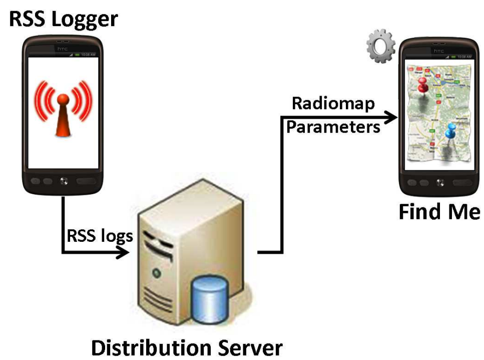

Fig. 1. The Airplace Positioning System Architecture.

Fig. 2. Screenshots of the Airplace *RSS Logger* prototype application.

end mobile positioning platform developed on Android smartphones that facilitates the construction of the radiomap and features real-time positioning, while offering the additional option to assess the performance of different fingerprint-based algorithms with respect to the positioning time, accuracy and power consumption[2]. In this demonstration paper we start out by presenting the system architecture and providing the details for the components in our platform. Then, we describe the demonstration settings including the equipment, as well as the interactive and trace-driven scenarios that we will follow to present our system to the conference attendees.

II. Airplace Positioning System Architecture

The prototype system is developed on Android smartphones and follows a mobile-based network-assisted architecture, which has two highly desirable properties. First, the communication overhead is kept low and network congestion, in case of continuous positioning requests by multiple users, is avoided because the observed RSS fingerprint is not uploaded to the network side (i.e., Positioning Server) for estimating location, as in the case of a network-based mobile-assisted architecture. Second, security concerns are addressed and user privacy is respected because the location is estimated by the user. A typical positioning scenario comprises three steps: (i) a user enters an indoor environment, such as a university campus or a shopping mall, covered by several WLAN APs, (ii) the user's smartphone obtains the RSS radiomap and parameters from the local distribution server in a single communication round and (iii) the client positions itself independently using only local knowledge, i.e., the downloaded radiomap and the currently observed RSS fingerprint, and more importantly without revealing its personal state.

The Airplace system architecture is illustrated in Fig. 1 and consists of the *RSS Logger* and *Find Me* applications, as well as the *Distribution Server* that are detailed below.

A. RSS Logger Application

The *RSS Logger* is an application developed around the Android RSS API for scanning and recording data samples in specific locations at predefined intervals. These samples contain the MAC addresses and RSS levels (in dBm) of all neighboring WLAN APs, as well as the coordinates of the location where the user initiated the recording. The start-up screen with the basic functionality of starting/stopping the scan process and setting the preferences (e.g., the floorplan map, the log filename, etc.) is shown in Fig. 2 (left). First, the user defines the number of samples to be collected at each location (Fig. 2, middle) and then moves to the desired location while setting the ground truth in local coordinates (x, y) by clicking on the map (with zooming capabilities). Finally, the user records the data, as shown in Fig. 2 (right). In this way the collected fingerprints are associated with the corresponding physical locations and the RSS radiomap reflects the indoor environment.

The collected data are stored locally in log files and users can contribute their data to our system for building and updating the radiomap through crowdsourcing. Specifically, the users can set the IP address and port number of our *Distribution Server* in order to upload the desired log file using a text protocol. In the future we plan to automate this process by allowing users to provide their log files anonymously whenever a data connection is available.

B. Find Me Application

Our positioning application is a client that runs on Android smartphones and connects to the server in order to download the radiomap and algorithm-specific parameters, thus enabling the user to self-locate independently thereafter. The interface of the *Find Me* application is shown in Fig. 3 (left), where the user can set the preferences (i.e., floorplan map, radiomap download settings, etc.) and select any of the available positioning algorithms from the *Algorithms* configuration panel. We have implemented several fingerprint-based algorithms, including the deterministic *K-Nearest Neighbor (KNN)* and *Weighted K-Nearest Neighbor (WKNN)* algorithms [8], [9], as well as the probabilistic *Maximum A Posteriori (MAP)* and *Minimum Mean Square Error (MMSE)* algorithms [10], [11]. Our platform additionally supports two state-of-the-art algorithms developed in-house, namely the *Radial Basis Function (RBF)* networks [12] and the *Subtract on Negative Add on Positive (SNAP)* [13] approaches.

[2]An early version of our Airplace system was recently demonstrated inside a conference venue [7].

Fig. 3. Screenshots of the Airplace *Find Me* prototype application.

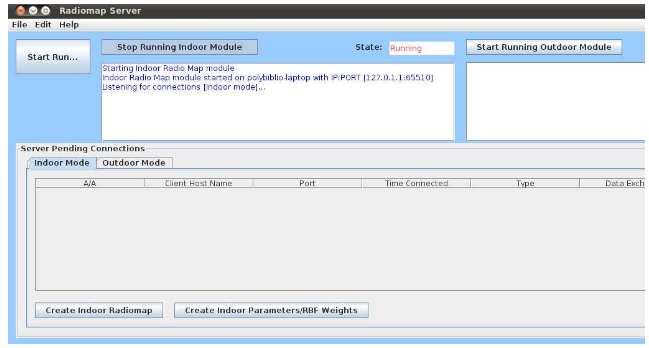

Fig. 4. The Airplace *Distribution Server* listening for connections.

Our positioning application supports dual operation mode for the provision of real-time positioning information (*Online Mode*) and the evaluation of different algorithms using prerecorded test fingerprints (*Offline Mode*). In the *Online Mode*, after downloading the radiomap and selecting an algorithm, the user can perform an one-time positioning by pressing the *Find Me Indoor* button and the estimated location is plotted on the floorplan map. Alternatively, the *Track Me* button can be switched on to initiate continuous positioning requests for tracking the user while walking inside the building. In this case, the current location estimate (green circle) is updated approximately every one second, while the past locations are shown as red dots; see Fig. 3 (middle).

In the *Offline Mode*, we offer the option to simulate a large number of successive positioning requests (i.e., a file with several RSS fingerprints collected at different locations can be stored on the flash media, as opposed to be collected in real time). This mode, shown in see Fig. 3 (right), is rather useful for loading test fingerprints collected a priori (e.g., a separate RSS log file collected with the *RSS Logger* application inside the area of interest) in order to assess the performance of different algorithms in terms of the following criteria:

- **Execution Time:** Measure the average time required in practice to perform positioning on smartphones.
- **Positioning Accuracy:** Display the average positioning error pertaining to the test fingerprints.
- **Power Consumption:** Investigate the battery depletion by using the PowerTutor utility [14] to measure the actual power consumption on smartphones during positioning.

C. Distribution Server

Our server is mainly responsible for the construction and distribution of the RSS radiomap by listening for connections from clients that either contribute the collected RSS data or request the radiomap and algorithm parameters to start positioning; see Fig. 4. To create the radiomap file the server parses all available RSS log files, which may be contributed by several users, and calculates the mean RSS value per MAC address by averaging over all samples collected at each distinct location. Finally, all averaged values are merged and stored in a single radiomap file so that each line corresponds to the mean RSS values at a specific location and each column to the respective MAC address. With this preprocessing (i.e., averaging) we alleviate the effect of noise (i.e., RSS values fluctuate over time even at the same location and do not usually follow well known distributions) and outlier values (e.g. erroneous RSS measurements due to transient behavior of the WLAN card). Moreover, with averaging the RSS radiomap is effectively compressed, thus reducing the user perceived delay for the system start-up (i.e., delay for receiving the radiomap) and decreasing the location estimation time. For instance, in our 560m^2 experimental setup at KIOS Research Center with 9 WLAN APs we collected 40 fingerprints per location at 105 distinct reference locations. After preprocessing these data with our server the size of the radiomap is around 32KB and takes only few seconds to download.

Another interesting feature in our server implementation is a methodology for fine tuning algorithm-specific parameters. For example, selecting an appropriate value for K in the KNN method is not trivial and may considerably affect the performance of the algorithm. We handle this issue by first using the *RSS Logger* to collect additional RSS samples for validation. Then, the KNN algorithm runs on the server side with different values of K and the appropriate value is selected in order to minimize the mean positioning error pertaining to the validation data. A similar approach is followed for all algorithms in our algorithm bank. In this fashion, our server first builds the RSS radiomap and then selects appropriate values for the parameters storing them in a separate configuration file which is also distributed with the associated radiomap. Note that in a deployed version of our Airplace system this fine tuning process could also be used to define the optimal algorithm for a specific environment, so that the best option is selected automatically in the *Find Me* positioning application without any user intervention.

III. Demonstration Settings

A. Equipment

For the demonstration of our indoor positioning platform at the conference venue we will use a Motorola Xoom tablet, with a 10.1´ screen (i.e., much larger than a typical 3.7´-4.3´ smartphone to facilitate presentation), an Nvidia Tegra 2 T20 processor (1.0GHz) with 1GB of RAM running Android 3.1 Honeycomb. Our Xoom tablet is equipped with multiple wireless communication modules for voice and data connections, featuring Quadband GSM/GPRS/EDGE as well as Bluetooth 2.0, however the tablet clients will rely on its built-in 802.11b/g WLAN transceiver to connect to the *Distribution Server*, either for uploading the collected fingerprints through the *RSS Logger* application or downloading the radiomap and the related algorithm parameters through the *Find Me* application. The *Distribution Server* will be running on a linux-based workstation and the clients will be able to connect to it though the WLAN hotspots installed in the conference venue. Alternatively, if no wireless connectivity is available during the demonstration, the RSS fingerprints can by manually copied from the smartphone's sdcard to the laptop running the server through the USB cable in order to create the radiomap and parameter files and then copy these files back to the smartphone so that positioning can be carried out.

To make the demo more appealing for the participants, a floorplan map of the demo area in .jpg format will be required for facilitating the collection of the data for the radiomap and displaying the location estimates during positioning. We will exploit the available WLAN infrastructure of the conference venue, as well as neighboring WLAN APs installed in other buildings in near vicinity. Note that for positioning with the *Find Me* application no active data connection is required because the client only passively scans the neighboring WLAN APs to get the required RSS values. In case there is a limited number of APs detected we will deploy a couple of HTC Desire smartphones that we own, using the readily available Portable WiFi functionality, to improve coverage and deliver a satisfactory level of positioning accuracy; this practice was tested successfully in previous demonstrations [7]. Finally, we might also use a projector along with a display export utility to present the interactions on a smartphone directly on a wall.

B. Demonstration Scenarios

Interactive: First, we will hand out the Android tablet to selected participants and ask them to collect a few RSS samples inside the conference venue and contribute them to the system in order to experience the *RSS Logger* application. Note that our team will have collected adequate samples before the demo, covering the whole demo area, to guarantee good performance. Next, the participants may start positioning themselves with the *Find Me* application in the *Online Mode* of operation using the currently observed RSS fingerprint. The participants will be able to view their current estimated position on the floorplan map of the venue by pressing the *Find Me Indoor* button and compare it with their actual location. Alternatively, they may switch the *Track Me* button on and start walking around in the demo area in order to compare their estimated trajectory (i.e., successive position estimates displayed as red dots on the floorplan map) with the traveled path. In this interactive scenario, our objective is to demonstrate the efficiency of our indoor positioning platform both for setting up and using the system, as well as the effectiveness of the underlying positioning algorithms.

Trace-driven: In this scenario we will demonstrate how our Airplace system can be utilized to evaluate different fingerprint-based positioning algorithms under the same conditions in terms of execution time, positioning accuracy and power consumption. Our team will have collected additional test fingerprints inside the demo area that will be employed in the *Find Me* application running in *Offline Mode*. The attendees will have the opportunity to see and compare the performance of different algorithms in the demo area, while our team provides the details and explains the trade-offs related to each algorithm. Our goals in this scenario are twofold; give the participants a better understanding of how WLAN RSS fingerprint-based positioning works in practice and illustrate how to use our system to get an early idea of what is the expected positioning error in the area of interest, in order to decide whether more WLAN APs should be installed or more dense fingerprints should be collected for the radiomap.

Acknowledgements: This work was supported by the Cyprus Research Promotion Foundation and by the fifth author's Startup Grant, funded by the University of Cyprus.

REFERENCES

[1] Source: Strategy Analytics http://www.strategyanalytics.com.

[2] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, pp. 13–32, 2009.

[3] M. Kjærgaard, "A taxonomy for radio location fingerprinting," in *LoCA*, 2007, pp. 139–156.

[4] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit, "Place lab: Device positioning using radio beacons in the wild," pp. 116–133, 2005.

[5] P. Bolliger, "Redpin - adaptive, zero-configuration indoor localization through user collaboration," in *1st ACM international workshop on Mobile Entity Localization and Tracking in GPS-less environments (MELT)*, 2008, pp. 55–60.

[6] J. Ledlie, J.-g. Park, D. Curtis, A. Cavalcante, and L. Camara, "Molé: A scalable, user-generated WiFi positioning engine," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011, pp. 1–10.

[7] C. Laoudias, G. Constantinou, M. Constantinides, S. Nicolaou, D. Zeinalipour-Yazti, and C. G. Panayiotou, "A platform for the evaluation of fingerprint positioning algorithms on android smartphones – demos," in *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2011.

[8] P. Bahl and V. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *IEEE INFOCOM*, 2000, pp. 775–784.

[9] B. Li, J. Salter, A. Dempster, and C. Rizos, "Indoor positioning techniques based on wireless LAN," in *1st IEEE International Conference on Wireless Broadband and Ultra Wideband Communications*, 2006, pp. 13–16.

[10] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *ACM MobiSys*, 2005, pp. 205–218.

[11] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen, "A probabilistic approach to WLAN user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, Jul. 2002.

[12] C. Laoudias, P. Kemppi, and C. G. Panayiotou, "Localization using radial basis function networks and signal strength fingerprints in WLAN," in *IEEE GLOBECOM*, 2009, pp. 1–6.

[13] C. Laoudias, M. Michaelides, and C. Panayiotou, "Fault tolerant fingerprint-based positioning," in *IEEE International Conference on Communications (ICC)*, 2011, pp. 1–5.

[14] PowerTutor: A Power Monitor for Android-Based Mobile Platforms http://powertutor.org/.