A *Bluetooth Smart* Analyzer in iBeacon Networks

Maria Varsamou and Theodore Antonakopoulos
University of Patras
Department of Electrical and Computer Engineering
Patras 26504, Greece
e-mails: *mtvars@upatras.gr* and *antonako@upatras.gr*

***Abstract*—Bluetooth Low Energy (BLE) is a newly emerged technology targeting low-power, low-cost wireless communications within medium or short range. BLE has extended the already wide acceptance of Bluetooth and is an ideal choice for a variety of sensor-based products, as well as ubiquitous mobile devices. Recently, it has been proposed that indoor positioning can be achieved when scattering BLE sensor devices, called iBeacons, in various locations of a particular venue. In this work, we present an Android-based application for analyzing iBeacon networks and determining the best signal map. We also demonstrate the application of the presented analyzer for measuring the radiation pattern of iBeacon devices.**

I. Introduction

Bluetooth is a ubiquitous low-cost wireless communication technology used for short-range connectivity, e.g. laptops with keyboards, mice, headsets. It was originally designed for continuous, streaming data applications and has successfully replaced wires in many consumer, industrial and medical monitoring applications. Today, the increasing use of mobile devices along with the constantly increasing appearance of low-cost sensor devices and actuators powered by coin-cell batteries has opened new areas for wireless applications, such as "Internet of Things" and Wireless Body Area Networks (WBANs). Since the extremely low-power communication requirements imposed by these applications cannot be met with the power-hungry original Bluetooth, in 2010, the Bluetooth Special Interest Group (BT SIG) incorporated an extension of the Bluetooth standard in Version 4.0 of its specifications, named Bluetooth Low Energy (BLE) [1].

Although BLE reuses the majority of the components of the classical Bluetooth, its physical layer has been redesigned and its state machine has been simplified to reduce power consumption and support asynchronous communications. With these changes, the standby power consumption of BLE devices is ultra-low and the operating power consumption is significantly reduced. BLE is ideal for applications requiring infrequent or periodic transfers of small amounts of data and can be used in a wide range of medical, industrial, and consumer applications. Compared with other similar technologyies, like Zigbee [2], BLE is more suitable for low-complexity devices with small data throughput, expected to operate for years on a coin-sized battery. Therefore, it is a leading candidate for the newly emerged market of "appcessories", i.e. small and simple devices that can serve as peripherals for smartphones and other mobile devices [3]. One category of "appcessories" includes various biometrics sensors that are used in health-care, sports and fitness applications. Another type of applications is related with proximity sensing, where one or more coin-cell battery powered devices are paired with a host device, in a way that enables the host device to either find misplaced and out-of-sight objects (proximity sensors) or raise an alarm when a selected object travels more than a specified distance from the host (security tags). Physical proximity is estimated using the receiver's RSSI (Received Signal Strength Indicator) value. Since signals traveling in a space reduce non-linearly in strength with respect to the traveling distance, the received signal strength can be measured at the receiver's site and after a few measurements the direction of relative movement can be estimated with some confident. RSSI is a commonly used metric for such applications in wireless sensor networks [4].

The proximity sensing capabilities of BLE-enabled devices has inspired Apple to propose an indoor proximity system based on such devices. So, in 2013, Apple announced that iOS 7 included the BLE-based indoor proximity technology, called "iBeacon" [5], which allows a mobile device (iPhone, iPad, iPod, etc.) to understand its position by knowing how close it is to low-complexity, low-cost, wireless transmitters called "hardware iBeacons" (or just "iBeacons"). Each iBeacon transmits periodically short-duration identification packets that are received by the mobile device and the distance between the device and the iBeacon is estimated by using the measured RSSI value. It is obvious that the efficiency of this novel indoor proximity system depends heavily on the accuracy of the RSSI values observed for each iBeacon. Unfortunately, RSSI measurements are significantly error-prone and are strongly influenced by the surrounding environment and the transmission system used, mainly the transceivers and antennas of the communicating devices [4]. Therefore, during the installation of iBeacons in a venue, to ensure accurate region positioning by the user's mobile device, an RSSI scene analysis of the infrastructure should be performed.

In this work, we present an Android-based application that serves as an iBeacon Analyzer. This tool collects statistics regarding the temporal and spatial variation of the RSSI values observed in the Android-enabled mobile device for all iBeacons it can detect. Various sets of measurements can be collected by placing the mobile device in different places and orientations. The tool provides also the possibility to send the measurements to a host computer where they can be analyzed by a suitable high-level software tool. Therefore, it can serve as a basis for developing advanced applications, for example, an area signal mapping tool or a radiation pattern analysis tool for iBeacons, as the one presented in Section V. Section II gives a concise description of BLE, while Section III gives a detailed description of the iBeacon technology. The iBeacon Analyzer is described in Section IV, while Section V presents a radiation pattern analysis tool for commercially available iBeacons.

II. Bluetooth Low Energy

BLE operates in a license-free ISM spectrum band (2402-2480 MHz), divided in 40x2MHz physical channels and uses GFSK modulation, achieving a data rate up to 1 Mbps [6]. Three advertising channels (i.e. 2402, 2426, and 2480 MHz) are defined for broadcasting purposes, such as device advertising. The other 37 data channels, are used for data transmission. A major advantage of BLE over the classical Bluetooth is that it has been designed to support both connectionless and connection-oriented applications. Depending on the required functionality, a BLE device may operate in different modes, i.e. advertising when it has to notify its presence, scanning when detecting of smart devices in its vicinity has to be achieved, connected as a master or a slave when data transmission is desirable. When a connection is required, a BLE master device may connect to unlimited slave devices in a star topology piconet, thus outperforming the classical Bluetooth, which only supports seven active slaves in a piconet. Figure 1 shows a potential BLE network. The sensor nodes can be a variety of smart devices and a smartphone can be used as the router to provide Internet connectivity.

The Bluetooth Low Energy defines several profiles for low energy devices. Each profile is a specification for how a device works in a particular application. Although manufacturers are expected to implement the appropriate specifications for their device in order to ensure compatibility, they are also allowed to define their own protocols for custom applications. A device may contain implementations of multiple profiles. Two commonly used profiles are the FMP -“Find Me” Profile- and the PXP -Proximity Profile- that rely on the receiver’s RSSI value to provide a distance estimation for proximity sensing applications.

Generally, these applications take into account that the RSSI value cannot provide an accurate distance estimation due to many reasons. For instance, the 2.4-GHz band, in which BLE operates, is the most prevalent one in wireless communications, used also by the ubiquitous 802.11/WiFi and other personal area network technologies, like ZigBee. To reduce interference by other technologies operating in the same band, BLE employs the frequency hopping mechanism, but this vulnerability cannot be totally eliminated. Furthermore, the radio signals quality depend on the operational characteristics of the transceivers, such as the antenna gains and the receiver sensitivity and they are affected by various environmental conditions and physical obstacles, like walls, moving persons, etc. Reflection, scattering, diffraction as well as the antenna gains produce significantly different path losses for equal distances. All the above means that the RSSI value from a BLE-enabled device presents significant variations in time and space [7].

III. iBeacon Indoor Proximity Technology

One promising proximity sensing application is the iBeacon Indoor Proximity technology proposed by Apple, where a BLE-enabled mobile device can determine its position in an area by knowing how close it is to strategically-placed simple wireless transmitters, named iBeacons. The advantage of this approach is that it can operate efficiently in indoor locations, where smartphones or tablets are not able to pick up reliable GPS signals. Moreover, GPS is not suitable for reliably determining distances within 15 to 20 meters. For that reason, some mobile applications also use triangulation of signals from cellular towers or WiFi hotspots to obtain the device’s geo-location, but the results are quite approximate regarding both stability and precision. Although iBeacon was initially proposed by Apple, since it is just an application that uses BLE, it is compatible not only with Apple devices, but can be also implemented by any smart device compatible with BLE, such as Android devices with Android 4.3 and above.

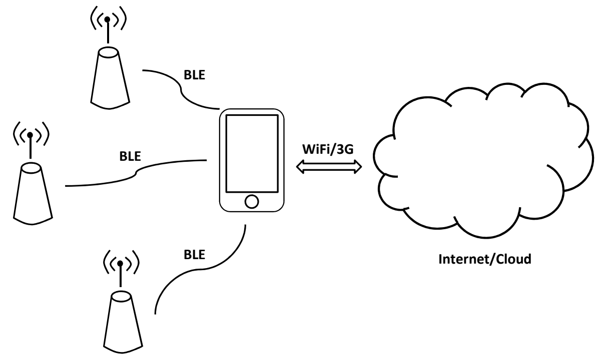


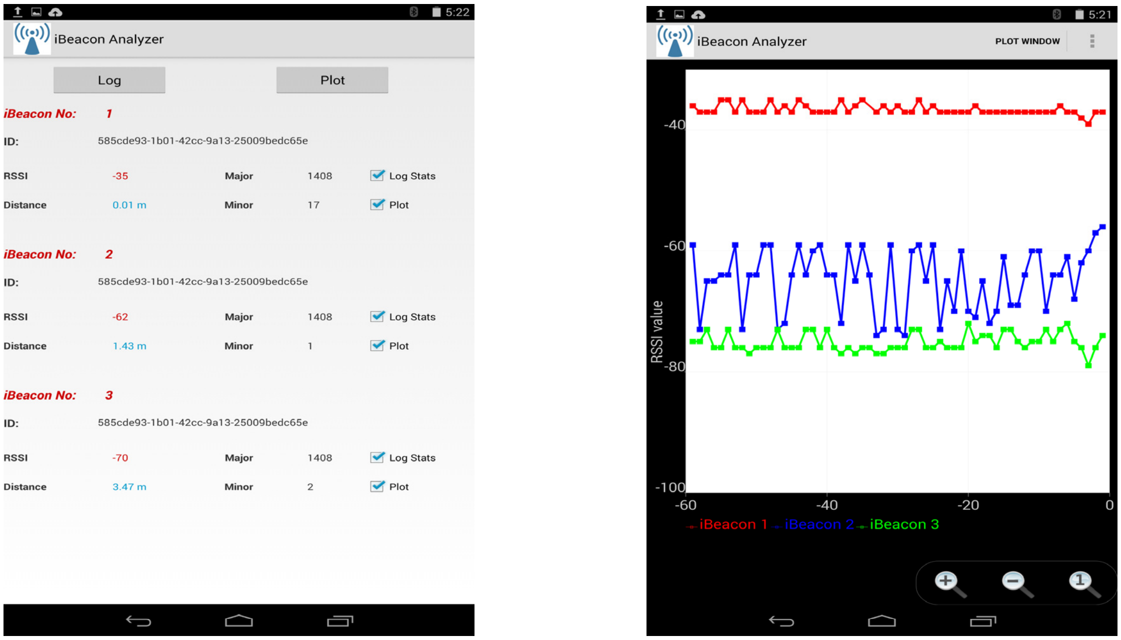
Fig. 1. A typical scenario of a BLE network in a star topology.

In more details, iBeacons are low-complexity transmitters that advertise a particular BLE payload with identifying information:

- *A proximity UUID (universally unique identifier):* 128-bit value that uniquely identifies one or more iBeacons as being of a certain type or from a certain organization. This identifier is mandatory.
- *A major value:* 16-bit unsigned integer used to differentiate iBeacons that have the same proximity UUID. This value is optional.
- *A minor value:* 16-bit unsigned integer used to differentiate iBeacons that have the same proximity UUID and major value. This value is also optional.

iBeacons that advertise the same UUID form a beacon region. Then, when a mobile device with an iBeacon-enabled app enters the beacon region, it will receive a relevant notification. These apps can also monitor the relative distance to the beacon, using the RSSI value. iBeacons advertise their nominal RSSI (in dB) at one meter, so apps can monitor their proximity to the beacon by comparing the advertised RSSI with their own receiver’s RSSI. This allows apps to know where they are located, not in terms of a map location using longitude and latitude like GPS, but relatively to known points. Using strategically installed iBeacons, the mobile device’s location can be determined with an accuracy far higher than that of GPS.

Although the iBeacon has minimal communication with the user’s device, it can initiate specific app functionalities through delivery of specialized notifications. In this case, the actual app content is downloaded to the device from servers in a private or public cloud, via WiFi or cellular.



(a) iBeacon Identification

(b) iBeacon RSSI Graphical Representation

Fig. 2. The Android iBeacon Analyzer App.

Test applications of the iBeacon technology have already appeared in the retail market, i.e. iBeacons have been installed in all Apple US stores, along with some other department stores. iBeacons can also be employed in several other crowd-related venues, for example NFL has rolled out a iBeacon network in MetLife Stadium, with the intention to provide personalized advertisements to football fans during Super Bowl.

As mentioned before, the location of a mobile device relatively to a specific iBeacon is determined by its RSSI value. Since the location cannot be accurately estimated by this value, location regions are defined. Apple defines three ranges/regions: *Immediate* (less than 50 centimetres), *Near* (approximately between 50 centimetres and 2/5 meters) and *Far* (more or less between 2/5 meters and 30/50 meters). The temporal variations of the RSSI values for a specific distance may be so intense that may cause spurious notifications to the iBeacon-enabled mobile apps. To prevent that, special consideration is taken so that region events do not happen immediately after a region boundary is crossed. For example, Apple's iOS does not deliver region notifications until certain threshold conditions are met. Specifically, the user's location must cross the region boundary and move away from that boundary by a minimum distance and remain at that minimum distance for at least 20 seconds before the notifications are reported.

IV. The Android iBeacon Analyzer

It is obvious that the efficiency of the iBeacon Indoor Proximity System depends heavily on the quality of the RSSI measurements of each iBeacon. Since there is no deterministic way to calculate the propagation delays, path losses and interferences of radio signals in restricted spaces, we have implemented an Android iBeacon Analyzer tool that facilitates the RSSI scene analysis of any iBeacons infrastructure. Using this mobile application during the installation of iBeacons in a venue, their locations and certain operating characteristics, such as their transmit power and their advertising interval, can be selected in a way that ensures accurate region positioning by the user's mobile device.

The iBeacon Analyzer app collects statistics regarding the RSSI values observed in an Android-enabled mobile device for all the iBeacon devices that it can detect and associates time-stamps for synchronization purposes. It has been developed using the Android Studio development environment and is compatible with all BLE-compatible Android versions, i.e. 4.3 and higher. For the iBeacon protocol implementation, the open-source Radius Networks' Android Beacon Service library [8] was employed. Using this library, the app gets notifications when one or more iBeacons appear or disappear. Additionally, it gets a ranging notification update from all detected iBeacons once every second. For each iBeacon, this update includes the receiver's RSSI for the latest received advertising packet along with the resulting distance estimation, which is calculated in accordance to the advertised nominal RSSI at one meter. Based on these updates, the app provides real-time graphical representation of the evolution of the detected iBeacons' RSSI values. The app offers also the possibility to store the measurements from user-selected iBeacons and then share the log files either with a host computer or through the Internet with a remote host.

Figure 2 shows example screenshots of the iBeacon Analyzer app, when three commercial iBeacon devices, that belong to the same beacon region, are placed in three different spots in a room. The first screen presents the identifying information for each detected iBeacon. In each ranging notification, the measured RSSI values along with distance estimations are updated. The user can select the iBeacons for which he wishes to log statistics, as well as the specific iBeacons, whose RSSI he wishes to observe in the graph window. By pressing the “Log” button, the user initiates the statistics collection operation and a log file with a user-specified name is created in the storage area of the mobile device. Each ranging notification results in a record in the log file, including the time-stamp of this notification, the IDs of the detected beacons and their RSSI values. At the same time, the “Plot” button leads to the second screen where the real-time graphical representation of the selected beacons’ RSSI is presented.

V. iBeacon Radiation Patterns

The capability of the iBeacon Analyzer for dynamically storing and sharing data, i.e. RSSI values and time-stamp information, from multiple iBeacons along with its connectivity over Internet, gives the basic functionality for developing advanced applications. In this section we present a radiation pattern analysis tool for commercially available iBeacons. This tool is based on the presented iBeacon Analyzer and a high-level software for data analysis and presentation. Due to reciprocity, the measurement of radiation patterns presented in this section are the same for both the transmit and receive modes of operation.

Ideally, iBeacons have to use isotropic (omnidirectional) antennas in order to cover uniformly and irrespective of their orientation, the communications distance between the iBeacon and the mobile device, where the user’s application is running. In practise the isotropic coverage is not usually satisfied. The iBeacon antennas are printed-circuit antennas and the radiation pattern of the whole device is determined by the antenna itself and the iBeacon assembly. ICs, capacitors, connectors, other passive components along with the battery assembly also affect the device’s radiation pattern. The effective radiation pattern of such a device diversify according to the surroundings, due to reflections on adjacent objects. Usually iBeacon vendors do not provide the radiation pattern of their devices and accurate characterization of radiation patterns can be performed in electromagnetically anechoic chambers, an expensive and time consuming process. For practical usage of iBeacons, a less accurate measurement of their radiation pattern is also sufficient, and such a method is presented hereafter, where the iBeacon Analyzer along with a custom software running on a host computer can be used to estimate the radiation pattern of an iBeacon device, either in open space or in a specific indoor environment.

An iBeacon transmits periodically short-duration packets that are received by the iBeacon Analyzer and its RSSI value is estimated by measuring the power of the received signal. The measured RSSI value is affected by the transmitter’s output power, the transmitter’s antenna radiation pattern, the communications channel (if multiple reflections exist this is time-varying), the receiver’s antenna radiation pattern and the receiver’s sensitivity. By changing the orientation of the iBeacon device relative to the mobile device, where the iBeacon Analyzer is executed, and keeping all other factors as constant as possible (the communications channel is the most sensitive component), we can collect receive power measurements and by normalizing them (either to the maximum value or to the power received in a given orientation) we can estimate, with acceptable accuracy, the iBeacon’s radiation pattern.

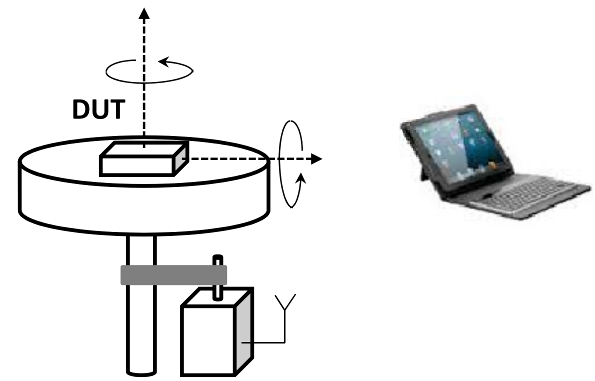


Fig. 3. The experimental set-up.

For that purpose, we developed a measurement method that is based on the iBeacon Analyzer. As it is shown in Fig. 3 the proposed method is based on a mobile device running the iBeacon Analyzer which is located at a fixed distance from the under test iBeacon (DUT). Initially the iBeacon is oriented so that its longitudinal axis is towards the mobile device. The iBeacon transmits the advertising packets and the iBeacon Analyzer collects a set of prespecified measurements. Then the iBeacon is rotated by a fixed angle at a given plane and a new set of measurements is collected. This process continues until a full rotation has been performed. Then the same process can be applied in a perpendicular plane and a new set of measurements is collected. At the end of this experiment, the data are transmitted to a host computer, where the high-level data analysis software is executed, and the iBeacon’s radiation pattern is estimated. It has to be mentioned that for accurate results, the measurements have to be performed in open-air and at maximum possible distance from walls and other obstacles. The above mentioned process can be performed either manually or by setting up an experimental set-up with a remotely controlled rotational table and by automating the whole process. The mobile device, where the iBeacon analyzer is executed, can remotely control the rotational table and start collecting measurements when the DUT is in the proper orientation.

Figs. 4 (a) and (b) show the measurements of a commercially available iBeacon on two perpendicular planes, following the experimental methodology described above. The horizontal plane corresponds to the pcb plane of the device. During this experiment, the rotation step was 10^0, RSSI was measured with 1 dB accuracy, the distance between the two devices was 2 meters, a set of 64 measurements was collected in each orientation and the time interval between successive measurements was 1 sec. Fig. 5 shows part of the data collection process and the respective measurement angles. The shaded intervals corresponds to transition phases between adjacent measurements. Fig. 4 (c) shows the radiation pattern of the same iBeacon, measured with the same method and the same tools, in an indoor environment. The effective radiation pattern is slightly different than the outdoor, actual pattern. It has to be emphasized that the proposed method is targeting commercial iBeacon deployments.

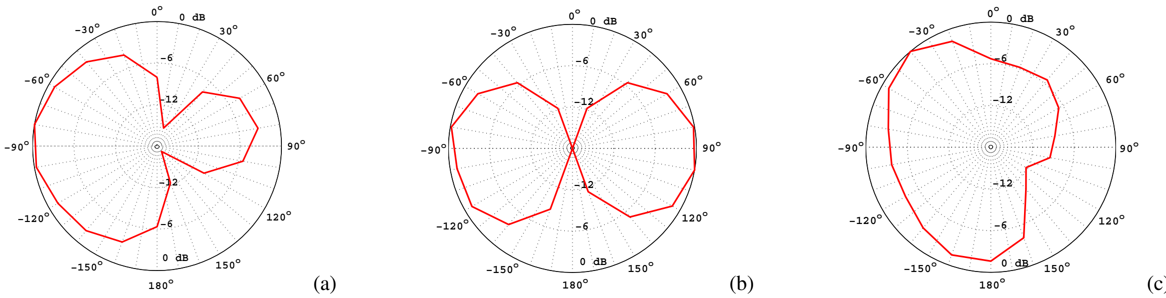


Fig. 4. The outdoor horizontal (a) and vertical (b) radiation patterns of an iBeacon, and an indoor horizontal (c) radiation pattern of the same iBeacon.

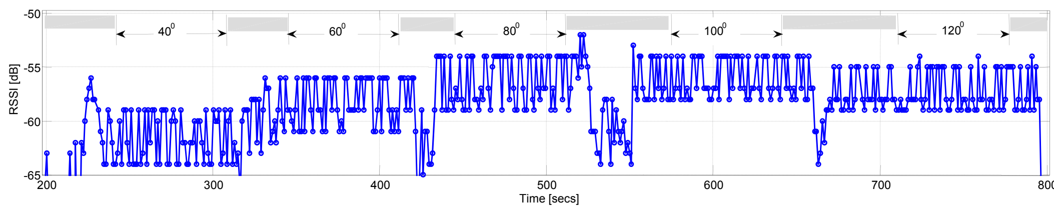


Fig. 5. RSSI measurements example for estimating radiation pattern.

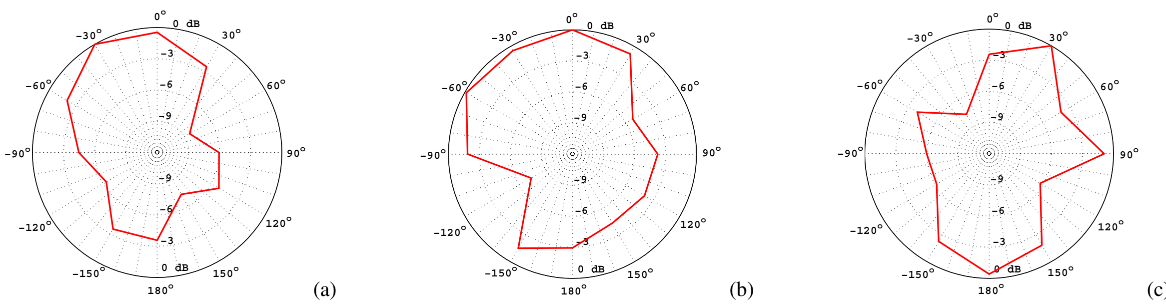


Fig. 6. Indoor radiation patterns of iBeacons from various vendors.

Fig. 6 shows the radiation patterns of three iBeacons from different vendors in the same indoor environment. These radiation patterns correspond to the horizontal plane, while the longitudinal axis of each iBeacon (direction opposite to the battery location) corresponds to 0^o. During this experiment, the rotation step was 30^o, while all other parameters remaining the same as previously described. From these measurements it is evident that different vendors have different radiation patterns and antenna directionality, while the omni-directionality is not usually satisfied.

The same methodology can be used for measuring the radiation pattern of a mobile device, like a tablet or a smartphone. In this case the iBeaconEmulator has to be used. iBeaconEmulator is an application that deactivates temporarily all wireless communications of the mobile device, except of the Low Energy Bluetooth which remains active, and using a user friendly interface the emulated iBeacon's ID and retransmission intervals are specified and the mobile device acts as a typical iBeacon device. Using the iBeacon Analyzer in another mobile device, as described previously, the mobile device under test can be measured and characterized in terms of its radiation pattern. Although currently this approach is applicable only to mobile devices of a specific vendor, we estimate that in the near future this capability would be feasible at most available mobile devices.

VI. Conclusions

We presented an Android-based software tool for collecting statistics regarding the temporal and spatial variation of RSSI values observed in an iBeacon network. We also demonstrated the application of this analyzer for measuring the radiation pattern of iBeacon devices. Experimental results show that the proposed measurements method provides a reliable and useful tool for analyzing iBeacon devices and installations.

References

[1] "Bluetooth Core Specification v4.1," Bluetooth SIG, Tech. Rep., 2013.

[2] "Zigbee specification 2012," ZigBee Alliance, Tech. Rep., 2012.

[3] J. Decuir, "Introducing Bluetooth Smart-Part II:Applications and Updates," *IEEE Consumer Electronics Magazine*, vol. 4, pp. 25–29, 2014.

[4] M. Ilyas and I. Mahgoub, Eds., *Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems.* Boca Raton, Florida: CRC Press LLC, 2005.

[5] A. Cavallini. iBeacons Bible 1.0. [Online]. Available: http://www.gaia-matrix.com

[6] R. Heydon, *Bluetooth Low Energy, The Developers Handbook.* Englewood Cliffs, NJ: Prentice Hall, 2012.

[7] S. M. S. Slijepcevic and M. Potkonjak, "Characterization of location error in wireless sensor networks: analysis and applications," in *The 2nd International Conference on Information Processing in Sensor Networks - IPSN'03*, Palo Alto, CA, USA, 2003, pp. 593–608.

[8] Radius Networks. Android Beacon Library. [Online]. Available: http://developer.radiusnetworks.com/ibeacon/android/