2016 IEEE/ACM Symposium on Edge Computing

# Smart LaBLEs: Proximity, Autoconfiguration, and a Constant Supply of Gatorade™

Albert F Harris III, Vansh Khanna, Güliz Seray Tuncay, and Robin Hillary Kravets
University of Illinois at Urbana-Champaign
{aharris,vkhanna2,tuncay2,rhk}@illinois.edu

***Abstract*—The availability of the low-power, simple communication model of Bluetooth™Low Energy (BLE) has resulted in the explosion of Internet of Things (IoT) enabled devices. IoT in the retail space has the potential to improve both user experience as well as business practices. For example, IoT-enabled retail systems could eliminate the need for human intervention to update product signage when product locations are shifted or in response to changes in product information (*e.g.*, price changes).**

**Such smart inventory systems would rely on the ability to automatically determine which products were nearest to relevant labels. In this paper, we present a detailed study of BLE channel characteristics using a battery of tests with real devices. We then present the design and implementation of Smart LaBLEs, our BLE-based, autoconfiguring product labels. Smart LaBLEs detect BLE-tagged products in their environment, determine the nearest shelved product, and autoconfigure a colored LCD with product information from the nearest products advertising message. The Smart LaBLEs act as decentralized IoT hubs, opening the door for product tags detected by the Smart LaBLE to reduce the frequency at which they send advertising messages, thus conserving bandwidth and energy.**


Fig. 1. Failed to Roll-back

## I. Introduction

Low-cost, low-power Internet of Things- (IoT-) enabled devices can enhance every-day objects with basic communication and computation abilities at the edge of the network. One major target for IoT is smart shopping, where retail spaces and inventory systems are outfitted with IoT technology to improve both the management of the store and the user's shopping experience. In this paper, we tackle a major problem for large stores: inventory control and management of pricing labels. In a traditional store, the movement or reshelving of products can result in major reconfiguration of product signage and labels. Such reconfiguration typically requires manual intervention to match the new product information. Unfortunately, human-driven updates can take a significant amount of time and often introduce errors into the displays (*see*, Figure 1[1]).

While there are numerous wireless tagging technologies to choose from, including near field communication (NFC) technologies such as those found in modern smartphones or RFID tags, BLE has entered the IoT community providing a number of benefits over such solutions. First, technologies such as NFC are limited to very short distances (approximately 2 in). Such a limitation does not support the wide variety of applications expected of IoT environments, where, for example, a shopper might want to get information from all products as they traverse an isle. Second, the amount of data transferable rapidly by these technologies is also too limited. While it is true that BLE devices require batteries, many such devices can run for years without a battery change (e.g., Estimote Beacons[2]).

To enable automated inventory management, we present the design and implementation of Smart LaBLEs: IoT-enabled inventory labels that monitor Bluetooth Low Energy (BLE)-enabled products, essentially acting as decentralized IoT hubs. Such IoT hubs provide edge services such as collecting and presenting data from numerous sensors without impacting or utilizing centralized servers. Specifically, Smart LaBLEs collect information from surrounding products and auto-configure to display up-to-date product information about the nearest product. Although the solutions presented in this paper focus on inventory control, they can be expanded to enhance the user's shopping experience as well. For example, nearby products could light up when they meet nutritional characteristics defined by a shopper's diet; gym equipment could give feedback and encouragement based on a personal trainer's suggestions; and products in big-box electronic stores could transmit extra information to a shopper's device when the shopper stops in front of a display. The common theme across all of these scenarios is monitoring nearby devices, and the

[1]Taken from http://www.rd.com/funny-stuff/top-15-funniest-signs/, last visited on Feb. 26, 2016)

[2]See, http://estimote.com/, last visited June 9, 2016

978-1-5090-3322-5/16 $31.00 © 2016 IEEE
DOI 10.1109/SEC.2016.43

challenge lies in defining what is nearest and collecting such information in a bandwidth- and energy-efficient manner.

One obvious solution for determining the nearest object is simply to apply traditional RSSI ranging techniques to find the location of all objects in the vicinity of a scanning device. Once such a map of the environment is built, the scanning device could employ simple algorithms to choose the nearest object. However, applying these traditional techniques is doomed to fail due to a number of factors related to the size and quality of the radios and antennas used in small IoT objects (*e.g.*, fast multipath fading [1], [2]).

Taking cues from traditional mechanisms used in the Wi-Fi setting, current attempts at solving the proximity problem in BLE begin by focusing on determining distances to each object in the environment and then attempting to define a distance-based ordering (*see, e.g.*, [3]). However, in many IoT scenarios, such as the retail inventory environment, exact distance is not important. Instead, determining pairings requires a different metric: the relative nearness of objects. Similar to the idea of Lamport clocks [4] in the time domain, in the physical object domain, it is not necessary to determine the exact location or physical coordinates of each object in the environment; instead it is only necessary to order the objects in terms of nearness to the scanning device.

Such nearness orderings based on BLE RSSI values require a good model of the wireless channel. Although there has been recent analysis of the BLE channel, focusing on reliability or the ability to resolve exact distances (*see, e.g.*, [2], [1]), building a system capable of choosing a nearest object in a real IoT environment requires a more comprehensive channel analysis of such BLE-based IoT environments. To fill this gap, we present an analysis of an IoT environment using real devices with experiments that control for various effects related to the channel, energy consumption, and device and antenna characteristics. These experiments and the channel analysis provide design boundaries within which the Smart LaBLE design (or any BLE-based IoT system) must exist to be effective.

Our analysis uncovers a number of characteristics regarding the BLE channel that define the design space of proximity-based systems. First, we show that the distance between a scanning device and the nearest BLE tag must be less than twice the distance between the nearest tag and adjacent tags. Second, we show that long-term averages of RSSI values do not produce accurate estimates of the nearest tag due to channel fluctuations. Third, we show that instantaneous values collected from tags with a small (less than a one second window) are sufficient to pick the nearest tag assuming the antenna of all tags are oriented correctly. Additionally, we show that performing averages over short windows (on the order of 500 ms) can smooth out fluctuations due to tag orientation, thus producing a system capable of functioning in real environments. Finally, our analysis provides insight into how the density of tags constrains the choices of advertising periods for each tag.

The second contribution of this paper is the design and implementation of the Smart LaBLE inventory system. Using the insights gained in our analysis, we designed and built Smart LaBLEs to sense products in their environment, determine which product is nearest based on running averages of 500 ms of RSSI data, determine how many of that product are in range of the Smart LaBLE, and then autoconfigure a color LCD display based on information in the advertising messages from that product. Our evaluation of the Smart LaBLEs show a false detection rate of approximately 1%. Furthermore, since the Smart LaBLEs act as decentralized hubs, once they are configured, there is no reason for each of the products to send frequent advertising messages. Thus, our system allows the conservation of significant energy and bandwidth by allowing the expected large number of tags in a retail environment to significantly reduce the frequency with which they transmit information.

The rest of this paper is organized as follows. Section II presents an overview of IoT environments and presents related work in proximity algorithms (Section II-A) and decentralized IoT hubs (Section II-B). Section III presents our experiments and in-depth analysis of the BLE-based IoT environment, including presenting the conclusions used to develop our Smart LaBLE system. Section IV present the design and implementation of Smart LaBLEs. Section V presents an evaluation of Smart LaBLEs. Finally, Section VI presents conclusions and future directions.

## II. IoT Environments

In IoT environments such as retail stores and gyms, physical objects outfitted with computation and sensing capability can be treated as resources to be discovered and managed through low-power wireless connections (*e.g.*, RFID, IEEE 802.15.4 (ZigBee), Bluetooth Low Energy (BLE)). BLE is specifically designed for IoT environments, and as such, provides similar communication ranges to the original Bluetooth devices with significantly lower energy consumption (*see, e.g.*, [5], [6]). These savings stem from the fact that BLE supports connectionless communication, where a tag periodically sends small updates via advertising messages without requiring receivers to pair.

By monitoring the object in a user's environment and the user's proximity to the objects, IoT applications can support a wide range of interactions between the user and the objects. Such proximity information is specifically relevant for IoT applications that require a user to be paired with a physical object. For example, in a smart gym environment, a user's fitness device may want to pair with an exercise machine with which the user is interacting. In the retail inventory environment, our Smart LaBLEs need to determine which product's information to display.

Previous work in smart retail environments was not focused on leveraging IoT-enabled physical objects to determine user interest [7]. Since nearness is a good first indicator of interest in physical objects, a scanning device (either held by a user as they navigate through an IoT environment or our Smart LaBLEs mounted on a shelf) must first determine which objects are nearby and then cull down that list of potentially many objects to determine which are of interest. In our smart inventory scenario, the Smart LaBLEs first monitor for all nearby products and then determine which product is nearest for displaying the correct product information.

Effective discovery and ranging introduce difficult challenges for both bandwidth, in terms of managing channel contention, and energy, in terms of minimizing the energy consumed by the BLE tags on the products. In a retail environment, if each product on a store's shelves is equipped with a BLE transmitter, hundreds of devices within range of a scanning device could be transmitting at the same time. Such density of BLE devices will quickly flood the wireless channel, increasing channel contention and decreasing the likelihood that a scanning device will successfully receive useful information. Additionally, changing batteries on hundreds of shelved products is simply impractical. Therefore, energy must be conserved to ensure that each product's battery is maintained until the product is sold. To address these challenges, each Smart LaBLE acts as a decentralized IoT hub that can take control of its associated products, allowing the IoT device on each product to dramatically reduce the frequency by which it sends periodic messages, thus conserving both bandwidth and energy.

### A. Proximity

Traditional ranging algorithms have been developed in a number of environments. For example, a large group of techniques have been built around the possibility of using dead reckoning, or other user-motion-tracking techniques (*e.g.*, through the use of accelerometers) in combination with perceived signal strength to accurately determine a user device's location. [8], [9], [10]. Other solutions rely on complex path loss models to predict user location based on fixes from one or more transmitting stations with known locations. [11], [12], [2], [1], [3]. These solutions generally rely on pre-calculated RF maps and fixed points with known locations. To get accurate location with BLE radios, RSSI is not sufficient [1] and more complex algorithms have been proposed that collect RSSI values at the scanning device but process it remotely in the cloud [3].

All of this prior work is focused on determining the quantitative location of a tag. This distance estimate is then used to build orderings of objects. Since BLE is not capable of providing accurate resolution without additional measurements and as discussed above, many IoT environments do not require absolute locations, Smart LaBLEs use relative distance to select the nearest product with a false positive rate of approximately 1% instead of attempting to determine actual distances to products.

### B. IoT Hubs

Although the concept of IoT has been around for a while, there has yet to be an established communication protocol or architecture to access the 'things' in an Internet of Things. Depending on the needs of the application, different technologies and architectures can be adopted, and this creates the need for an IoT hub (also called a gateway) that can aggregate information from heterogeneous IoT devices. These hubs can be used in many IoT scenarios, from home health care to smart grids to smart cities. Additionally, such hubs have the potential to act as decentralized sinks and controls, potentially allowing bandwidth and energy conservation in IoT systems.

The use of a Web of Things architecture has been proposed to support large scale data aggregation in IoT [13]. The IoT hub architecture operates based on web technologies such as HTTP and JSON as well as Representational State Transfer (REST) architecture for interoperability. The 'things' can simply communicate with the hub through the hub's exposed RESTful web services, and the hubs also can employ access control on the 'things'. The use of this architecture is demonstrated in smart city scenarios, where the sensor data from 'things' can help monitor different aspects of city life [14].

In health-related scenarios, one can imagine the use of many different types of biosensors (*e.g.*, heart rate, blood glucose, weight sensors, and respiratory meters) to monitor the health of individuals. Thus, an interoperable IoT hub is a critical component in such scenarios to deal with the extreme heterogeneity of biosensors. The Home Health Hub Internet of Things (H3IoT) architecture [15] supports interoperability by providing IoT devices access to the hub via several different communication protocols (*e.g.*, BLE, ZigBee, Infra Red, USB). Similarly, IoT hubs can be used in smart gym scenarios [16]. The IoT hub aggregates data about users' current workout (*i.e.*, acceleration) from the sensors placed on exercise machines (*e.g.*, lat pull down machine) and other exercise equipment (*e.g.*, dumbells).

IoT hubs also have received interest from some of the leading companies in tech industry. Companies such as Intel and VMware have produced their own versions for IoT gateways [17], [18], and are now competing in IoT business. However, most of this previous work only considers multi-technology communication. In comparison, our work focuses on the use of hubs to allow the management of IoT devices in an environment to conserve bandwidth and energy.

## III. Analyzing The IoT Environment

For Smart LaBLEs to function usefully, they must be able to reliably determine the nearest shelved product. However, finding that nearest product based on RSSI requires an in-depth understanding of the wireless environment and the space of BLE RSSI values. Furthermore, to understand the potential impacts on bandwidth and energy resources that the use of decentralized IoT hubs can have, it is important to understand the impact on both resources that large numbers of IoT Tags transmitting in a small environment might have. This knowledge can then be used to design and develop systems using currently available BLE transceivers. Such a system must take into account the channel conditions while leveraging information available from standard BLE devices. We begin by presenting a brief introduction to the BLE channel and then present our experiments and analysis.

### A. Bluetooth Low Energy - BLE

BLE operates in the 2.4 GHz band and divides this band into forty 1 MHz wide channels. According to the BLE specification, there are three advertising channels (37, 38, and 39) (*see*, Figure 2). The BLE discovery channels are positioned so as to not overlap with the primary IEEE 802.11 orthogonal bands (*see, id.*). The remaining channels are used as data channels for connected-mode communication.

BLE offers three basic modes of communication: passive scanning mode, active scanning mode, and connected mode. In passive scanning mode, the tags send advertising message periodically on each of the three advertising channels. Scanning devices simply listen for advertising messages in their area, never transmitting any information to the tags. In active scanning mode, again tags send a advertising message periodically on each of the advertising channels; however, scanning devices can respond with a scan request message that triggers one additional message from the tag on the advertising channel of interest. Active scanning effectively adds 31 bytes to the amount of data that can be transmitted by a tag without the need for full connection establishment. Finally, connected mode allows a scanning device to connect to a tag and receive an arbitrary amount of data. For the purposes of our Smart LaBLE system, we assume the relevant product information could be transmitted in the initial advertising message sent in passive scanning mode; however, extension to active scanning mode or connected mode is straightforward.

In passive scanning mode, each BLE tag sends an advertising message on each of the advertising channels (37, 38, and 39, in that order) (*see*, Figure 3) during each advertising period. Thus, one advertising period includes three identical advertising messages. The time between each advertising period is adjustable. However, in normal operation, if advertising periods are frequent, the wireless environment becomes flooded with interference and few advertising messages from any of the tags in an environment can be decoded successfully. This effect is particularly dramatic in tag-dense environments. Generally, advertising periods during normal operation are spaced around 750 ms. As an additional mechanism to help prevent the collision of advertising messages, each tag randomly adds up to 10 $\mu$s of jitter to the transmission time of each advertising message. Each peer device transmits 150 $\mu$s after detecting the last packet.

The BLE packet format, depicted in Figure 4, includes a Packet Data Unit (PDU) that contains packet formats for each of the modes discussed above. Passive scanning mode is indicated by setting the first four bits of the PDU header to `0010` (`ADV_NONCONN_IND`). The length field in the PDU header indicates the size of the variable length payload. For passive scanning mode, the maximum payload size is 31 bytes. Thus, in terms of transmit times, the minimum packet size is 80 $\mu$s and the maximum packet size is 328 $\mu$s.

### B. Tag Transmission Propagation Analysis

Previous analyses of the channel used by BLE either focused on theoretical or simulation results [19], [20], focused on energy consumption and (as opposed to more proximity-relevant metrics) [21], interference with other technologies [22], or focused on specific environments not relevant to a retail BLE environment (*e.g.*, body networks [23], [24], vehicular networks [25]). To analyze the radio environment expected in our target IoT scenario, we utilized the nRF51822 Bluetooth Smart Beacon Kit [26] ("tags") produced by Nordic Semiconductor as IoT-enabled objects. Nordic Tags are coin-sized BLE devices that operate with a 3 V battery (*see*, Fig. 5).

The tags have transmit power control ability [26]. To test the impact of transmit power on the ability to determine proximity, we use three transmit power settings: 0 dBm (1 mW) - High; -8 dBm (0.158489 mW) - Medium; and -16 dBm (0.025119 mW) - Low. Given our goal of both accurately choosing the closest object and preserving channel and energy resources, the ability to adjust a tag's transmit power level yields another possible variable to adjust in any protocol.

We performed our experiments with two main layouts for the placement of tags and the scanning device: circular and linear. Each of these cases helps unveil different aspects of the BLE channel. In the circular case, the tags are evenly placed around a circle, and the goal is to identify the effect of orientation on signal strength. As for the linear case, the tags are placed linearly on a wall with even gaps between them. The idea in this case is to explore the limits of the use of instantaneous RSSI values for nearness determination.

*1) Circular Scanning:* We began our analysis of the BLE environment with two experimental configurations: (*see* Figure 6):

(a) *Tags in a circle around the scanning device*: Twelve objects are placed with 30° separation around a circle. The scanning device sits in the middle with its antenna oriented closes to the tag at location 1. The experiment is repeated with radii of 25 cm, 50 cm, and 100 cm.

(b) *Scanning device in a circle around an object*: One tag is placed in the middle of a circle. The scanning device is placed at 30° increments around a circle centered on object with its antenna closest to the center of the circle at all times. The experiment is repeated with radii of 25 cm, 50 cm, and 100 cm.

We evaluated the circular configurations to account for irregularities in the propagation patterns from both the tags and the scanning device. Essentially, in a real retail environment, all tags may not be on a single side of a particular smart label. In fact, if one considers a typical retail shelf, one would expect products to be located all around a particular label.

In addition to the circular layout, we also varied the advertising period and the transmit power for each of the tags. The advertising period was set to five different values: 100 ms, 250 ms, 500 ms, 750 ms, and 1 s (recall that conventional wisdom sets the advertising period to 750 ms). The tags were configured to utilize each of the three transmit power

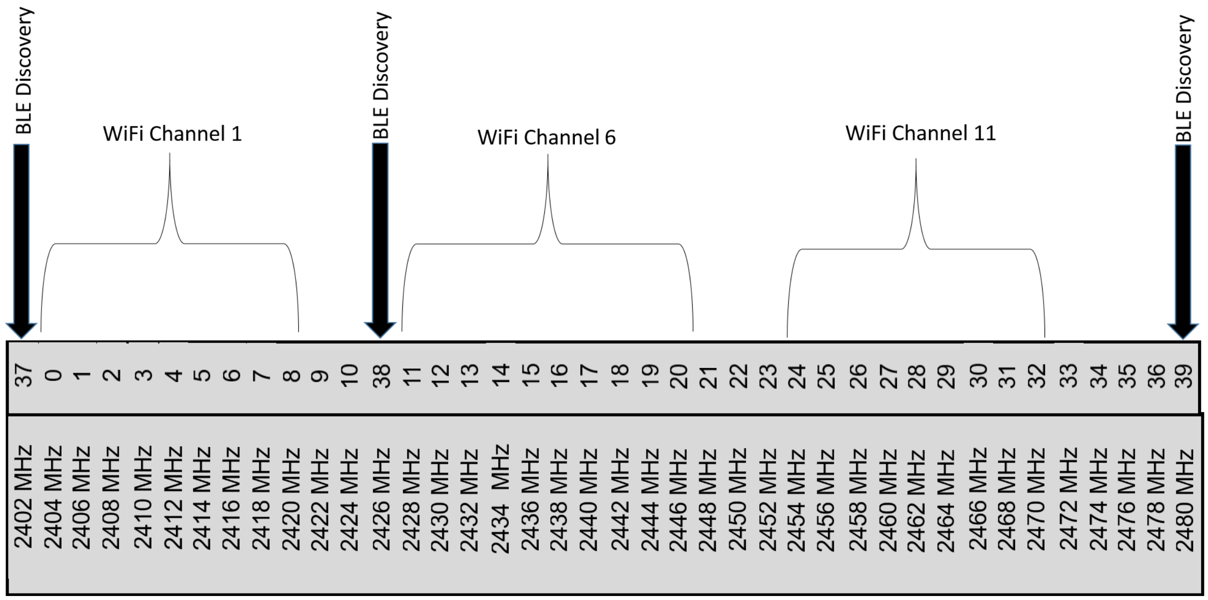


Fig. 2. The 2.4 GHz Channel

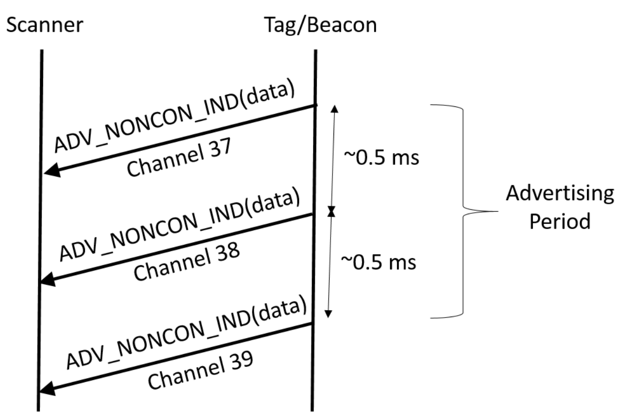


Fig. 3. Advertising Period

levels in two different regimens. First, the tags used each transmit power level for 15 seconds and then transitioned to the next transmit power level (*see* Figure 7(a)). Second, the tags changed transmit power level at every advertising period (*see*, Figure 7(b)). The scanning device listened for advertising messages from all tags in each scenario and recorded the time received, the advertiser's MAC address, and perceived RSSI for all advertising messages.

**Central scanning device**. Figure 8 depicts the average RSSI for tags in a circle around the scanning device (see Figure 6(a)). The tags are each using the high transmit power to send advertising messages. Each line represents a different circle radius. A number of observations become instantly apparent. First, over the long run, it is generally impossible to use RSSI measurements at the scanning device to determine a nearness ordering. First, averaging (in this case over 15 secs) does not work, as is apparent because the lines themselves cross. Second, arbitrary instantaneous values will not effectively produce an accurate nearness ordering, as is apparent because the error bars overlap. A second observation is that the same types of tags with new batteries can produce different signal strengths. This could be due either to the orientation of the scanning device's antenna with respect to the tag, the differences in the propagation properties of each individual tag's antenna, or both. This is apparent because each of the lines are not horizontal, as would be expected.

Furthermore, reducing the transmit power to the lowest setting does not change the outcome (*see* Figure 9). As we would expect, the relative distances between lines is maintained, with the RSSI values overall decreasing proportionally with the transmit power decrease. We omit the results depicting the medium transmit power experiments as they completely mirror what would be expected, falling between the two sets of results presented here.

**Central scanned object**. To attempt to isolate some of the effects of antenna orientation, we ran sets of experiments with a single tag and single scanning device. In these experiments, we rotated the scanning device around the tag, keeping the orientation of the scanning device constant through the rotations (*see* Figure 6(b)). Again, transmit power had no interesting effects on the results; therefore, we present the medium transmit power experiment herein (*see* Figure 10).

First, these results show again that long-term averaged RSSI values (here again, over 15 s) and arbitrarily-spaced instantaneous values are not sufficient for determining a nearness ordering. Another important observation from these experiments is that tag orientation itself can greatly impact the RSSI. In fact, when the scanning devices is located at position 6, the RSSI values at the farthest distance tested are

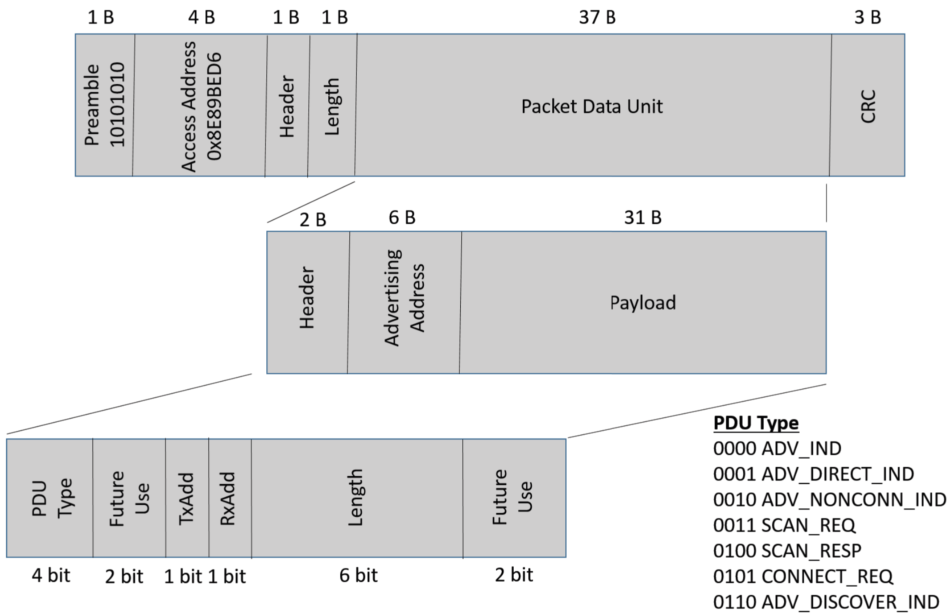


Fig. 4. BLE Packet Format

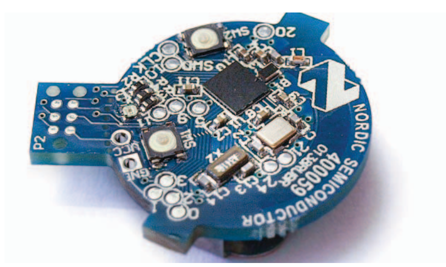

Fig. 5. nRF51822 BLE Smart Beacon

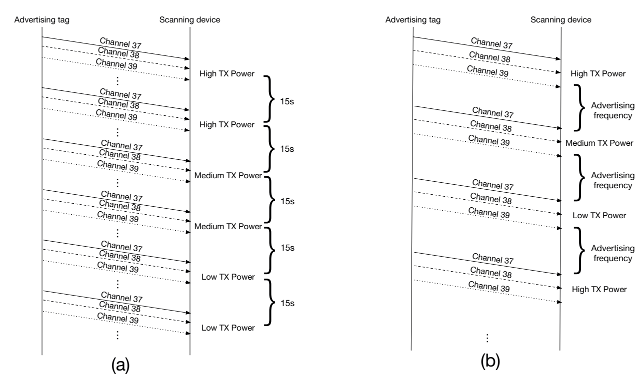


Fig. 7. Transmit power level control: (a) tag switches transmit power levels every 15 seconds; and (b) tag switches transmit power level every advertising period.

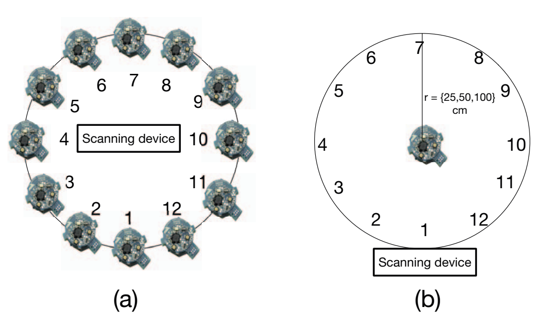


Fig. 6. Circular Setups

strong enough to appear closer than the RSSI values for any other position at any other distance tested. At close distances, this implies that tag orientation could have the largest impact on successful nearness ordering. Thus, it may be critical to develop methods to attach tags to products in such a way that when they are shelved, the tags are oriented to prevent the possibility that aberrant readings due to antenna orientation create false positive results.

*2) Linear Sensing:* In our next set of experiments, we arranged seven tags linearly along a wall, all oriented in the same and placed the scanning device in front of the middle tag (*see* Figure 11). The distance between the scanning device and the middle tag was 25 cm in the first set of runs and 50 cm for the second set. We varied the spacing between tags on the wall as follows: 10 cm, 25 cm, 50 cm, and 100 cm. Finally, we varied the transmit power levels between the three possible settings. The advertising period was set to 100 ms for these experiments.

The goal with this set of experiments was, while controlling for the propagation effects explored with the previous two

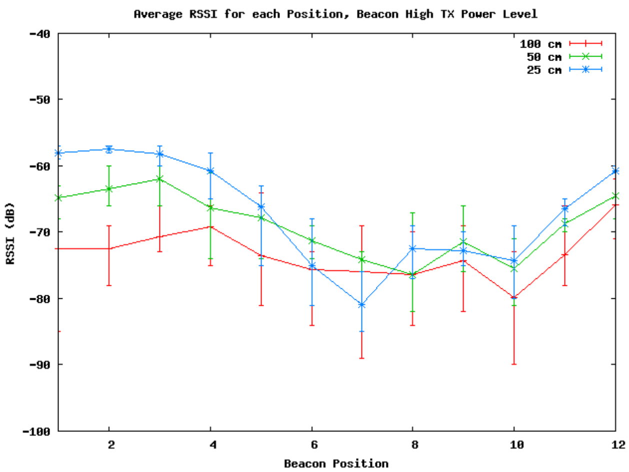


Fig. 8. Setup a - high power

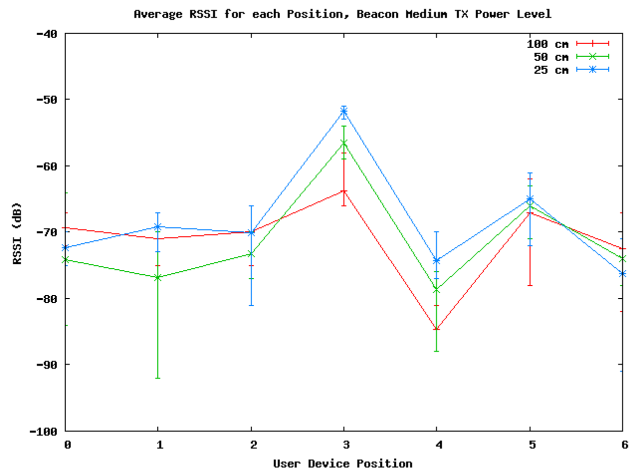


Fig. 10. Setup b - medium power

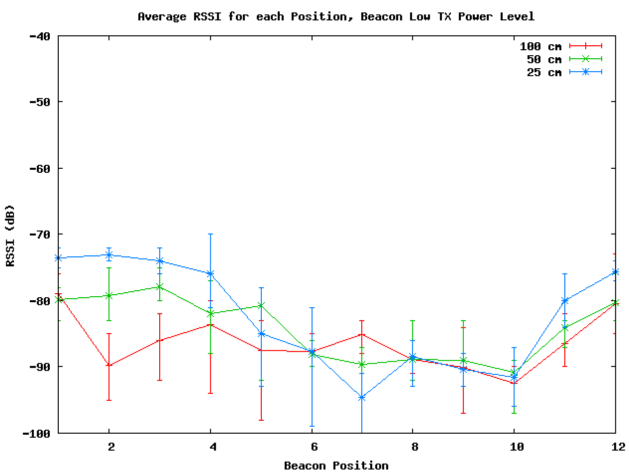


Fig. 9. Setup a - low power

Fig. 11. Linear objects (Fig. 6(c) *supra*)

experimental setups, to explore the variation of instantaneous RSSI readings in each of the scenarios described above. Essentially, since long-term averages of RSSI values will not produce accurate nearness orderings, we explore the limits of accuracy on instantaneous RSSI values.

The first set of results are from experiments where the scanning device is located 50 cm from the center tag (tag number 4 in the figures). When the distance between between the tags was greater than 25 cm, the RSSI value from the closest tag was always higher than the remaining tags; however, as the inter-tag distance shrinks to 25 cm or below, distinguishing between the nearest tag and those directly next to based on instantaneous RSSI values becomes unreliable (*see* Figure 12).

If we reduce the distance between the nearest tag and the scanning device to 25 cm, then the inter-tag distance where instantaneous RSSI values becomes unreliable is 10 cm, which is again roughly $1/2$ the distance between the scanning device and the nearest tag (*see* Figure 13). In fact, this trend continued throughout our experiments. Thus, if we are to use instantaneous RSSI values, our smart labels must be placed less than twice the inter-tag distance apart from the expected location of the nearest product. In a retail environment, the distance between tags, however, is likely to be based on the product size.

### C. Transmit Power Control

Another factor to consider is transmit power control. While energy conservation plays a key role in IoT systems such as our Smart LaBLE system (consider attempting to change batteries on tags embedded in thousands of products), it is also clear that tags must utilize enough energy to both send advertising messages frequently enough to accurately reflect the inventory situation and transmit those messages with enough energy to be successfully received by the scanning device.

The well-known inverse-square law relates distance to signal strength.

$$I = \frac{P}{4\pi r^2}, \tag{1}$$

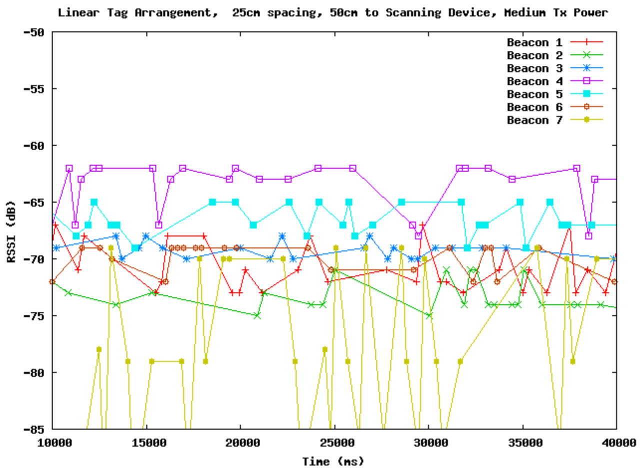


Fig. 12. Linear tags: 50 cm distance to scanning device, 25 cm gap between tags

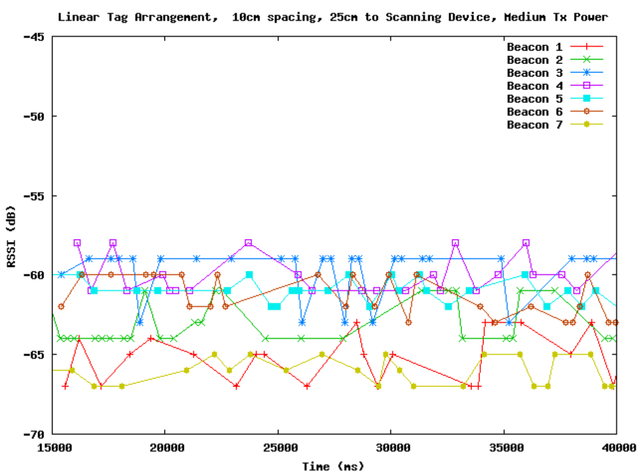


Fig. 13. Linear tags: 25 cm distance to scanning device, 10 cm gap between tags

where $I$ is the power per unit area at distance $r$ a transmitter emitting a signal with power $P$. However, distance is not the only reason signals attenuate and often times effects such as multipath fading outweigh the effects of distance. Additionally, variations in equipment can also alter the perceived energy at the receiver. As we demonstrated in Section III-B, antenna do not have perfectly circular propagation patterns and scanning devices generate their own electromagnetic interference that unevenly affects reception.

However, the attenuation due only to distance affects signals transmitted at different power levels by the same magnitude. In other words, if a signal $S(x)_1$ is transmitted at power $P_1$ and a signal $S(x)_2$ is transmitted at power $P_2$, both to a receiver at distance $r$ from the transmitter, then the difference in transmit powers $P_1 - P_2$ is equal to the difference in perceived signal strengths at the receiver, $I_1 - I_2$. This is demonstrable via a simple application of Equation 1.

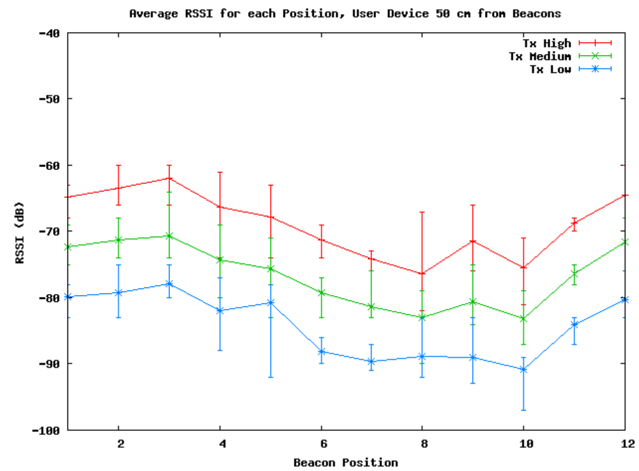


Fig. 14. Experimental Setup (a): 50 cm radius

Looking at the data from our experiments in a slightly different way, this trend becomes apparent. For example, Figure 14 depicts the signal strengths from the objects in the environment circled around the scanning device at a 50 cm distance (from the experiments as depicted in Figure 6(a)). Each line represents a different transmit power level. As is evident, the lines are nearly 8 dBm apart, which is the distance between transmit power levels. However, the lines are not *exactly* 8 dBm apart. This variation away from 8 dBm is due to non-distance-related attenuation.

As for the actual energy consumption for the tags, this is of course device dependent. The Nordic tags in our experiments have two internal voltage regulators that supply 1.7 V to the analog components of the device and 1.2 V to the digital components. Essentially, the entire device can be run off a 1.8 V source (if a larger source is used, the regulators simply burn off the excess voltage). During transmission, the tags use an average of 5.5 mA to 16 mA depending on the transmit power control settings (4 dBm to -30 dBm respectively).

Assuming that advertising packets attached to our products carry a full 31 B payload (recall Figure 4), each packet is 328 $\mu$s long. Thus the energy consumed to transmit a single advertising packet varies between 3,247 mJ to 3,542 mJ.

## *D. Channel Contention*

If instantaneous RSSI values are to be of use, messages from all tags of interest must be received within a small window of time, as was discussed in Section III-B. However, as a space becomes densely populated with tags, the probability of losing advertising messages due to collisions increases.

If we assume that transmissions are independent events (probability of a transmission occurring does not affect the occurrence of another transmission) and each advertising message is 80 $\mu$s long (the size we use in our experiments), then the probability of a collision is given by the following equation.

$$P_c = 1 - [1 - \frac{2 * (l_p - 1)}{l_p + f_s}]^{(N-1)}, \quad (2)$$

where $f_s$ is the size of the advertising period in $\mu$s, $l_p$ is the length of a packet in $\mu$s, and $N$ is the number of tags within range of the scanning device. Clearly, $f_s$ must be greater than $N * l_p$ or we have guaranteed collisions at all times.

If we assume 328 $\mu$s packet lengths, which is the length for an passive scanning mode advertising message with a full payload, then the collision probabilities can be calculated using Equation 2. Figure 15 depicts the number of tags in an environment it takes to raise the collision probability to at least 1%, 5%, and 10% for various lengths of advertising period. If we want to receive advertising messages within a one second window (so that the RSSI values can be correlated for the purposes of determining a nearness ordering), it is immediately apparent that the window size must be chosen carefully based on the number of tags. Furthermore, even with a 750 ms window, the odds of getting messages from over 100 tags is not very good.

This further motivates the need for our Smart LaBLEs to work as decentralize hubs, causing tags for which a particular Smart LaBLE is responsible to transition to a much longer advertising period, given that we expect hundreds of products to be packed within a single retail aisle.

Given this analysis of the BLE channel, we next present the design and implementation of Smart LaBLE, our smart inventory system that utilizes an nearness algorithm developed based on the insights gained in the analysis presented in this section.

## IV. The Smart LaBLE System

Smart LaBLEs automatically configure their associated displays to show product information for the product that is shelved nearest to them. This allows products to be moved on shelves without the need to manually update any signage. Additionally, such automatic configuration allows the labels to display dynamic information such as the number of products of a certain type remaining on a shelf. As we demonstrated with our extensive analysis in Section III, so long as the distance between the Smart LaBLE and the nearest product is less than twice the distance between products of different types, the comparison of instantaneous RSSI values, *i.e.*, values received within approximately a one second window, are sufficient to accurately determine the nearness product. However, that assumes that the tags on the products are uniformly positioned. This is clearly not likely in a retail environment. Thus our algorithm averages RSSI values over a short window to smooth out anomalous positioning of the tags on the product shelf. The system then uses these average values to determine the nearest product. In the next section, we present experiments that compare different values of this window to determine the optimal window.

Based on our in-depth analysis of tags, the BLE channel, and proximity estimation using RSSI values, we designed and implemented our Smart LaBLE system for automating inventory control. For our product tags, we utilized the Nordic Tags used in the analysis presented in Section III. For our initial tests, we placed tags at the bottom, center of each product (*see* Figure 16).

We implemented the Smart LaBLEs using Nordic Semiconductor BLE Dongles and Arduino Uno devices with attached color LCD displays (*see* Figure 17). Each Smart LaBLE is attached to a central laptop for data collection to generate the results presented in this section. Essentially, the BLE dongle attached to each Smart LaBLE functions as a passive scanning device, listening for advertising messages from any products in its reception area.

Each product has a tag with a MAC Address that we utilize to encode the product identification. Recall the MAC address is 6 bytes (see Figure 4). We utilize the first 4 bytes of the address to encode the product identification (in our tests, the flavor of Gatorade$^{TM}$) and the last two bytes to encode a unique identifier for the product of that type. The unique identifier allows the system to track which specific products have been sold, for example to determine if the most recently shelved products were sold first. Such information could be useful in optimizing product shelving procedure or to analyze the habits of customers. For example, it could be used to answer the question of whether the customers typically skip the front-most product on the shelf or not. The 31 byte payload is reserved for other product information, such as cost, description, and so on. Essentially any other information that should be displayed on the Smart LaBLE can be encoded within the payload.

Each Smart LaBLE listens for all tags within its reception range and records the product type, unique id, and RSSI for the advertising messages. The Smart LaBLE also records a time stamp representative of the message received time. Our analysis in Section III-B showed that we need to compare advertising messages received within a short time of each other (typically less than a second). As we saw, averages over the long term fail to produce accurate nearness orderings. Thus the Smart LaBLE system does *not* require time synchronization between the tags themselves. Only the time window around which messages are received impacts the accuracy of the system.

The Smart LaBLEs in the test system are attached to color LCDs. Once a Smart LaBLE determines the product for which it should display information (in our tests, these are different flavors (and thus colors) of Gatorade$^{TM}$), the Smart LaBLE changes its display color to match that of the product and displays a product identification and the number of products of the same type that are on the shelf. Thus, each label will display information for the product nearest the label itself. If different products are mixed within a column behind each label, the front-most product information will be displayed until that product is purchased. However, the system maintains information about the total number of products of each type

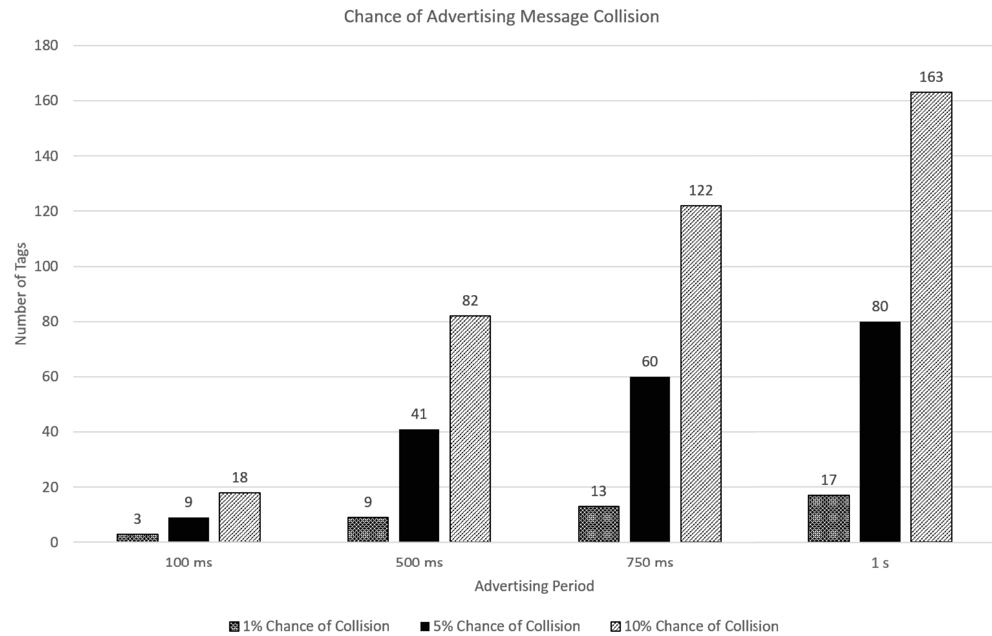


Fig. 15. Chance of Collision, Passive Scanning Mode Advertising Packet with Full 31 Byte Payload

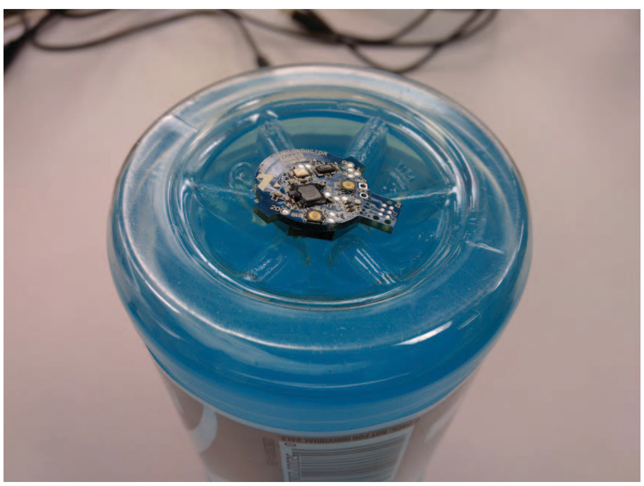

Fig. 16. Tagged Product

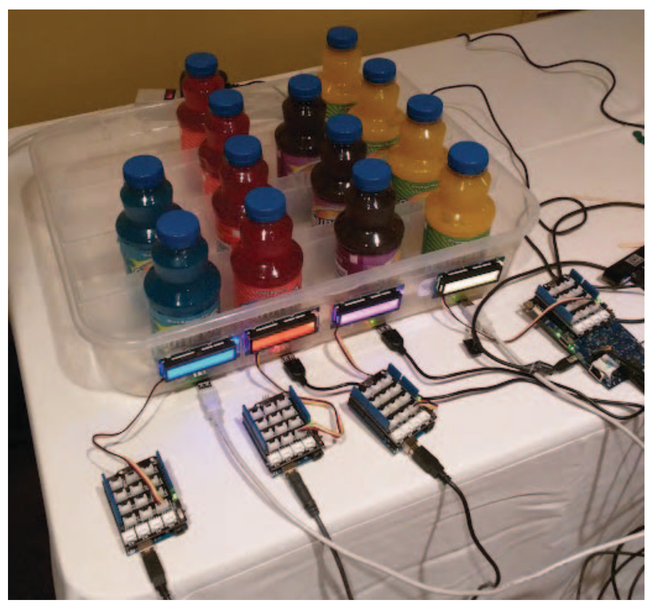

Fig. 17. Smart LaBLE System

near each label[3].

One final design decision has to do with the choice of an advertising period. As we saw in previous sections, if the advertising window is very short, channel contention can cause the loss of advertising messages, actually *increasing* the amount of time it takes to successfully receive advertising messages from each of the products on the shelf. Additionally, sending more frequent advertising messages consumes more energy and will drain the battery of the tags more rapidly: batteries that are essentially impossible to change. However, over-extending the advertising period will make the nearness ordering inaccurate. As our analysis shows, RSSI comparisons can only produce sufficient nearness orderings if the advertising messages from all the tags are received within a reasonably short window. Thus if the advertising period is long, the accuracy of the nearness ordering will suffer, potentially causing the Smart LaBLEs to display the wrong

[3]Given that the Smart LaBLE system has access to the total number of products near a label, other metrics could be used to decide what information to display. For example, a Smart LaBLE could display product information related to the product nearest the label in greatest numbers.

product information.

To solve this problem, a Smart LaBLE, once it has automatically configured itself, signals all tags, causing them to increase the advertising message period, reducing the frequency of advertising messages received from products the Smart LaBLE has already seen. The Nordic Tags have the ability to have their modes changed via over-the-air signals, easily facilitating this function. For our experiments in the next section, during Smart LaBLE auto-configuration, the tags on each product are set to have an advertising period of 100 ms. One possible simplification could be achieved by having a longer advertising period even during the autoconfiguration stage. Essentially, given that products are frequently stocked during times when stores are not busy, it is possible that taking longer to configure the Smart LaBLEs would not be a problem. In this case, there would be no need to utilize BLE radios capable of over-the-air configuration. This could have the benefit of making the tags cheaper to manufacture as well as making them more energy efficient.

## V. Evaluating the Smart LaBLE System

In this section, we present results from experiments run with our Smart LaBLE testbed. The testbed consists of four Smart LaBLEs. We instrumented four different types of Gatorade™bottles with Nordic tags set with an advertising period of 100 ms.

We ran a number of experiments with different bottle configurations. For each experiment, we collected data from each of the four Smart LaBLEs for five minutes. The first batch of experiments tested the sensitivity to changes in product placement. First, to accurately choose the nearest product, our Smart LaBLE must wait long enough to have high confidence it has heard advertising messages from all tags. Additionally, we would like to hear at least one advertising message from each of the tags within a small window. For each run, the bottles were shuffled so that the products were in different locations for successive runs. We tested two general types of arrangements: arbitrary shuffles, where different types of bottles could be located in the same columns; and column-organized shuffles, where product types were maintained in columns, but the columns were shuffled in each run. Since the results for each type were similar, we present results for the arbitrary shuffles throughout the rest of the paper.

We ran a number of experiments with 8 and 12 products to determine the impact of the number of products on the total time it takes to detect at least one advertising message from each product. During the experiments, each of the Smart LaBLEs kept track of the first time an advertising message from a particular product was received. Figure 18 presents the results with 8 products and Figure 19 presents the results with 12 products.

What is immediately apparent is that there are a couple of outliers in each case that raise the maximum time until all products are seen. For example, within half a second, nearly 3/4 of the products are detected when there are 8 products. It takes nearly double that amount of time to find 3/4 of

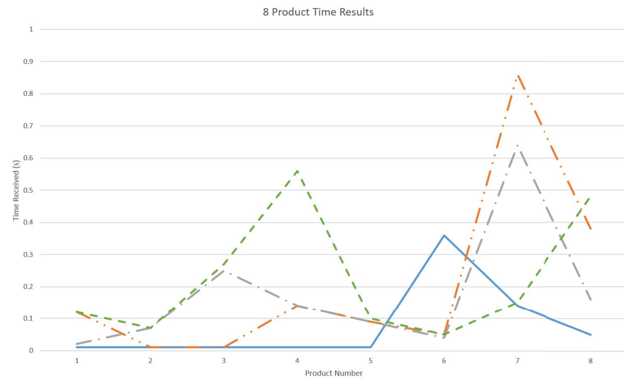


Fig. 18. Time Results: 8 Products

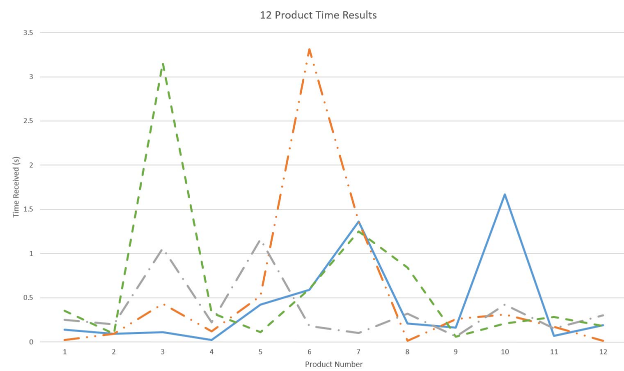


Fig. 19. Time Results: 12 Products

the products when there are 12. Finally, the time to hear advertising messages from all products is nearly tripled (from around 1 sec to just over 3 seconds).

From the results in Section III-D, we know we can expect the time to receive advertising messages from all products to increase as the number of products increases. Furthermore, we know from the results in Section III-B that to accurately choose the nearest product, we must compare RSSI values received within a short window. Therefore, we use a combined approach where we initialize values using instantaneous RSSI values and correct those values with averages over 500 ms windows. Essentially, our Smart LaBLEs insert each product seen into an array along with an RSSI value. Then, if another advertising message is seen from a product already in the array within the next 500 ms, the new RSSI value is used to create a moving average. Such averages allow fluctuations in the RSSI values to be smoothed without causing false positive results.

Figures 20 and 21 present results from one of the runs of our experiments using all four Smart LaBLEs in the setup seen in Figure 17. As is clear, one product is determined to be the nearest for each of the Smart LaBLEs, in fact, a clear nearness ordering has been derived. We verified each run as compared against ground truth determined manually. Over multiple runs, our Smart LaBLEs only had an error rate of 1% (we had 1 error in 100 test cases).

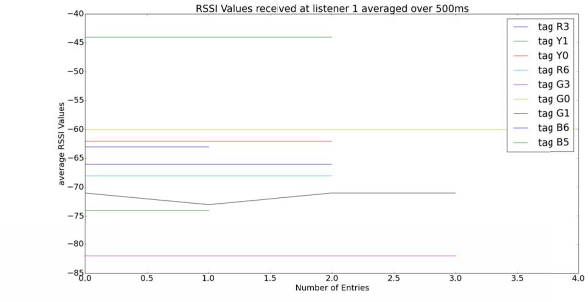


Fig. 20. Results from Smart LaBLE One

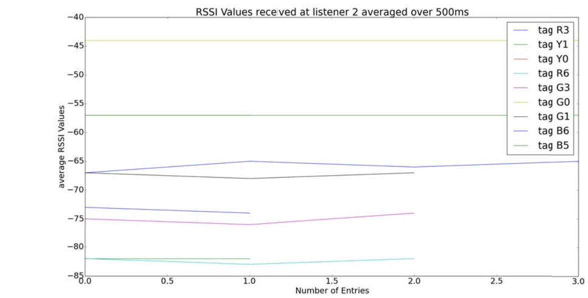


Fig. 21. Results from Smart LaBLE Two

## VI. Conclusions and Future Directions

In this paper, we have presented an in-depth analysis of the BLE channel characteristics. Since BLE devices are the foundation for recent IoT applications, a complete understanding of such characteristics can lead to better designs of IoT hubs and protocols at the edges of the network. Based on this analysis, we have presented the design and implementation of our Smart LaBLE system for automating an aspect of inventory control. We have shown that our algorithm has a very low false positive rate (1%). Furthermore, our system can reduce the advertising period for tags on products that have already been added to the system, further conserving both energy and bandwidth resources.

Our Smart LaBLE prototypes provide a nice base system that can support further research. First, we intend to further explore the use of dynamic transmit power control to further optimize the Smart LaBLE system. As is clear from our channel analysis, it should be possible to leverage variations in RSSI from back-to-back advertising messages transmitted at different transmit power levels to increase the distance between the scanning device and the nearest tag without also increasing the distance between tags. Currently however, the Nordic tags can only change transmit power levels between advertising periods. Such work will allow us to more carefully analyze system energy efficiency. Additionally, power control could be leveraged to increase scalability by limiting the number of products any particular Smart LaBLE can hear. This is particularly important considering the fact that a single retail isle may shelve hundreds or thousands of products.

We would also like to explore dynamically changing the advertising periods as a response to advertising message loss. If the system begins to perform poorly due to lost advertising messages, it should be possible, based on our analysis, to increase the advertising period of the tags in the environment. Attempting to dynamically adjust these periods is a future direction for our Smart LaBLEs.

Additionally, we would like to explore the impacts of placing tags in different modes once they have been identified and claimed by a particular Smart LaBLE. We have shown in this work that we can greatly increase the available bandwidth (and thus allow the environment to support a larger number of products), however, there are trade-offs to be considered. For example, if the tag permanently went to sleep, then the Smart LaBLE would not be able to tell when the product is sold. Thus there is a trade-off between system responsiveness and resource conservation. Similarly, when reshelving occurs, the Smart LaBLE needs to recognize that it is no longer responsible for the old products.

## VII. Acknowledgments

This research is funded in part by a Focused Research Grant from Google, Inc.

## References

[1] A. Kwiecień, M. Maćkowski, M. Kojder, and M. Manczyk, "Reliability of bluetooth smart technology for indoor localization system," in *Computer Networks*. Springer, 2015, pp. 444–454.

[2] R. Faragher, "An analysis of the accuracy of bluetooth low energy for indoor positioning applications."

[3] P. Martin, B.-J. Ho, N. Grupen, S. Muñoz, and M. Srivastava, "An ibeacon primer for indoor localization: Demo abstract," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, ser. BuildSys '14. New York, NY, USA: ACM, 2014, pp. 190–191. [Online]. Available: http://doi.acm.org/10.1145/2674061.2675028

[4] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Commun. ACM*, vol. 21, no. 7, pp. 558–565, Jul. 1978. [Online]. Available: http://doi.acm.org/10.1145/359545.359563

[5] M. Siekkinen, M. Hiienkari, J. K. Nurminen, and J. Nieminen, "How low energy is bluetooth low energy? comparative measurements with zigbee/802.15. 4," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE*. IEEE, 2012, pp. 232–237.

[6] Bluetooth smart technology: Powering the internet of things http://www.bluetooth.com/pages/bluetooth-smart.aspx.

[7] J. Müller, J. Exeler, M. Buzeck, and A. Krüger, *ReflectiveSigns: Digital Signs That Adapt to Audience Attention*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 17–24. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01516-8_3

[8] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: unsupervised indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 197–210.

[9] I. Constandache, R. R. Choudhury, and I. Rhee, "Towards mobile phone localization without war-driving," in *Infocom, 2010 proceedings ieee*. IEEE, 2010, pp. 1–9.

[10] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *Proceedings of the 18th annual international conference on Mobile computing and networking*. ACM, 2012, pp. 293–304.

[11] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2. Ieee, 2000, pp. 775–784.

[12] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, “Indoor localization without the pain,” in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*. ACM, 2010, pp. 173–184.

[13] M. Blackstock and R. Lea, “Iot interoperability: A hub-based approach,” in *Internet of Things (IOT), 2014 International Conference on the*. IEEE, 2014, pp. 79–84.

[14] R. Lea and M. Blackstock, “City hub: A cloud-based iot platform for smart cities,” in *Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on*. IEEE, 2014, pp. 799–804.

[15] P. P. Ray, “Home health hub internet of things (h 3 iot): An architectural framework for monitoring health of elderly people,” in *Science Engineering and Management Research (ICSEMR), 2014 International Conference on*. IEEE, 2014, pp. 1–3.

[16] K. Varshavskiy, B. Ehrat, V. Khanna, A. F. Harris III, and R. Kravets, “Working out with iot: Pairing and automated data collection in a gym environment. (demo),” in *Mobile Computing Systems and Applications (HotMobile), 2016 International Workshop on*, 2016.

[17] Intel iot gateway. [Online]. Available: https://www-ssl.intel.com/content/www/us/en/embedded/solutions/iot-gateway/overview.html

[18] Vmware iot gateway. [Online]. Available: https://blogs.vmware.com/cto/vmware-and-the-internet-of-things-motivating-the-three-tier-architecture/

[19] J. Liu, C. Chen, and Y. Ma, “Modeling and performance analysis of device discovery in bluetooth low energy networks,” in *Global Communications Conference (GLOBECOM), 2012 IEEE*, Dec 2012, pp. 1538–1543.

[20] N. Langhammer and R. Kays, “Enhanced frequency hopping for reliable interconnection of low power smart home devices,” in *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC)*, Aug 2012, pp. 305–310.

[21] M. Siekkinen, M. Hiienkari, J. K. Nurminen, and J. Nieminen, “How low energy is bluetooth low energy? comparative measurements with zigbee/802.15.4,” in *Wireless Communications and Networking Conference Workshops (WCNCW), 2012 IEEE*, April 2012, pp. 232–237.

[22] S. Silva, S. Soares, T. Fernandes, A. Valente, and A. Moreira, “Coexistence and interference tests on a bluetooth low energy front-end,” in *Science and Information Conference (SAI), 2014*, Aug 2014, pp. 1014–1018.

[23] M. Maman, F. Dehmas, R. D’Errico, and L. Ouvry, “Evaluating a tdma mac for body area networks using a space-time dependent channel model,” in *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, Sept 2009, pp. 2101–2105.

[24] E. B. Hamida, R. D’Errico, and B. Denis, “Topology dynamics and network architecture performance in wireless body sensor networks,” in *New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on*, Feb 2011, pp. 1–6.

[25] J. r. Lin, T. Talty, and O. K. Tonguz, “On the potential of bluetooth low energy technology for vehicular applications,” *IEEE Communications Magazine*, vol. 53, no. 1, pp. 267–275, January 2015.

[26] nrf51822 bluetooth smart beacon kit. http://bit.ly/1L34QCz.