DeepBLE - Indoor Navigation using Bluetooth LE

Dept. of CIS - Senior Design 2013-2014*

Eric Kim
erkim@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

Aditya Sood
asood@seas.upenn.edu
Univ. of Pennsylvania
Philadelphia, PA

ABSTRACT

Currently, mobile phones are the primary method of navigation. However, indoor positioning and navigation poses a unique problem because the GPS satellites normally used to navigate outdoors have limited use indoors. One solution is using Wifi access points as anchors, measuring the signal strength and calculating position using trilateration. Indeed there are companies today that provide the framework for such an implementation. However there are certain disadvantages with using Wifi, namely that there is no context to the position. A location must be enriched with a predefined context separately.

This application aims to address this issue by implementing a different method of indoor positioning, one that uses the recently released Bluetooth 4.0. With the new APIs released by Google we can now use Bluetooth devices to act as anchors. The key feature of Bluetooth 4.0 is the communicationbetween devices that is allowed; this has allowed us to enrich and contextualize position. This peer to peer messaging opens up many possibilities, ranging from applications in shopping malls to emergency response situations.

1. BACKGROUND

With the widespread availability of the smart phone, individual navigation has been refined such that a user can navigate to and from a particular address. The standard used today for outdoor navigation relies on GPS satellites to track the device location. GPS is generally not well suited for indoor use for two reasons - 1. GPS does not provide a high level of accuracy, and 2. the GPS signal breaks down indoors. So rather than using GPS satellites, indoor navigation and positioning has been accomplished largely through using networks of nearby "anchors" that have a static, known position. Most commonly used anchors today are Wifi networks. The device detects a Wifi network with a unique ID; with the wifi access point, we can triangulate the exact position of the device. Indeed there are several existing companies that will set up the necessary pieces to allow for step by step navigation through a shopping mall or a departmental store.[1]

Google and Apple have both introduced a technology called Low Energy Bluetooth, also known as BLE or Bluetooth Smart, that introduces a new way to navigate indoors. Apple in their recent release of iOS7 has included an API called iBeacon, that will use BLE extensively for the purpose of precise geolocation[1]. Any "beacon" that is set up will be available for general iPhone users to navigate with; what makes this technology remarkable is that BLE uses very little energy, as the name suggests, has considerable range, especially compared to Wifi networks, and most importantly, beacons can be setup anywhere. Any iPhone device can be set as a beacon, and devices designed specifically for the use of becaons can be purchased. Similarily, Android in their most recent OS release 4.3 has implemented BLE as well, providing a well defined API[5] to develop upon. As of the writing of this paper, only the most recent devices even have BLE hardware built in, and on top of that only select devices have the OS that provides a native API to utilize BLE, so suffice it to say BLE as a method of navigation is still in its early stages.

A method of context free positioning has many important use cases. One example is in emergency response situations, where location awareness is of utmost importance. "Existing indoor navigation solutions usually rely on pre-installed sensor networks, whereas emergency agents are interested in fully auto-deployable systems"[9]. Indeed the current Wifi implementation requires Wifi access points, a data service that computes location, and a location specific context in order to work correctly. Although this might be feasible for shopping malls and departmental stores, it does not have much use in a situation where there is no predefined context. We may be able to use pre existing Wifi networks, but that is too much of an assumption; especially in the emergency response situation, Wifi access points may be unreliable, inconvenient, and not suited for the situation. With BLE, all we need to do is drop a few anchors to detect devices and have the devices transmit small pieces of data, such as an ID, and we can successfully track the location.

Our app seeks to take advantage of bluetooth's low cost, easy setup, and rising ubiquity as leverage against the Wifi based positioning systems in place today.

2. INTRODUCTION

Existing indoor navigation systems have the problem of high setup costs and a lack of contextual awareness. Our application addresses this issue by using the aforementioned BLE technology. "The GATT profile is a general specification for sending and receiving short pieces of data known as 'attributes' over an LE link. All current Low Energy application profiles are based on GATT" [**?**]. Using the GATT profile, contextual awareness is now possible to add anywhere. Important to note here is that all current Low Energy application profiles are based on GATT - this means

*Advisor: Boon Thau Loo (boonloo@cis.upenn.edu)

[1] See meridianapps.com and senionlab.com

any device can act as an anchor, receiving and sending pieces of data, eliminating the lengthy and costly setup of wifi access points and a framework to manage them. This also means that there is no need for getting software out to the users in order for the application to work; since the GATT profile gives us the ability to distinguish between bluetooth devices, we can identify which devices we care about, which, in this case, are smartphone devices.

Currently, our application allows for detection of any bluetooth device within the phone's bluetooth range. It will succesfully receive the attributes (the pieces of information that make up a profile) of each device. Attributes can be anything from user id to geolocation. In the next months, we will implement our anchors, devices with a static known location. Since communication between devices has been established, this is the logical next step. Calculation of location will then be implemented, followed by the "virtual map" rendering, which will be explained in the sections to follow. Finally user friendliness and ease of use is one of the main goals; setup should be as automated as possible, so we will be keeping that in mind during our implementation. The rest of this paper will detail out the system specifications, and what possible challenges may come up.

3. RELATED WORK

Fully functional Indoor navigation apps, although a relatively recent invention, have been implemented before. For example, the company SenionLab provides a way for third parties to integrate an indoor navigation API to their existing application[8][2]. The API includes location based advertising, allowing for companies to send tailored advertisements to the customers that walk by their store, location analytics, the ability to gather data on user behavior and most importantly, a fully functional step by step navigation system at the granular level. However, there are some shortcomings that current indoor navigation apps have in common. Most widespread is the dependence on existing Wifi access points as the anchors. Although this allows for the application to pinpoint the exact location of the device and track it as it moves, it does not provide environmental awareness. These Wifi based implementations do not carry any data about a particular location, such as whether a store carries a particular product, or if the store is currently having a sale. A fundamental assumption is being made in this case, that is, users already know *where they want to go*, rather than *what they want to do*, and this is the fundamental issue we seek to resolve.

We also cannot forget the costs of implementing a framework that uses wifi access points. Wifi needs to be set up well in advanced, and a data service that calculates position must be implemented. Finally, Wifi has associated costs that the user must invest in, such as monthly internet charges, maintenance, and an app for users to download so that all of the positioning makes sense.

With BLE we no longer have to make this assumption. BLE signal transmitters are low cost: for example, for Apple's iBeacon, "beacons" as they're called, cost as little as 30 or 40 dollars. As their name suggests, BLE uses little energy. Most important of all, they can be ubiquitous - a shopping mall where every store has a proximity sensor, or an anchor, and we can achieve much more granularity than Wifi access points could ever provide. With this granularity comes enriched data - anchors no longer just provide a specific location, they can provide specialized promotions, specific directions, and most important for our purposes, contextualized notifications[6]. This means that a user, can navigate through a room and find information about their specific location, such as other users in the room or immediate area

Indoor navigation using older Bluetooth technology has also been implemented. "The system compares the signal strengths of surrounding Bluetooth devices to a database of measurements taken across the indoor area, in order to estimate the user's position. Through an evaluation of the system, an accuracy of approximately 1.5 meters has been obtained."[2]. Even using the older bluetooth technology, an accuracy of 1.5 meters is possible; we will use this as the goal for our implementation.

Lastly, as of the writing of this report, there are now companies that implement a Bluetooth based positioning system[7]. Indoo.rs for example, offers a package of bluetooth anchors, along with an API for navigation. Their system involves integrating their SDK with existing applications. Features include navigation (not specifically for indoor use, but optimized for indoors), routing, and analytics that track user movements. The main difference between our application and the SDK created by Indoo.rs is that our application will be completely stand alone, optimized for the emergency response use case. Our biggest motivation is to remove the need for setting up complex systems in order to start navigation.

4. SYSTEM MODEL

There are two needs that the application aims to address. First, the existing indoor navigation apps do not necessarily take full advantage of the existing technology. Contextual awareness is a common weakness that these apps suffer from, relying mostly on wifi access points to pinpoint position, and all other data is built upon this position. The app must know what object is currently at a particular coordinate in order to infer any additional information about the location. With BLE technology, we can enrich location with the data carried by the "anchors" that indoor navigation implementations achieve with a predefined context.

Second, due to the requirement of a predefined context, existing indoor navigation systems must be set up very carefully. To the Wifi access points that act as anchors, a set of coordinates means nothing. The people implementing the navigation must associate specific coordinates with bits of information. There is a manual process of distinguishing important locations; with our application the anchors and devices themselves can hold that information. Our implementation allows for a device to broadcast information such as userid, name, geolocation to all other devices and anchors. Clearly this allows for a more dynamic and fluid navigation system, where the transfer of all this information happens in real time.

Using BLE technology to gather richer data about geolocation, our app that allows a user to set a few BLE devices as anchors and then track location of other devices using the app, while also gathering some information that pertains to each of device (such as user id).

[2] SenionLab: The website contains a splendid video demonstrating the capabilities of their turn by turn navigation http://senionlab.com

This will be accomplished by using set devices as anchors, which will hold static location. This location will be calculated through RSSI signal strength, and direction given by the user. The devices with fixed location will send their coordinates to the device requesting its position. The device will then render a virtual map of its surroundings, based on the coordinates given by the anchors. It will also calculate its own position in real time as the device moves. Together, the user will be able to track the movements of themselves, as well as the movements of their friends in the room who are also using the application (although the device is capable of detecting all bluetooth devices, regardless of whether they are using the application).

5. SYSTEM IMPLEMENTATION

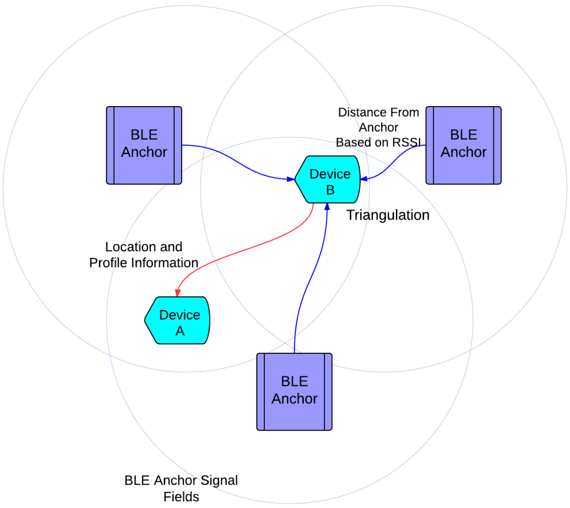

Figure 1: The Approach to BLE Navigation

Three main modules must be put into place. First, the central data processing unit will store all the relevant data pertaining to the anchors. For example, in the shopping mall implementation, the product catalog and other information about the store would be associated with a partcular anchor, and then stored in the central data unit. The central data service will then communicate with device, alerting it for example when an item in the shopping list is nearby. It will also store the exact location of each anchor, which it will use to calculate the position of a device when it comes into range of any anchor. The ideal structure that handles the central data service would be one that can parse different pieces of data.

Finally, the device to device communication must be well defined. When a device enters the range of an anchor, the anchor should send over to the central data service information about itself (for example, what products they currently have in stock) as well as the location of the nearby device. Communication between devices is also key; each device must be able to transmit profile information to another device reliably.

Calculating position using Bluetooth also poses an interesting problem. ”For a precise position estimation, the dependence between the distance and the received signal strength has to be determined. Especially in the indoor area, boundary conditions like reflections and wall damping make the use of the equation for the free-field propagation impossible. Therefore, the required calculations of the distances are estimated by an approximation of the Received signal Strength Indicator (RSSI)” [4]. We must use a method to calculate position that has been traditionally applied to older Bluetooth technology to the brand new BLE 4.0.

In order to accomplish this implementation, we will need to extensively test the capabilites of BLE. We need to be familiar with the range and limitations of the hardware; the range and strength of the bluetooth signal must be tested. Empirical tests on older versions of Bluetooth do exist, and we will use this as a benchmark and starting point for our research and testing[3]. Specifically, we must test Bluetooth's viability in the indoor setting; walls, stairs, and other obstacles, all pose potential problems, since we must successfully navigate a user relying on mostly signal strength. If signal strength is greatly hindered by a wall in the room, we must be able to correctly adjust. This may mean additional anchors will be required.

Here lies a big strength in using Bluetooth as the underlying hardware, instead of Wifi access points. Whenever a wall or an obstacle prevents better position information, and there is no way for the existing anchor points to determine location accurately, then the simple solution is to add more anchors. Since anchors (bluetooth devices) are relatively inexpensive, especially when considering how easy they are to place and set up, we can always add more anchors to increase the accuracy of the calculations.

The method we will use to caclulate location will be based of the Received Signal Strength Indicator (RSSI) that is associated with Bluetooth signal transmission.

One of the main motivations of the problem was relying as little as possible on predefined context; that is, the application shouldn't ask for specific coordinates that need to be calculated with GPS signal. The anchor implementation calls for absolute position, and we must determine this somehow. We meet this challenge by using a ”virtual map” implementation. Instead of forcing users to determine an absolute location for the anchors, we will instead set down an anchor as the origin, and set three other anchors that will provide the absolute location. These three anchors will call for a direction (north, south, east, west) based its location from the anchor designated as the origin. Then, the anchors will be able to calculate their distance from the origin using signal strength.

With distance and direction, we have an exact position in our virtual coorinate system. All of this information will then be transmitted to the device that is being navigated and it will be translated into a coordinate system, with the origin anchor set as the middle of the map.

What this achieves is an abstraction from any and all context. More specifically, we are no longer limited to putting the anchors in spots where GPS or Wifi signal is available.

The use case that our application will demonstrate is day to day use with friends. Those who have the app downloaded will be able to go into a room with a bluetooth device already set as an anchor, and be able to see which friends are in the room, and how to find them. The key features will be implemented through this use case, and the app can then be extended to handle more complex cases, such as shopping

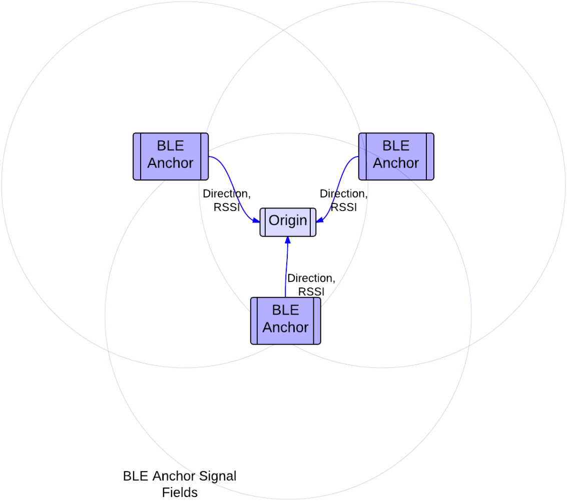

Figure 2: BLE Nav Step 1

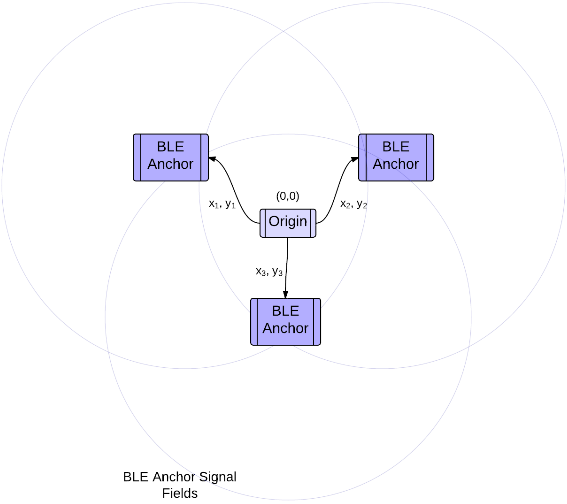

Figure 3: BLE Nav Step 2

mall navigation or emergency response. First, the anchor should be able to detect the device when it comes into proximity; similarily, the device should recognize the anchor in the room. The device will search for all devices in the room; it will need at least three anchors to calculate its own position. The device will also broadcast information about itself (in our case, user profile information), so that other devices in the vicinity can identify the device.

6. SYSTEM PERFORMANCE

A fully functional BLE navigation app is certainly the most basic goal. To demo this, we will be using two devices and an anchor to act as our scenario. The anchor should detect the two devices and successfully send the contextual and RSSI data to the central data service. Then, the data service should calculate the position of all the devices and send back to each of the device its own location, and the location of the other device, along with the user profile associated with the other device. In the end state, the two devices should know where they are, who the other device is, and where the other device is. Note additionally that this should happen in real time; if device B moves, the user of the device A should see the movement.

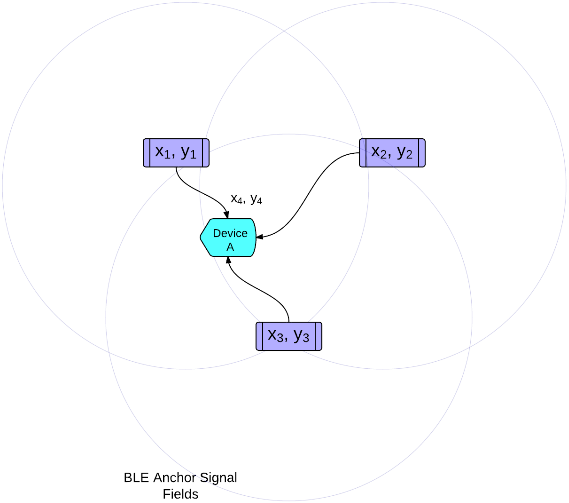

Figure 4: BLE Nav Step 3

Currently, our application is capable of detecting all devices in its range. With a range of about 40 meters (although the website lists the range as 50 meters, the reliability of the measurement starts to break down after 40 meters, and the device presence "flickers" in and out), both the anchors and the phones will cover a small room. It also receives the GATT profile from each of the devices. Although this is a basic test, we expect that the location positioning and anchor recognition will behave normally. Since each device is actively receiving information about other devices in real time, it is easy to see how, once location calculation is implemented, how a device might calculate its own position. Similarly, anchor implementation would be as simple as a boolean attribute in the profile, broadcasting to all phones that the device is an anchor.

The more difficult implementation will be the virtual coordinate system. We expect that most of our time in the next months will be devoted to correctly translating simple distances and directions into fully understandable maps, which then will be rendered on the screen for the user to see. This map will also need to be "scrollable", that is, when you leave the range of one set of anchors, and enter a second set of anchors, where both sets may share any number of anchors. The transition from one "map" made up of certain anchors to another "map" made up of other anchors, should be seamless

One resolution for this may simply be room to room navigation, with each set of anchors being associated with a particular room. The system will then prevent two sets of anchors from sharing data, so that each room is its own unit. Since the coordinate system is entirely virtual, it is susceptible to skewing, and the best way to prevent this may simply be maintaining independent sets of anchors for each room.

As mentioned previously, fully functional bluetooth navigation systems have been implemented and are out for sale currently (the main one being Indoo.rs). Although it is unfortunate that more robust and complete tools are already out in the wild, it does provide an example for us to see what works well and what doesn't.

7. REMAINING WORK

Progress so far makes up about thirty percent of the actual implementation of the project All research concerning triangulation methods, and bluetooth capabilities is finished, and the general framework of the entire project has been set in place.

- ALREADY COMPLETED: Preliminary reading.
- COMPLETED PRIOR-TO THANKSGIVING : Researched Bluetooth hardware, the implementation that will be used to calculate position will be a "virtual map" implementation, explained above. Formulated the framework of the app - the android platform will be used, no other structures are needed. Laid out the mathematics behind calculating the position of an app with triangulation.
- COMPLETED PRIOR-TO CHRISTMAS : The application fully implements BLE communication - Detecting BLE transmitters is the main feature accomplished. Also, the app can distinguish between devices, since it receives the GATT profile from each device.

Moving forward into the spring, the goal will be to implement more of the navigation specific features, such as the rendering of the virtual map, the position calculations, and the real time app use experience.

- ANCHOR RECOGNITION : Implement an anchor "profile", which basically broadcasts to other devices that the location of the anchor will remain static. It can possibly hold other information about its location, for example, what floor of the building the anchor is on.
- LOCATION CALCULATION : The method we will be using is triangulation based on signal strength. We have the math that will calculate location based on three fixed points; the bulk of this task will be correctly calculating location in the presence of walls, obstacles, and anything else that might interfere with the signal strength. We may also need to test how resource intensive the calculations are. This differs from the previous design laid out in the proposal, where a central data service would receive information about all the devices and send back the calculations to each of the devices.
- VIRTUAL MAP : This task involves translating all the locations that have been coordinated and the fixed locations that have been given to the anchors into a coordinate system, which will then be rendered for the user to intuitively use.
- USER INTERFACE: Finally, the user should be able to intuitively set up the anchors, and use the application.
- IF THERE'S TIME... : In situations where there is a large user base, the application needs to be able to calculate and report on the positions of the other users, without it being resource intensive. There are no tests to see how many attributes a Bluetooth device can receive without breaking down. One of the goals is to successfully scale to work with a large amount of users.
- IF THERE'S TIME : Another feature that might be added is the implementation of moving anchors. If we know the position of a device at every clock cycle or signal refresh, we can use that location to calculate the location of other devices. With three devices set as anchors, and another three devices set as users, we can greatly increase the range of position tracking if the user devices can act as temporary anchors for other devices that would otherwise lie outside the range of the anchors.
- IF THERE'S TIME : With the increasing popularity of Bluetooth LE on Android and iOS devices, more and more people are getting access to the technology required to use the app. The original purpose for BLE based geolocation systems was for proximity based awareness. When a phone enters the field of an anchor with a particular set of attributes may be broadcasted. We may like to leverage this feature by allowing user inputted profiles for anchors, so that they can mark the location with a set of details that are particular to the location.

8. REFERENCES

[1] Apple. iOS Developer Library: Core Bluetooth Programming Guide. `https://developer.apple.com/library/ios/documentation/NetworkingInternetWeb/Conceptual/CoreBluetooth_concepts/AboutCoreBluetooth/Introduction.html`.

[2] Anja Bekkelien. Bluetooth indoor positioning. Master's thesis, University of Geneva, 2012.

[3] Kenneth C. Cheung, Stephen S. Intille, and Kent Larson. An inexpensive bluetooth-based indoor positioning hack, 2008.

[4] Silke Feldmann, Kyandoghere Kyamakya, Ana Zapater, and Zighuo Lue. An indoor bluetooth-based positioning system: Concept, implementation and experimental evaluation. In Weihua Zhuang, Chi-Hsiang Yeh, Olaf Droegehorn, C.-T. Toh, and Hamid R. Arabnia, editors, *International Conference on Wireless Networks*, pages 109–113, 2003.

[5] Google. Android Developers: Bluetooth Low Energy. `http://developer.android.com/guide/topics/connectivity/bluetooth-le.html`.

[6] Gottipati Harry. With iBeacon, Apple is going to dumb on NFC and embrace the internet of things. `http://gigaom.com/2013/09/10/with-ibeacon-apple-is-going-to-dump-on-nfc-and-embrace-the-internet-of-things/`, 2013.

[7] Indoo.rs. BLE based indoor positioning. http://indoo.rs/. Indoo.rs offers a BLE based indoor navigation SDK to integrate with existing applications. It includes navigation capabilities, analytics on user movement, and routing tools.

[8] SenionLab, Meridian. Wifi based navigation systems. `http://senionlab.com` and `http://meridianapps.com`. SenionLab and Meridian both offer wifi based indoor navigation APIs to integrate with existing applications.

[9] Valerie Renaudin, Okan Yalak, Philip Tome, and Betrand Merminod. Indoor navigation of emergency agents. *European Journal of Navigation*, 5(3), July 2007.