



Universidade de Brasília

Instituto de Ciências Exatas
Departamento de Ciência da Computação

Avaliação de Técnicas de Localização de Dispositivos BLE para a Physical Web e Prova de Conceito

Guilherme David Branco
Samuel Vinícius Vieira Pala

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Jacir Luiz Bordim

Brasília
2017

Dedicatória

Dedico esse trabalho a Deus e a minha eterna namorada, Isabella.

- Samuel Vinicius Vieira Pala

Dedico este trabalho a minha família, especialmente minha mãe, Maria Luiza, e meu pai, Geraldo, assim como minha namorada, Aline, que me motivaram a continuar a graduação.

- Guilherme David Branco

Agradecimentos

Agradecemos ao professor e orientador, Jacir Luiz Bordim, que proporcionou um ótimo tema tanto no que tange a parte teórica, quanto a parte prática, desta forma nos passando os conhecimentos necessários para o nosso desenvolvimento na área acadêmica. A nossa revisora, Maria Luiza David, devido a sua dedicação e empenho, viabilizando o nosso aprendizado em relação a melhoria da fluidez no texto. Aos nossos amigos, companheiros de curso, Pedro Henrique Lima e Silas Souza Fernandes pela ajuda durante o desenvolvimento do trabalho, seja por motivação ou por sugestões nesta trajetória. E a todos os funcionários do Departamento de Ciência da Computação, que trabalham favorecendo o crescimento e aprendizado dos alunos.

Resumo

A possibilidade de pequenos dispositivos informando um serviço constitui um dos principais aspectos previstos pela *Physical Web* (PW). A PW consiste em uma tecnologia que visa fornecer interações contínuas com objetos físicos e suas respectivas localizações. Os dispositivos da PW utilizam o protocolo *Bluetooth Low Energy* (BLE) para fornecer conteúdo e informações aos usuários em localizações bastante específicas. Contudo, para atingir o seu objetivo, esses dispositivos devem ser capazes de fornecer informações de localização a um nível aceitável. Este estudo investiga a precisão de informações de localização fornecida por um dispositivo BLE. Para isso, implementou-se uma *test-bed* por meio do *Google Eddystone* com *beacons* BLE e avaliou-se sua precisão de localização usando as técnicas de estimativa de localização baseadas no Indicador de Intensidade do Sinal Recebido (do Inglês, *Received Signal Strength Indicator*) (RSSI). Os resultados experimentais mostraram que os *beacons* BLE utilizados foram capazes de fornecer uma estimativa de localização aceitável, fornecendo acurácia média de 0,20m, para distâncias de até 5 metros.

Palavras-chave: *Bluetooth Low Energy*, Posicionamento, Ambientes internos, Filtro Kalman, Estimativa de distância, *Physical Web*, Trilateração, *Beacons*

Abstract

The possibility of having small devices informing the availability of a service is one of the key aspects envisioned by the *Physical Web* (PW). PW is an enabling technology aiming to provide seamless interactions with physical objects and locations. PW devices are based on *Bluetooth Low Energy* (BLE) protocol for delivering content and information to users at very specific locations. To achieve its goal, however, such devices are expected to provide acceptable location information. This paper investigates the location information accuracy provided by a BLE device. For this purpose, we have implemented a test-bed using the Google's Eddystone open format with BLE beacons and evaluated its location accuracy using renowned Indicador de Intensidade do Sinal Recebido (do Inglês, *Received Signal Strength Indicator*) (RSSI) location-estimation techniques. Experimental results shows that the BLE beacons used are capable of providing reasonable location estimation up to 5 meters.

Keywords: Bluetooth Low Energy, Positioning, Indoor Enviroments, Kalman Filter, Distance estimation, Physical Web, Trilateration, Beacons

Sumário

1	Introdução	1
1.1	Problema	4
1.2	Objetivos	4
1.2.1	Objetivo Geral	4
1.2.2	Objetivos Específicos	4
1.3	Metodologia da Pesquisa	5
1.4	Estrutura do Trabalho	5
2	Fundamentação Teórica	7
2.1	Pilha TCP/IP	7
2.1.1	Camada de Aplicação	9
2.1.2	Camada de Transporte	9
2.1.3	Camada de Rede	10
2.1.4	Camada de Enlace	10
2.1.5	Camada Física	12
2.2	Tecnologias para Comunicação Sem Fio	13
2.2.1	IEEE 802.15	14
2.2.2	IEEE 802.15.1 - <i>Bluetooth</i>	14
2.2.3	IEEE 802.15.4	17
2.3	<i>Physical Web</i>	18
2.3.1	<i>Beacons</i>	19
2.3.2	Protocolos de Encapsulamento de Informações	20
2.4	Discussão	22
3	Mecanismos para Localização de Dispositivos BLE	23
3.1	<i>Fingerprinting</i>	25
3.2	<i>Time Difference of Arrival</i> e <i>Time of Arrival</i>	25
3.3	<i>Angle of Arrival</i>	26
3.4	Modelos de Propagação do Sinal	26

3.5 Ruído do RSSI	29
3.6 Trilateração	30
3.7 Estado da Arte	31
3.8 Discussão	36
4 Avaliação de Técnicas de Localização de Dispositivos BLE	37
4.1 Metodologia de Coleta de Informações	37
4.2 Flutuação da Intensidade do Sinal	41
4.3 Análise de Filtros	42
4.4 Ambientes de Teste	46
4.5 Resultados	47
4.5.1 Calibração dos Modelos de Propagação	48
4.5.2 Calibração da Convergência de RSSI	48
4.5.3 Estimativa de Distância	50
4.5.4 Posicionamento	53
4.6 Prova de Conceito	54
5 Conclusão	58
Referências	60

Lista de Figuras

1.1	Possibilidade para uso de <i>Physical Web</i>	2
1.2	<i>Beacon</i> Ion KST <i>Technologies</i>	3
2.1	Caminho de uma mensagem desde o remetente até o destinatário	8
2.2	Camadas da pilha TCP/IP	9
2.3	Endereço MAC de acordo com a IEEE	11
2.4	Exemplo do cálculo para o CRC	12
2.5	Abrangência das diferentes redes	13
2.6	Dispositivo mestre estabelecendo conexões com os escravos	14
2.7	Relógio monitor de batimentos cardíacos	15
2.8	Modelo de pacotes <i>Bluetooth</i>	16
2.9	Canais para anúncio BLE	17
2.10	Espectro BLE X <i>Wi-Fi</i>	17
2.11	Coexistência IEEE 802.15.4 e IEEE 802.11	18
2.12	Arquitetura <i>Zigbee</i> (adaptado de Rivero [29])	18
2.13	Modelos de dispositivos BLE	19
2.14	Especificação da moldura do iBeacon	20
2.15	Moldura dos pacotes UID,URL e TLM do Eddystone	22
3.1	Divisão entre maneiras de localização e posicionamento	24
3.2	Posicionamento por TDoA	26
3.3	Posicionamento por AoA	26
3.4	Ruído do valor RSSI	29
3.5	Trilateração	31
3.6	Arquitetura do sistema (implementado por Deordica e Alexandru [44])	33
4.1	Fluxograma da interface preliminar	39
4.2	Telas da interface preliminar	40
4.3	Testes de flutuação do sinal de potência	42
4.4	Variância do RSSI presente na Média Móvel	43

4.5	Variância RSSI presente no filtro de Kalman	44
4.6	Variância RSSI presente no Filtro Duplo	45
4.7	RSSI ao longo do tempo observado a 1m de distância	46
4.8	Experimento de desvanecimento de RSSI e estimativa de distância	46
4.9	Experimentos de posicionamento	47
4.10	Variância no filtro Kalman devido a mudança de valores iniciais	49
4.11	Quantidade de leituras para convergência no filtro Kalman	49
4.12	Distância máxima devido ao desvanecimento do sinal	51
4.13	Quantidade de leituras para convergências - Média Móvel e Filtro Duplo	52
4.14	Comparação de estimativa de distância - Ion	52
4.15	Trilateração real e estimada	54
4.16	Fluxograma para interface caso de uso	55
4.17	Tela da aplicação <i>food truck</i>	56

Lista de Tabelas

2.1	Tipos de PDU	16
2.2	Indicação para tipos de moldura	21
2.3	Prefixo para terceiro <i>byte</i> Eddystone-URL	22
3.1	Comparação entre os modelos <i>Frii's Transmission Model</i> , LDPLM e RPLM	28
3.2	Matriz de confusão para um dos classificadores (retirada de Filippopolitis <i>et al.</i> [45])	35
3.3	Comparação entre a distância real e a calculada (retirada de Vargas [2]) . .	36
4.1	Comparação entre os modelos <i>Frii's Transmission Model</i> , LDPLM e RPLM	48
4.2	Valores iniciais e leituras necessários para convergência do filtro de Kalman	50
4.3	Estimativa de distância - Ion	52
4.4	Variação e desvio padrão para cada metro e filtragem	53
4.5	Posição real e estimada	54
4.6	Checação de desencadeamento de eventos	57

Lista de Abreviaturas e Siglas

AoA *Angle of Arrival.*

API *Application Program Interface.*

ARP *Address Resolution Protocol.*

ARPANET *Advanced Research Projects Agency Network.*

BLE *Bluetooth Low Energy.*

BPS *Bluetooth Positioning System.*

BSIG *Bluetooth Special Interest Group.*

CEO *Chief Executive Officer.*

COMNET *Computer Network Laboratory.*

CRC *Cyclic Redundancy Check.*

CSV *Comma Separated Values.*

DF *Filtro Duplo (do Inglês, Double Filter).*

DNS *Domain Name System.*

DoA *Direction of Arrival.*

FTP *File Transfer Protocol.*

GLONASS *Global Navigation Satellite System.*

GPS *Global Positioning System.*

GSM *Global System for Mobile communication.*

HTTP *Hypertext Transfer Protocol.*

ID *Identification.*

IEEE *Institute of Electrical and Electronics Engineers.*

IoT *Internet das Coisas (do Inglês, Internet of Things).*

IP *Internet Protocol.*

IRDA *Infrared Data Association.*

ISM *Industrial, Scientific and Medical.*

K *Kalman.*

LAN *Redes Locais (do Inglês, Local Area Network).*

LDPLM *Log Distance Path Loss Model.*

LoS *Linha de visada (do Inglês, Line of Sight).*

MAC *Media Access Control.*

MAN *Redes Metropolitanas (do Inglês, Metropolitan Area Network).*

MM *Média Móvel.*

PAN *Redes Pessoais (do Inglês, Personal Area Network).*

PDA *Personal Digital Assistant.*

PDU *Packet Data Unit.*

PPP *Point-to-Point Protocol.*

PW *Physical Web.*

RPLM *Rappaport's Path Loss Model.*

RSSI *Indicador de Intensidade do Sinal Recebido (do Inglês, Received Signal Strength Indicator).*

SDK *Software Development Kit.*

SF *Sem Filtro.*

SMTP *Simple Mail Transfer Protocol.*

SPI *Serial Peripheral Interface.*

TCP *Transmission Control Protocol.*

TDoA *Time Difference of Arrival.*

TLM *Telemetry.*

ToA *Time of Arrival.*

UDP *User Datagram Protocol.*

UID *Unencrypted Identification.*

UnB *Universidade de Brasília.*

URL *Uniform Resource Locator.*

USB *Universal Serial Bus.*

UUID *Universally Unique Identifier.*

WAN *Redes de Longa Distância (do Inglês, Wide Area Network).*

WLAN *Wireless Local Area Network.*

WPAN *Redes Pessoais Sem Fio (do Inglês, Wireless Personal Area Network).*

WWW *World Wide Web.*

Capítulo 1

Introdução

Com o advento de novas tecnologias, o mundo volta a se revolucionar como nos anos 1990, àquela época por meio da Internet e da *World Wide Web* (WWW), agora pela Internet das coisas e pela *Physical Web* (PW). De forma semelhante ao passado, faz-se necessário identificar e entender a diferença entre as duas entidades [1].

A Internet das Coisas (do Inglês, *Internet of Things*) (IoT) representa o conjunto de dispositivos capaz de conectar com a internet e proporcionar atividades relacionadas ao meio físico, como sensores e dispositivos que possam emitir informações sobre o ambiente diretamente ao dispositivo do usuário (como exemplo, *beacons* [2], relógios inteligentes, *smartphones* entre outros). Já a *Physical Web* corresponde a uma camada global, ubíqua, que traz fluidez a esses dispositivos, proporcionando um trabalho mais eficiente com relação à facilidade de transmissão de informações aos usuários [3].

A *Physical Web* ou Internet interativa compreende um conceito de tecnologia que visa fornecer interações contínuas com objetos físicos e suas respectivas localizações, muitas vezes utiliza de sinais transmitidos por *beacons*, pequenos dispositivos que funcionam como faróis modernos, enviando um sinal para que receptores próximos os identifiquem, como marinheiros acompanham a luz de um farol para se orientarem na localização.

PW e *beacons* não constituem novidade, posto já terem sido estudados e até mesmo implementados alguns sistemas empregando sensores infravermelhos e *Personal Digital Assistant* (PDA), mas sob a nomenclatura de *Web Presence* [4]. Muito parecida com a PW, tem como motivação interligar a *World Wide Web* com objetos físicos, possui projetos para seu desenvolvimento desde os anos 2000. Entretanto, devido as tecnologias vigentes à época seu custo tornou-se muito alto, dificultando sua popularização, além disso, outro fator limitante foi o consumo de energia, que tornou-se baixo apenas no ano 2010 com o *Bluetooth Low Energy* (BLE). O artigo da *Hewlett Packard Laboratories* (HP) [4] explora e exemplifica algumas técnicas para envio de informações, no formato *Uniform Resource Locator* (URL), assim como projetos de infraestrutura e arquitetura dos modelos

constantes na *Web Presence*.

Promovida pela *Google*, a PW [5] consiste em uma evolução da *Web Presence* [4] que utiliza a ideia de gatilhos a partir de localidade como em pesquisa realizada pela HP [6]. Com o intuito de simplificar a conexão entre Internet e dispositivos físicos, segundo o *Chief Executive Officer* (CEO) da BKON [7], a PW representa uma evolução natural em direção à simplicidade [8].

Para empresas, escolas, lojas e outros estabelecimentos, a tecnologia configura uma forma eficiente de comunicação com usuários. Por exemplo, uma loja pode enviar propagandas direcionadas para clientes potenciais; uma pessoa que se encontra na sessão de sapatos muito provavelmente possui interesse em receber descontos para este tipo de produto [9].

Um caso real vem acontecendo na escola de Brookwood - Alabama, Estados Unidos - onde decidiu-se utilizar *beacons* para melhorar a comunicação entre pais, alunos e docentes [10]. Agora, alunos da escola podem receber notícias referentes à programação semanal, professores podem enviar resumos das aulas, entre diversas outras aplicações dos *beacons*.

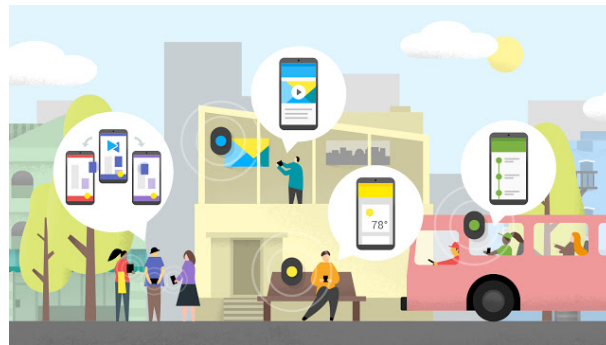


Figura 1.1: Possibilidade para uso de *Physical Web*

Antes da iniciativa da *Google*, o usuário necessitava baixar diversos aplicativos para cada loja, empresa ou local, mas com a PW tornou-se possível utilizar apenas um navegador *Web* [11]. Com o devido suporte e de acordo com a posição do *beacon*, por exemplo, um *beacon* em um parque poderia enviar informações de eventos programados para o mês, semana ou dia, como ilustrado em uma das funcionalidades presente na Figura 1.1. Assim, de maneira extramente simples, o usuário recebe um *link* em seu celular ou computador e passa a usufruir de suas facilidades, poupando o trabalho de manter e usar diferentes aplicativos para cada situação.

A PW encontra-se baseada no protocolo BLE para fornecer conteúdo e informações aos usuários em localizações bastante específicas com baixo consumo de energia. Segundo a *ABI research* [12], empresa norte-americana de pesquisa e consultoria sobre tecnologias emergentes, o número de *beacons* BLE deve superar a marca de 400 milhões em 2020.

Este crescimento encontra embasamento na expectativa de os dispositivos PW permearem os mais diversos ambientes, fornecendo informações sobre mercadorias, auxiliando na localização de objetos, ou direcionando o usuário para seu local de interesse, no qual o *Global Positioning System* (GPS) não esteja disponível. Estas informações podem estar acessíveis em uma lista de URL apresentada, por exemplo, na distância entre o usuário e o dispositivo transmissor. Recentemente, a *Google* disponibilizou um conjunto de protocolos em formato aberto com o intuito de difundir e permitir a utilização de *beacons* BLE, chamado Eddystone [13]. O protocolo Eddystone-URL define os parâmetros e os formatos para acesso às informações disponibilizadas pelo *beacon* BLE.

Criada com o intuito de proporcionar melhor autonomia energética aos dispositivos *Bluetooth*, a tecnologia BLE tornou possível a utilização de baterias moeda de 3V e de pilhas para manter ativos pequenos equipamentos chamados *beacons*. Um exemplo de *beacon* pode ser visto na Figura 1.2, estes equipamentos enviam mensagens por *Bluetooth*, codificadas no protocolo iBeacon (*Apple*) [14] ou Eddystone (*Google*) [11], captadas por um receptor de mesmo padrão e protocolo *Bluetooth* 4.0.



Figura 1.2: *Beacon* Ion KST *Technologies*

O protocolo de codificação utilizado não implica resultados muito diferentes, pois o auferido de fato será a intensidade do sinal. Porém, existem limitações em termos do raio de comunicação, tendo em vista a baixa potência do sinal de transmissão e suas atenuações em virtude do desvanecimento do sinal, variação do sinal recebido, reflexo do sinal entre outros problemas a ser considerados ao empregar a PW.

Para o estudo ora desenvolvido, o principal emprego da *Physical Web* encontra-se na identificação dos *beacons*, pois com eles se pode localizar espacialmente o receptor de sinais, gerando um navegador para ambientes internos, nos quais o sinal de satélites não alcança os dispositivos. Com isso, poder-se-á, por exemplo, guiar os usuários até uma sala de eventos, como palestras ou reuniões ou encontrar assentos em cinemas e estádios ou ainda transmitir informações específicas, de acordo com o local em que a pessoa se encontra, de tal modo que o usuário possa descobrir coisas novas a cada local visitado [9].

1.1 Problema

Embora a localização por meio de GPS tenha evoluído, permitindo, em certos casos, receber sinais de satélites para determinar um local dentro de um edifício, a localização resultante, em geral, não possui precisão suficiente para ser útil. Os sinais dos satélites encontram-se atenuados e dispersos por telhados, paredes e outros objetos. Além disto, outras tecnologias, como *Global System for Mobile communication* (GSM) ou Wi-Fi, necessitam de infraestrutura ou custo elevados. Portanto, tecnologias de redes Pessoais Sem Fio (do Inglês, *Wireless Personal Area Network*) (WPAN), como *Bluetooth*, buscam amenizar os custos mantendo precisão satisfatória em relação a localização. Neste contexto, a busca por mecanismos que permitam uma boa estimativa da localização de um objeto em ambientes fechados se mostra de grande interesse, devido as inúmeras possibilidades de aplicações [2], [15], [16]. O uso do Indicador de Intensidade do Sinal Recebido (do Inglês, *Received Signal Strength Indicator*) (RSSI), tem sido explorado, porém, problemas como, flutuações do sinal recebido, requerem algumas técnicas de atenuação para remoção de ruídos, como por exemplo, o uso de filtros [17]. Diante disso, define-se como questão central deste trabalho, a proposição de um sistema de proximidade e posicionamento, fundamentado na tecnologia BLE, para ambientes internos.

1.2 Objetivos

1.2.1 Objetivo Geral

A presente pesquisa possui como objetivo central verificar a acurácia da estimação de distância e do posicionamento de um receptor, baseando-se no sinal de potência de dispositivos *Bluetooth Low Energy*, e sua aplicabilidade no contexto de localização de objetos em ambientes fechados.

1.2.2 Objetivos Específicos

- Identificar e estimar a distância e o posicionamento de um dispositivo móvel receptor por meio do sinal BLE.
- Avaliar as técnicas de minimização de ruído, através de filtros passa-baixa.
- Aferir a acurácia das técnicas de estimativa de distância e localização em ambientes fechados.
- Desenvolver aplicação/protótipo que permita explorar e avaliar as características das estudadas.

1.3 Metodologia da Pesquisa

A metodologia aplicada neste trabalho consistiu inicialmente de uma investigação do estado da arte dos principais tópicos abordados. Nesta revisão, utilizou-se artigos da área, bem como livros e textos técnicos que permitiram identificar os principais avanços relativos às técnicas de estimativa de distância entre dispositivos sem fio em ambientes internos. Em seguida, avaliou-se as técnicas existentes, seus benefícios e limitações.

Após a fase inicial, passou-se para o desenvolvimento do aplicativo preliminar equipado de ferramentas para armazenamento e coleta de dados requeridos pelo projeto. A partir de pesquisas laboratoriais realizadas com o aplicativo, observou-se a ocorrência de ruído presente no recebimento do sinal causado por reflexos do ambiente. Para compreender esse fenômeno retomou-se a pesquisa bibliográfica, objetivando buscar, nos fundamentos teóricos, o suporte necessário à explicação do problema identificado. Por fim, por meio de modelos de propagação de sinal e filtragem de dados, tornou-se possível concluir o projeto, uma vez que os objetivos pré-definidos foram alcançados.

1.4 Estrutura do Trabalho

Este trabalho organiza-se conforme a seguinte estrutura:

- Capítulo 2:
Apresenta-se a fundamentação teórica do trabalho que aborda: *I* o funcionamento das camadas de rede e da pilha TCP/IP; *II* as tecnologias para comunicação sem fio, em especial as relacionadas com redes pessoais sem fio; *III Physical Web*; *IV beacons*; *V* e os protocolos de encapsulamento para transmissão de mensagem.
- Capítulo 3:
Detalha-se os mecanismos para localização com foco em ambientes internos, para tal explica-se: *I* modelos de propagação de sinal; *II* métodos de minimização de ruídos por meio de técnicas de filtragem; *III* métodos de estimativa de distância; *IV* métodos de posicionamento; *V* este capítulo encerra com uma revisão dos principais trabalhos na área diretamente relacionada com o problema de localização em ambientes internos.
- Capítulo 4:
Apresenta-se e analisa-se as informações referentes à coleta de dados que ofereceu suporte para a elaboração da pesquisa.

- Capítulo 5:

Relata-se as conclusões e as recomendações de trabalhos futuros com relação a este trabalho.

Capítulo 2

Fundamentação Teórica

A fundamentação teórica visa nortear, objetivando a revisão de conceitos e de técnicas aplicados em sua elaboração. Assim, este capítulo intenta elucidar os diversos tópicos abordados nesta pesquisa, para situar as questões e limitações da *Physical Web*, como o funcionamento da pilha *Transmission Control Protocol (TCP)/Internet Protocol (IP)*, das tecnologias para comunicação sem fio, dos protocolos de encapsulamento de informação e dos *beacons*.

2.1 Pilha TCP/IP

O texto referente a esta seção tem como base os autores Kurose e Ross [18], Tanenbaum [19] e Leon-Garcia e Widjaja [20]. Segundo Kurose e Ross [18], a Internet constitui um sistema extremamente complexo e multifacetado. Assim, na comunicação entre dois computadores (mesmo com *hardware* e *software* diferentes), foram instituídos conjuntos de padrões e normas, denominados, no ambiente de redes, de protocolos.

Os protocolos de rede se encontram, geralmente, projetados e organizados em uma estrutura de camadas, onde cada camada se torna responsável por promover e executar algumas ações específicas, fornecendo algum serviço à camada superior. Isto ocorre devido ao modelo ter como base o conceito de encapsulamento, ou seja, mensagens pertencentes a outra camada não podem ser modificadas, apenas encapsuladas junto às mensagens da camada atual. Desse modo, uma camada que esteja localizada acima de outra, não precisa conhecer o seu funcionamento interno, somente saber os serviços que aquela provê. Isso facilita a modularidade, pois quando se altera uma camada, precisa-se simplesmente assegurar que os serviços fornecidos antes por ela continuarão à disposição da camada superior, garantindo que tudo prossiga funcionando perfeitamente sem a necessidade de realizar alterações nas demais camadas.

Dois importantes protocolos se destacam nesta tarefa, o TCP e o IP, constituindo, respectivamente, a camada de transporte de serviço com conexão e a camada de rede, formando assim a pilha TCP/IP, projetada em uma arquitetura na qual as diversas camadas de software se comunicam somente com as camadas que se encontram imediatamente acima e abaixo.

A Figura 2.1 exemplifica, de uma maneira geral, a transmissão de uma mensagem entre dois usuários e ilustra o conceito de encapsulamento. Mostra o caminho que uma mensagem percorre, passando por cada camada desde a origem até o destino. Após ser gerada na camada de aplicação pelo remetente, a mensagem será repassada à camada inferior, no caso, a de transporte, a qual insere um cabeçalho, ou seja, informações extras a serem usadas por esta mesma camada no receptor. Neste ponto, a mensagem será denominada de segmento. Assim, cada camada insere suas informações de cabeçalho e repassa o pacote à camada subjacente, até chegar na camada física, que a depender do tipo de transmissão, transformará os *bits* dos dados em sinais elétricos ou ondas eletromagnéticas, para os transmitir à rede, até o destino final. O termo pacote encontra-se de forma genérica para caracterizar um conjunto de dados, pois em cada camada possui um nome específico.

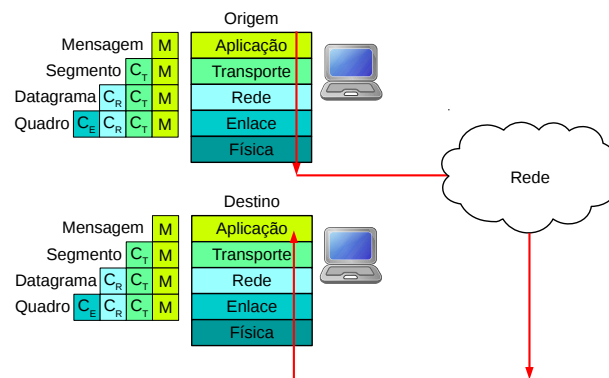


Figura 2.1: Caminho de uma mensagem desde o remetente até o destinatário

De acordo com Tanenbaun [19], a pilha TCP/IP surgiu da necessidade de conectar várias redes de maneira homogênea. Quando diversas universidades passaram a se conectar à *Advanced Research Projects Agency Network* (ARPANET), rede de pesquisa patrocinada pelo Departamento de Defesa dos Estados Unidos da América, usando diferentes protocolos, começaram a surgir problemas, que foram reparados com o advento do TCP/IP. Este protocolo permite a continuidade de comunicação entre dois *hosts* ativos, mesmo se algumas linhas de transmissão intermediárias falharem. O TCP/IP possui capacidade de se flexibilizar em diferentes tipos de aplicações com requisitos distintos, pois sendo orientado a conexões, permite uma entrega confiável e sem erros de um conjunto de dados. De

acordo com Kurose [18], o modelo divide-se em 5 camadas: Aplicação, Transporte, Rede, Enlace e Física, conforme apresentado na Figura 2.2.

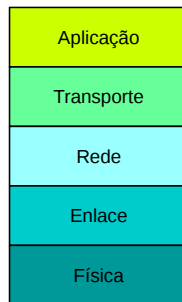


Figura 2.2: Camadas da pilha TCP/IP

2.1.1 Camada de Aplicação

Posicionada no topo, logo acima da camada de transporte, constitui os protocolos de níveis mais elevados, responsáveis pelos diferentes tipos de comunicação de dados entre os usuários, permitindo que sistemas finais diferentes se comuniquem pela rede. Apresenta como principais protocolos o *Simple Mail Transfer Protocol* (SMTP) que permite o envio de *e-mails*, o *Hypertext Transfer Protocol* (HTTP) utilizado principalmente em sites, o *File Transfer Protocol* (FTP) usado para transferência de arquivos, e o *Domain Name System* (DNS) responsável por traduzir endereços IP para nomes e vice-versa, possibilitando a identificação e localização de um determinado *host* em um domínio.

2.1.2 Camada de Transporte

Localizada abaixo da camada de aplicação e acima da camada de rede mostra-se responsável por manter a comunicação entre dois computadores. Nesta camada, encontram-se dois principais protocolos, o TCP, que apresenta como característica estabelecimento de conexão orientada, para a qual devem ser seguidos alguns passos definidos no protocolo a fim de ocorrer a interação entre os usuários, permitindo a entrega confiável sem erro dos segmentos, a recuperação de perdidos ou com dados corrompidos e a eliminação dos duplicados. Conta também com mecanismos para controlar o fluxo das mensagens, evitando que o receptor seja sobrecarregado no caso de o transmissor enviar mais segmentos do que o receptor consegue suportar.

Já, o *User Datagram Protocol* (UDP), diferente do TCP, por não possuir mecanismos que garantam ao receptor o recebimento de mensagens e por não haver necessidade de se estabelecer uma conexão entre os *hosts*, definindo-se como não confiável. Embora existam meios de verificação da integridade dos segmentos, cabe ao programador implementar estes

meios. Sua principal vantagem compreende a velocidade de processamento, tornando seu uso muito comum em serviços que necessitem de fluxo de dados em tempo real como jogos *online* e *streaming* de áudio e vídeo.

2.1.3 Camada de Rede

Posicionada logo abaixo da camada de transporte desempenha o papel de permitir o deslocamento de pacotes do transmissor ao receptor, mesmo em redes diferentes. Para tal, utiliza o IP, um endereço virtual único, identificador de cada dispositivo na rede. Desse modo, quando um pacote chega à camada de Rede recebe endereços de IP do remetente e do destinatário, para que durante seu caminho, cada nó da rede, pelo qual o pacote passe, saiba para onde encaminhá-lo. Aqui, faz-se necessária a distinção de duas funções desta camada evitando-se, pois, serem confundidas ou usadas como sinônimas: o repasse e o roteamento. O repasse ocorre no interior de um roteador, assim, quando um pacote chega ao seu enlace de entrada, o roteador examina um campo do cabeçalho do pacote e por meio da tabela de repasse escolhe o enlace de saída para o qual o pacote será conduzido. Já, o roteamento envolve todos os roteadores pelos quais o pacote passa, desde sua origem até o destino, ou seja, diz respeito ao caminho ou rota percorrida pelo pacote até chegar ao seu destinatário. Para estipular as rotas de um pacote, os roteadores utilizam os algoritmos de roteamento, cuja função reside em definir o melhor caminho a ser tomado por ele.

2.1.4 Camada de Enlace

Logo abaixo da camada de rede, o enlace resulta na penúltima camada da pilha. Nela, roteadores e hospedeiros recebem o nome de nós e um canal de comunicação entre dois nós denomina-se enlace. Exerce a tarefa de enviar o datagrama (nome dado ao pacote na camada de enlace) do nó atual ao nó adjacente, por um único enlace de comunicação. Para tal, existem diferentes protocolos, como o *Ethernet*, o IEEE 802.11 - *Wi-Fi*, o *Point-to-Point Protocol* (PPP), entre outros. Vale salientar que um datagrama não necessariamente deve ser transportado pelo mesmo protocolo ao longo dos diferentes enlaces do caminho, posto que, a cada enlace, o protocolo pode ser diferente, o que não prejudica o movimento do datagrama para o próximo nó. Dentre os serviços oferecidos na camada de enlace figuram: enquadramento de dados, acesso ao enlace, entrega confiável, controle de fluxo, detecção de erros, correção de erros etc, que podem variar dependendo do protocolo utilizado. Geralmente, a camada de enlace encontra-se implementada em um adaptador de rede. Nesta camada, cada nó possui um adaptador de rede que por sua vez possui

um endereço de camada de enlace identificador, conhecido como endereço *Media Access Control* (MAC).

Endereços MAC

Endereços físicos únicos (geralmente expressos em notação hexadecimal), identificadores do *hardware* a ser utilizado para se conectar à rede. Formados por um conjunto de 6 *bytes* de comprimento, o que possibilita 2^{48} endereços diferentes. Como dito inicialmente, os endereços MAC revelam-se únicos para cada adaptador e para garantir tal unicidade, o espaço físico desses endereços encontra-se controlado pelo instituto IEEE.

Neste cenário, uma empresa que deseja fabricar adaptadores de rede, deve entrar em contato com IEEE, que fixará os primeiros 24 *bits* do endereço MAC, e os associará a tal empresa, ficando os últimos 24 *bits* disponíveis para a empresa solicitante criar diferentes combinações para seus adaptadores. A Figura 2.3 exemplifica esse processo. Como os nós possuem dois endereços diferentes, o IP e o MAC, torna-se necessário um mecanismo que realize a tradução entre eles, o que pode ser feito por meio do protocolo *Address Resolution Protocol* (ARP), maiores detalhes podem ser obtidos em [18].

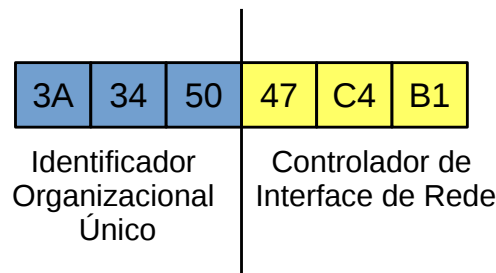


Figura 2.3: Endereço MAC de acordo com a IEEE

Detecção de Erros

A detecção de erros encontra-se implementada no *hardware* e ocorre na camada de enlace, consistindo na identificação de quadros que possam chegar com erros ao receptor. Vale ressaltar que as técnicas de detecção de erro podem apresentar falhas, ou seja, mesmo com métodos de detecção, existe a possibilidade de um quadro que contenha erros não ser detectado pela camada de enlace e o mesmo ser repassado à camada de rede com erros.

Quanto mais confiável o método de detecção, ou seja, quanto menor a chance de um falso negativo, maior o custo de processamento e maior a quantidade de *bits* a ser transmitida junto aos dados. Essas técnicas de detecção se mostram necessárias, pois, de acordo com o modo de transmissão do dados, os *bits* podem sofrer alguma alteração ao longo do caminho, devido aos campos magnéticos criados pela informação passando

através dos cabos, ou a dispositivos faltosos. Canais de transmissão como as fibras óticas possuem pouca interferência, logo a chance de que os dados sejam corrompidos se torna menor; já, em meios de transmissão sem fio, a quantidade de ruído aumenta, tornando-os bastante suscetíveis a erros. Quando detecta-se um erro, geralmente, necessita-se o reenvio do pacote. Entre os principais métodos de detecção de erro, destacam-se: *Cyclic Redundancy Check* (CRC), verificação de paridade e o *Checksum*.

Cyclic Redundancy Check

Técnica de detecção de erros disponível na camada de enlace, com funcionamento bastante simples. Primeiramente, remetente e destinatário devem acordar com uma sequência de $r + 1$ bits, chamada de gerador e denotada por G , na qual o bit mais à esquerda de G deve ser 1. O remetente que deseja enviar um dado de tamanho D escolhe r bits adicionais R e os anexa a D , de forma que ao G dividir $D + R$ o resultado seja exato. Assim, quando o destinatário recebe $D + R$, torna-se possível saber se o dado encontra-se correto ou não. A Figura 2.4 exemplifica o cálculo na verificação de uma mensagem recebida, sendo o resto da divisão igual a 0, conclui-se que a mensagem recebida não possui erros.

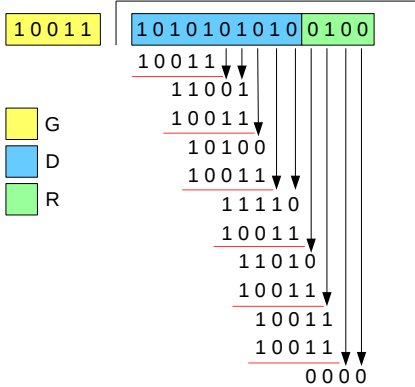


Figura 2.4: Exemplo do cálculo para o CRC

2.1.5 Camada Física

Última camada da pilha TCP/IP, na qual se encontra a parte física, responsável por enviar ou receber fisicamente os dados da rede. Há basicamente três tipos de transmissão: cabeada, sem fio e por satélite. Cada um desses modos de transmissão possui diferentes propriedades, capazes de interferir diretamente na performance e estrutura da rede utilizada. Sua unidade básica constitui-se em bits, que podem ser transmitidos por meio de sinais elétricos (no caso de redes cabeadas) ou de ondas eletromagnéticas (em redes sem fio).

2.2 Tecnologias para Comunicação Sem Fio

Redes de computadores crescem a cada dia, devido a esse crescimento existem algumas divisões que limitam o tamanho da rede [19], como Redes Pessoais (do Inglês, *Personal Area Network*) (PAN), Redes Locais (do Inglês, *Local Area Network*) (LAN), Redes Metropolitanas (do Inglês, *Metropolitan Area Network*) (MAN) e Redes de Longa Distância (do Inglês, *Wide Area Network*) (WAN). Cada uma dessas redes possui uma abrangência diferente, sendo WAN a maior entre elas, utilizada para transmissão de dados em longa distância, entre países e para outras WAN, LAN ou MAN. Redes LAN, por outro lado, conectam ambientes menores, como escolas, empresas, residências entre outros. A Figura 2.5 ilustra dispositivos e abrangência das redes citadas.

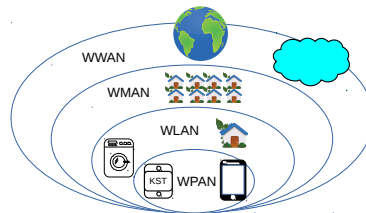


Figura 2.5: Abrangência das diferentes redes

As tecnologias mais comuns para este tipo de rede consistem em *Ethernet* e *Wi-Fi*. Por fim, uma PAN ou WPAN, representam redes pequenas, compostas por celulares, computadores, *tablets*, *beacons* comumente baseadas no padrão IEEE 802.15 [21]. O objetivo dessas redes se resume em facilitar operações entre dispositivos de casa ou empresas e sistemas através de comunicação de curta distância, em geral até 10 metros de distância, não necessariamente conectado à Internet e operam de mesma forma a outras redes através das camadas de rede. Nas redes WPAN congregam diversas tecnologias, entre elas a *Infrared Data Association* (IRDA), presente em controles de televisão; o padrão IEEE 802.15.4 encontrado em chips proprietários (*Zigbee*) [22]; o padrão IEEE 802.15.1 - *Bluetooth* [23], em controles de mídia para computadores, *mouses*, celulares e também *beacons* entre outros dispositivos. Portanto, encontra-se nas redes WPAN a utilização da IoT e da *Physical Web*.

O foco principal da pesquisa recai no *Bluetooth*, devido, não somente, ao objetivo de prover um aplicativo *Android* para localização e posicionamento em ambientes internos, como também pela raridade de celulares usufruírem da tecnologia contida em *chips Zigbee*, *Z-Wave* ou outras presentes na WPAN. Entretanto, dispositivos modernos, em geral, incorporam a tecnologia *Bluetooth* (versão 4.0 ou superior), o que oferece suporte para BLE, possibilitando um baixo consumo de energia, fator indispensável para dispositivos móveis.

2.2.1 IEEE 802.15

Define-se por uma família do *Institute of Electrical and Electronics Engineers* (IEEE) para especificar padrões de comunicação sem fio para as redes WPAN, especificamente as camadas física e de enlace. Dentre os padrões mais conhecidos destacam-se o IEEE 802.15.1 - *Bluetooth*, o IEEE 802.15.3, para WPAN que necessitam de taxas de transmissão altas, um máximo de 5,3GBits/s, e o IEEE 802.15.4 para taxas de transmissões pequenas.

2.2.2 IEEE 802.15.1 - *Bluetooth*

Bluetooth refere-se a uma tecnologia de comunicação sem fio que possibilita dispositivos se comunicarem por pequenas distâncias e opera sobre a banda não licenciada de 2,4 GHz. A tecnologia possui um grupo regulador conhecido por *Bluetooth Special Interest Group* (BSIG), padronizado pelo IEEE no IEEE 802.15.1 [24].

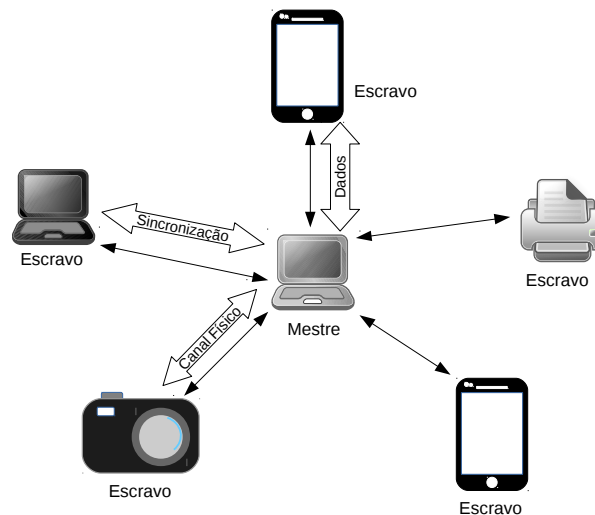


Figura 2.6: Dispositivo mestre estabelecendo conexões com os escravos

O protocolo *Bluetooth* utiliza 79 canais para transmissão de dados, começando na frequência 2402 e terminando em 2408 MHz. Para diminuir a interferência com outros sinais de rádio se utiliza uma técnica conhecida por *frequency hopping* [25], para o dado ser transmitido em canais diferentes periodicamente. Neste cenário, torna-se necessário estabelecer uma conexão entre os dispositivos.

Um dispositivo *Bluetooth* possui capacidade para conversar com até 7 outros dispositivos, simultaneamente, por meio do método chamado de “mestre-escravo”, de modo que quem inicia a conexão se torna o mestre da rede, demonstrado na Figura 2.6. Existem, também, dispositivos para os quais a conexão não se torna necessária, conhecidos como *broadcasters*. Neles a troca de informações se faz em mão única, ou seja, apenas enviam

informação, não podendo receber, como por exemplo, relógios inteligentes para monitoramento cardíaco, ilustrado na Figura 2.7. Com a disseminação da IoT e da PW, houve a necessidade de se desenvolver dispositivos *Bluetooth* que consumam menos energia, surgindo assim a tecnologia BLE.



Figura 2.7: Relógio monitor de batimentos cardíacos

Bluetooth Low-Energy

BLE ou *Bluetooth Smart* consiste em uma derivação do *Bluetooth Core 4.0* e objetiva reduzir a energia gasta durante o seu funcionamento. Isto ocorre devido ao modo de divulgação de pacotes, otimizado de tal maneira que o dispositivo permanece em estado dormente na maior parte de seu tempo de atividade, contribuindo também para o baixo consumo de energia. Sua potência dispõe de valor máximo em 0 dBm para um alcance teórico de 10m de distância, seu uso comum apresenta-se entre -40 a -70 dBm. O BLE resulta diferente do *Bluetooth* usual devido a uma técnica de modulação especial utilizada. Este tipo de *Bluetooth* possui capacidade de operar em quatro modos: periférico, central, *broadcast* e observador.

1. **Periférico** - Anunciante, opera em modo escravo e pode receber conexões.
2. **Central** - Varredor, pode iniciar conexões e trabalha como mestre em uma ou mais conexões.
3. **Broadcast** - Anunciante, não conectável.
4. **Observador** - Varredor, não conectável.

Para *beacons*, o primeiro e o terceiro modo, que se referem a anunciantes, constituem os principais. Ambos enviam o mesmo tipo de pacote, entretanto, mudando-se uma indicação eles se tornam conectáveis ou não. O modo conectável indica o dispositivo *Bluetooth Smart*, podendo receber conexões que mudam o estado de seus dados de maneira simples.

Muitos *beacons* possuem essa tecnologia para que seja possível modificar livremente o anúncio em transmissão. No caso do *Smart*, isso pode ser feito por conexão de aplicativos e/ou *Application Program Interface* (API)/*Software Development Kit* (SDK) da empresa fornecedora do *beacon*. Já para os não conectáveis se faz necessária a utilização de *Universal Serial Bus* (USB), *Serial Peripheral Interface* (SPI) ou alguma conexão cabeada semelhante. Na Tabela 2.1 encontra-se o código referente a cada tipo de funcionamento para o modo anunciante e varredor. Este código aparece associado a parte de *Packet Data Unit* (PDU) evidenciado na Figura 2.8. Neste pacote, há uma divisão de cabeçalho e de dados (*payload*), na primeira parte encontram-se os dados para identificação de modo de operação e na segunda, informações referentes ao anúncio.

Tabela 2.1: Tipos de PDU

PDU	Nome Pacote	Descrição
0000	ADV_IND	Conectável
0001	ADV_DIRECT_IND	Conectável
0010	ADV_NONCONN_IND	Não conectável
0011	SCAN_REQ	Varredor
0100	SCAN_RESP	Resposta
0101	CONNECT_REQ	Resposta
0110	ADV_SCAN_IND	Não conectável

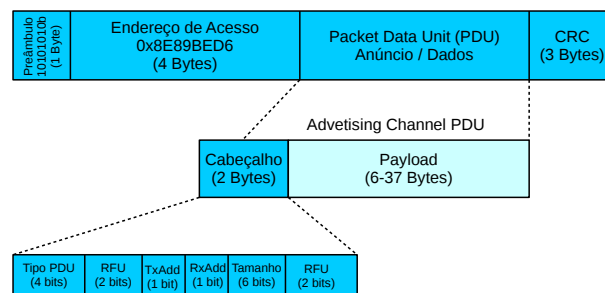


Figura 2.8: Modelo de pacotes *Bluetooth*

O BLE utiliza a mesma frequência de diversos outros protocolos de comunicação sem fio, banda 2,4 GHz *Industrial, Scientific and Medical* (ISM), como por exemplo *Wi-Fi* e telefones sem fio, e ainda eletrodomésticos como o micro-ondas, permitindo que ocorra interferência entre os sinais. Diferentemente do *Bluetooth* convencional, o BLE utiliza 40 canais de 2MHz cada, sendo três para anúncios, como apresentado na Figura 2.9.

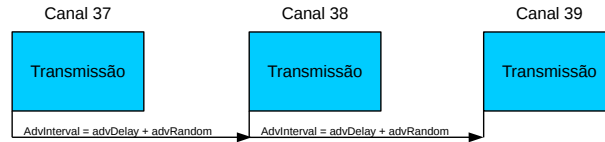


Figura 2.9: Canais para anúncio BLE

Esses canais foram escolhidos para gerar menos conflito com os mais usados pelo sinal *Wi-Fi*, como pode ser visto na Figura 2.10. Entretanto, como *Wi-Fi* possui maior potência em relação a dispositivos BLE, o sinal pode ser distorcido, provocando erro nos dados, caso os sinais sejam transmitidos muito próximos uns dos outros. Vale lembrar que, por mais estrategicamente posicionados para não interferir com os canais *Wi-Fi* mais usados, a possibilidade de interferência continuará existindo [26].

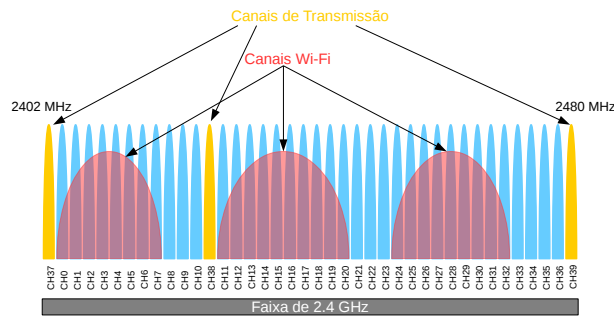


Figura 2.10: Espectro BLE X *Wi-Fi*

2.2.3 IEEE 802.15.4

De acordo com Gutierrez [27], o protocolo IEEE 802.15.4 se define por uma WPAN de baixo custo, em geral menor que uma rede *Bluetooth*, e portanto esse protocolo diminui o alcance do sinal e velocidade de transmissão de dados, apresentando, geralmente, taxas abaixo de 0,25 Mb/s e apenas 16 canais operando na faixa ISM na frequência de 2,4 GHz. Para trabalhar de maneira conjunta ao *Wi-Fi*, padrão IEEE 802.11, necessita-se que a frequência entre os canais ocorra a cada 5MHz em relação aos 25MHz do *Wi-Fi*, como mostra a Figura 2.11, isso permite diminuir a interferência entre os sinais. Para uma interferência ainda mais baixa, Liang *et al.* [28] expõem, entre outras técnicas, a possibilidade de identificar os pacotes para que um transmissor *Wi-Fi* aguarde, a fim de deixar o IEEE 802.15.4 avançar durante um certo período de tempo.

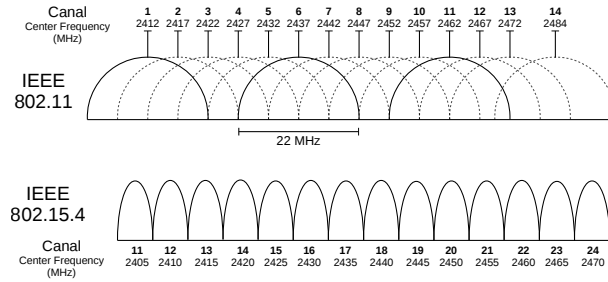


Figura 2.11: Coexistência IEEE 802.15.4 e IEEE 802.11

Dentre as tecnologias aqui apresentadas para o IEEE 802.15.4, destaca-se a *Zigbee*, desenvolvida para uso residencial. Essa tecnologia não se encontra completamente definida dentro do padrão da IEEE, pois apenas utiliza as camadas MAC e as camadas físicas, como mostra a Figura 2.12.

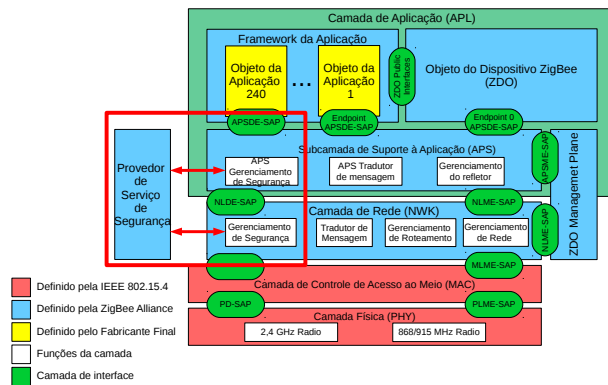


Figura 2.12: Arquitetura *Zigbee* (adaptado de Rivero [29])

Zigbee constrói sobre as camadas seus próprios métodos, a fim de garantir melhor custo/benefício em relação ao consumo de energia e a vazão. As camadas MAC e física mantêm o padrão IEEE 802.15.4, porém, a camada de rede e aplicação possuem interfaces e *softwares* especificados de maneira proprietária pela empresa *Zigbee Alliance*, criadora do protocolo. Assim, de acordo com a empresa, o *Zigbee* possui maior segurança, confiabilidade e poder de transmissão quando comparado ao padrão IEEE 802.15.4. Vale lembrar, que tais vantagens trazem consigo um custo elevado, em relação a concorrentes, devido a natureza privada de suas tecnologias.

2.3 *Physical Web*

Diversas empresas começaram a fabricar *hardware* com o suporte ao BLE, porém, cada dispositivo possuía sua própria especificação, dificultando para o desenvolvedor criar e dar manutenção às suas aplicações. Com a finalidade de promover a PW a *Google* projetou

o Eddystone, modelo de padronização da informações contidas no campo *payload* do protocolo BLE, rivalizando o modelo iBeacon da *Apple*.

A PW fundamenta-se em um conceito que tem como princípio a proximidade, relacionando objetos físicos com interações, proporcionando conteúdos com maior interesse ao usuário, baseado em sua localidade. As aplicações possíveis na PW mostram-se inúmeras e podem facilitar o dia-a-dia de diversas maneiras, por exemplo: um ponto de ônibus pode fornecer informações a respeito das linhas que por ali circulam; a coleira de um cachorro pode ser usada para localizá-lo ou informar às pessoas próximas a respeito de seu dono; um cliente em uma loja pode receber ofertas sobre produtos de seu interesse.

2.3.1 Beacons

Constituem dispositivos BLE, e assim aproveitam do gasto de energia minimizado, podendo utilizar baterias menores, como moeda ou pilhas, alguns modelos podem utilizar conexão USB ou uma tomada de energia para seu funcionamento. Além disso, possuem um tamanho pequeno, ideais para serem carregados nos bolsos ou colocados em mochilas e paredes de um ambiente, sem qualquer incomodo aos usuários. Alguns modelos se encontram na Figura 2.13.

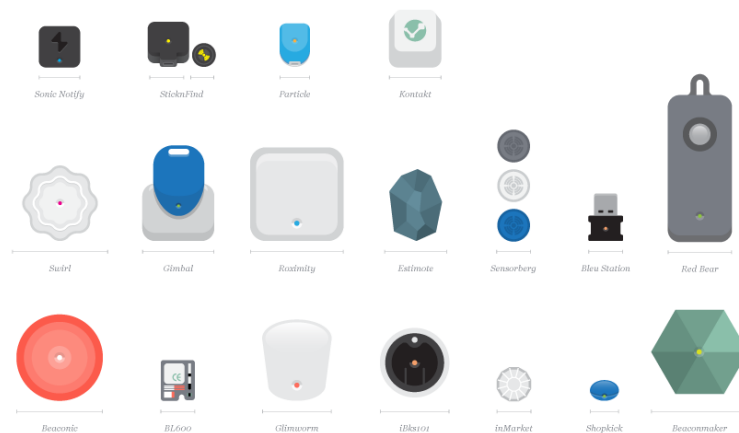


Figura 2.13: Modelos de dispositivos BLE

Beacons desempenham a função de enviar, periodicamente, pequenos pacotes de dados, padronizados em protocolos de encapsulamento de informações, utilizados para formatar o campo de *payload* do BLE. Possui, portanto, 37 *bytes* livres para envio de informação. Mesmo com o menor gasto de energia os *beacons* se mantêm em estado de espera (sono-lento) pela maior parte do tempo, reativando (acordando) em intervalos predefinidos para envio de informação.

2.3.2 Protocolos de Encapsulamento de Informações

Os protocolos de encapsulamento de informações objetivam segmentar 32 *bytes* dos 37 de dados, encontrados no *payload*, os últimos 5 *bytes* ficam reservados para uso futuro, em pequenos campos que contêm informações úteis para o cliente, como potência, frequência do sinal, nome do dispositivo e identificação. Os protocolos mais populares foram desenvolvidos por duas grandes empresas, *Apple* com o *iBeacon* e *Google* com *Eddystone*, estes serão detalhados abaixo.

iBeacon

O *iBeacon* consiste em um serviço de notificação sem fio, criado pela *Apple*, em 2013, com a finalidade de transmitir páginas *Web*, via *smartphones*, entre dois usuários. Considerado o primeiro modelo de formatação de dados BLE bem aceito.

Este sistema se encontra composto por campos, sendo que o campo *Universally Unique Identifier* (UUID) especifica o aplicativo que usará o *beacon*. Todo dispositivo com mesma identificação se relaciona com um único *software*. Os campos *Major* e *Minor* ocupam-se em delimitar sub-regiões, onde *Major* demarca o subgrupo e *Minor* um *beacon* individual. O campo TX_{Power} representa a potência do dispositivo, calibrado pela da força do sinal recebido a 1m de distância do receptor [30]. A Figura 2.14 ilustra o formato do protocolo.

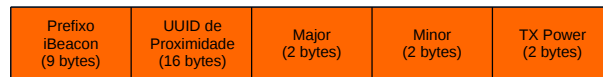


Figura 2.14: Especificação da moldura do *iBeacon*

Este protocolo apresenta algumas limitações, como a taxa de intervalo entre transmissões fixa em 100ms, o que provoca um gasto de energia maior que o necessário, fazendo com que a carga da bateria termine rapidamente. Além disso, devido à UUID estrita para cada aplicação, há a obrigatoriedade do uso de uma aplicação diferente para cada empresa ou loja que deseja implementar um sistema com *iBeacon* em seus projetos. Embora exista a possibilidade de portabilidade para *Android*, em geral, restringe-se a produtos *Apple*.

Eddystone

Desenvolvido pela *Google* e disponibilizado ao público em 2015, o *Eddystone* consiste em uma especificação de protocolo para definir o formato de mensagens utilizadas em *beacons*. Essas mensagens encontram-se re-encapsuladas em cima do protocolo de comunicação BLE. Entre os objetivos do *Eddystone* registram: o bom funcionamento tanto em *Android* quanto em *IOS*, a compatibilidade com a especificação *Bluetooth Core* e a

fácil implementação em diversos dispositivos BLE. O protocolo possui diferentes molduras para envio de informação, entre elas destacam-se como mais importantes: *Unencrypted Identification* (UID), URL, *Telemetry* (TLM) presentes na Tabela 2.2 e as especificações das molduras podem ser vistas na Figura 2.15.

Tabela 2.2: Indicação para tipos de moldura

Frame Type	High-Order 4 bits	Byte Value
UID	0000	0x00
URL	0001	0x10
TLM	0010	0x20
EID	0011	0x30
RESERVED	0100	0x40

- EddyStone-UID

Um tipo de moldura sem criptografia que transmite de maneira *broadcast* uma carga útil de 16 *bytes*. Dentro destes *bytes* há uma divisão, os primeiros 10 limitam o *namespace* e os últimos 6, a instância dos *beacons*.

O *namespace* serve para os reunir em um mesmo grupo virtual, desta forma se sabe qual *beacon* pertence a qual aplicação, empresa ou indivíduo. Por outro lado, o número de instância serve para identificar um *beacon* específico dentro destes grupos.

Além desse formato, o Eddystone conta também com *Ephemeral Identification* que possui as mesmas propriedades do UID, porém com criptografia associada, de modo que apenas pessoas autorizadas possam acessar a informação.

- EddyStone-URL

A moldura URL foi desenvolvida para ser a coluna da PW, tornando conteúdos *web* fortemente relacionados com o indivíduo e seu entorno. Nessa moldura, a divisão de *bytes* ocorre de modo que os últimos 17 servem ao propósito principal: codificar uma URL para ser transmitida pelo *beacon*. Os primeiros *bytes* resumem-se organizados como no UID, mas utiliza o terceiro *byte* como prefixo da URL, como exemplificado na Tabela 2.3.

Tabela 2.3: Prefixo para terceiro *byte* Eddystone-URL

Decimal	Hexa	Prefix
0	0x00	http://www.
1	0x01	https://www.
2	0x02	http://
3	0x03	https://

- EddyStone-TLM

Esta última moldura serve para enviar informações referentes ao *beacon* propriamente dito, tais como voltagem ou temperatura.

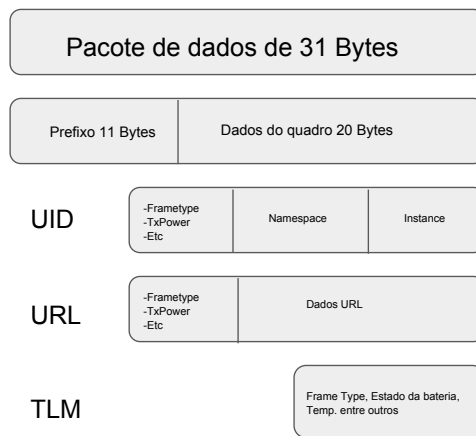


Figura 2.15: Moldura dos pacotes UID,URL e TLM do Eddystone

2.4 Discussão

Aborda-se a fundamentação teórica para situar o leitor acerca dos assuntos necessários para compreensão do estudo. Primeiramente explica-se a estrutura da pilha TCP/IP. Segundamente, as tecnologias para comunicação sem fio em redes WPAN e, por fim, defini-se a *Physical Web* e os dispositivos e protocolos utilizados em seu funcionamento. No Capítulo 3 será explicado os mecanismos para localização de dispositivos BLE.

Capítulo 3

Mecanismos para Localização de Dispositivos BLE

Mecanismos de localização apresentam raízes em fins militares, para rastreamento de soldados feridos em campo ou para entrega de suprimentos, como comida, munição e armas. Mas, atualmente, não se restringe a causas militares, pois a população civil e a própria área de segurança e da saúde passaram a notar utilidade dos sistemas de posicionamento em inúmeras situações, como indicação de caminhos mais eficientes para chegar ao seu destino, quer por policiais ou em ambulâncias, quer por pessoas comuns na realização de uma viagem sem perder tempo com mapas impressos e muitas vezes imprecisos.

Diversos métodos foram utilizados para se obter a localização e o posicionamento de transmissores ou receptores, dentre eles destacam-se: *Time of Arrival* (ToA), *Time Difference of Arrival* (TDoA), *Angle of Arrival* (AoA), RSSI e *Fingerprinting* [31], [32]. Cada um deles possui vantagens e desvantagens relacionadas ao ambiente em que se deseja utilizar, por exemplo, o GPS possui boa acurácia, em torno de 3 metros, para campos abertos ou estradas, e utiliza o sistema ToA. Assim, a partir do tempo gasto para receber o sinal estima-se a distância do satélite e um receptor. Necessitando-se de Linha de visada (do Inglês, *Line of Sight*) (LoS) dos satélites, pelo menos 3 e preferencialmente 4 deles, para que seja possível utilizar o método de multilateração ou multiangulação, na obtenção do posicionamento do dispositivo. Entretanto, em cidades com prédios altos, o sinal do satélite pode ser bloqueado devido ao concreto e metais presentes nas paredes das construções, dificultando a localização e o posicionamento, não somente em ambientes internos como também nos ambientes externos, devido à quantidade de obstáculos ao redor do receptor [33].

Outra solução quanto aos mecanismos de localização nas cidades, surgiu dos sinais de telefones móveis. As torres de comunicação, utilizam o modelo AoA, que se baseia na direção e ângulo das torres referenciais conhecidas, alcançando uma boa estimativa

para a localização do dispositivo receptor. Outro fator positivo reside na proximidade muito maior em comparação a um satélite no espaço, o sinal possui melhor potência para transpor os prédios e o dispositivo atinge um sinal satisfatório. Este método vem sendo empregado por órgãos de segurança pública na identificação de locais e telefones utilizados em crimes e informações falsas para números de emergência [34].

Atualmente existem diversos sensores móveis, nos quais, na grande maioria, encontram-se presentes nos dispositivos de baixa potência computacional, como celulares. Assim, faz-se necessário algoritmos eficientes, a fim de manter um bom tempo de resposta para o usuário [17] ou enviar as informações contidas no dispositivo a um servidor, para que se calcule a localização desejada [35].

Existem dois mecanismos principais de localização e posicionamento, sem infraestrutura, utilizados para dispositivos móveis: *Fingerprinting* e RSSI, demonstrado na Figura 3.1. Já, dependente de infraestrutura há o GPS ou similares, como o *Global Navigation Satellite System* (GLONASS) [36] e as torres de comunicação para telefones móveis. O trabalho, ora desenvolvido, terá foco no segundo mecanismo, sem dependência de infraestrutura, assim por meio de RSSI desenvolve-se um sistema capaz de localizar objetos em ambientes fechados.

A grande dificuldade de estimativas por meio de RSSI consiste em determinar a distância entre os nós com baixo erro, pois as leituras RSSI não se apresentam tão boas quanto outros métodos, devido aos modelos de antena, sem direção e controle de ganho, bem como a potência dos transmissores ser menor. Segundo Dalce [17], Wang [37] e Viswanathan [38], alguns modelos como *Friis's*, *Two Rays*, *Shadowing*, *Log Distance Path Model* e *Rapaport's* foram testados, apresentando resultados satisfatórios. A estimativa da distância também pode ser melhorada por meio da calibração, possibilitando diminuir a margem de erro em aproximadamente 15% de acordo com Whitehouse *et al.* [39].

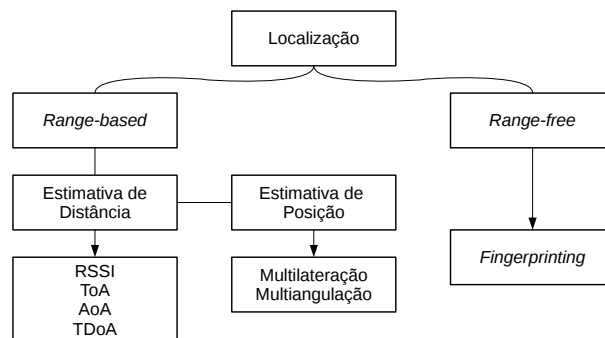


Figura 3.1: Divisão entre maneiras de localização e posicionamento

3.1 *Fingerprinting*

Método por reconhecimento de padrões, geralmente dividido em duas fases, *offline* (aprendizagem) e *online* (execução). A primeira, resume-se em coletar informação de RSSI do local em que se deseja implementar o sistema, estudando-se as calibrações de RSSI no ambiente e armazenando este dado junto da informação espacial, assim como posteriormente mapeando as leituras para cada área. A segunda, consiste na coleta do RSSI pelo dispositivo alvo, de posição desconhecida, e no envio para um servidor executar um algoritmo de posicionamento, a fim de se encontrar a região mais provável para sua localização. De acordo com as leituras RSSI, localiza-se regiões onde o receptor pode se encontrar no mapeamento da fase anterior.

Este sistema não se mostra muito interessante para locais dinâmicos [17], que sofrem mudanças frequentes, seja com troca de objetos de lugar ou pessoas se movendo, isto devido as pré-leituras se modificarem em função dos objetos que se encontram na frente do transmissor e/ou receptor. Caso o ambiente sofra muitas alterações se faz necessário refazer a primeira fase, fato que ocorre com frequência em redes móveis.

3.2 *Time Difference of Arrival e Time of Arrival*

ToA utiliza o tempo de recebimento do sinal para estimar a distância entre os dispositivos. De forma semelhante, TDoA realiza os mesmos passos, mas utilizando a diferença de tempo entre dois sinais recebidos em intervalos diferentes. A partir desta diferença ou, no caso de ToA, do tempo levado para receber o sinal, estima-se a distância entre os dispositivos através da multiplicação deste tempo pela velocidade de propagação da onda. Realiza-se o processo de estimativa diversas vezes ao longo de diferentes intervalos, a fim de manter atualizado a distância estimada, um exemplo pode ser visto na Figura 3.2.

Devido a sincronização de tempo necessária para obter uma estimativa com boa acurácia, as técnicas raramente tem utilidade em contextos *Bluetooth*, pois ele não permite a sincronização com precisão suficiente.

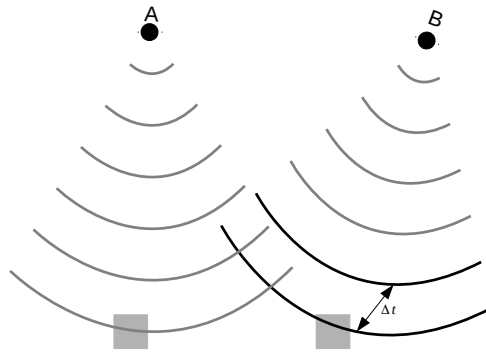


Figura 3.2: Posicionamento por TDoA

3.3 *Angle of Arrival*

AoA, ou *Direction of Arrival* (DoA), baseia-se na técnica TDoA e constrói uma matriz de antenas. A partir da medição de cada TDoA da matriz, estima-se a direção e o ângulo do receptor, como ilustrado na Figura 3.3. Apresenta ótimo desempenho onde há grande campo de visão, porém para ambientes internos se mostra ineficiente devido a obstáculos que causam efeito de sombreamento e geram flutuações nas leituras da onda e devido a reflexo do sinal. Tanto em ambientes externos quanto internos ocorre o efeito de multipercurso do sinal, em que este transita por diferentes caminhos até alcançar o dispositivo receptor, resultando no enfraquecimento da onda e conseqüentemente na imprecisão da estimativa. Os dispêndios computacionais dos receptores e físico das antenas e dispositivos receptores se tornam maiores devido a necessidade de uma matriz de antenas e de sincronização entre dispositivos, portanto, no contexto *Bluetooth*, o uso do método se torna inviável seu uso.

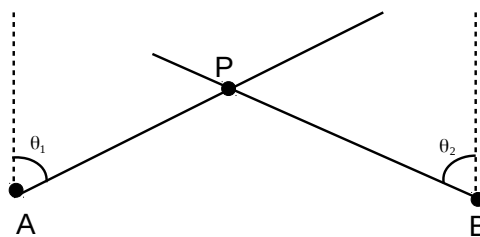


Figura 3.3: Posicionamento por AoA

3.4 Modelos de Propagação do Sinal

Devido a fatores como interferência de outros sinais sem fio, meio em que se propaga o sinal, entre outros motivos, a força da comunicação decai com a distância percorrida.

Diversos modelos foram propostos e utilizados na literatura para melhor prever o desvanecimento, entre eles: *Friis's*, *Log Distance Path Loss Model* (LDPLM), *Rappaport's Path Loss Model* (RPLM), *Two Rays*, *Shadowing* [17].

1. ***Frii's Transmission Model***: O modelo de propagação em espaço livre, proposto por *Harald Trap Friis*, objetiva prever a potência de sinal recebido quando não há obstáculos entre os campos de visão do transmissor e do receptor. O modelo consiste na Equação de *Friis* apresentada a seguir:

$$P_r = P_t \left(\frac{\lambda}{4\pi d} \right)^n G_t G_r \quad (3.1)$$

Onde P_r representa a potência recebida, P_t a potência transmitida (em Watts ou Miliwatts), λ o comprimento de onda em metros, d a distância entre transmissor e receptor em metros, n o coeficiente de perda de percurso, G_t o ganho da antena no transmissor e G_r o ganho da antena no receptor. Para uma antena ideal e $n = 2$, coeficiente para o espaço livre, a equação se da por:

$$\frac{P_t}{P_r} = \frac{(4\pi f d)^2}{c^2} \quad (3.2)$$

Onde f representa a frequência em Hertz, e c a velocidade da luz, aproximadamente $3 * 10^8 m/s$. Isolando a distância d , obtém-se:

$$d = \sqrt{\frac{P_t}{P_r}} \left(\frac{c}{4\pi f} \right) \quad (3.3)$$

Porém ainda há variáveis como ganho da antena e perdas do circuito, variáveis estas de difícil acesso quando se utiliza *beacons* e telefones móveis. Um modelo baseado em *Friis* e modificado para uso em dispositivos móveis já foi utilizado por diversos autores como, por exemplo, Vargas [2] ou Dalce *et al.* [17], revelando-se eficaz.

2. ***Log Distance Path Loss Model***: Modelo padrão para desvanecimento de potência de sinal para antenas sem controle de direção, ou conhecimento de ganho da antena. Amplamente utilizado por Wang [37] e Subhan [15].

$$RSSI = RSSI_0 - 10 * n * \log_{10}(d) \quad (3.4)$$

Onde $RSSI$ se refere ao valor estimado para RSSI, $RSSI_0$ o valor para o RSSI a 1m de distância do transmissor, n coeficiente de perda de percurso, para ambientes

livres $n = 2$, e d a distância. Para calcular a distância basta isolar d :

$$d = 10^{\frac{RSSI_0 - RSSI}{10 * n}} \quad (3.5)$$

3. **Rapapport's Path Loss Model:** Adiciona-se o termo X no modelo original LD-PLM para considerar o fenômeno *Slow-fading*, que ocorre devido a obstruções no caminho. O termo X , assim como n , coeficiente de perda de percurso, objetiva dar melhor controle ao modelo, em geral, utiliza-se a variância como valor para o termo X .

$$RSSI = RSSI_0 - 10 * n * \log_{10}(d) + X \quad (3.6)$$

Para calcular a distância basta isolar d , assim como no modelo anterior:

$$d = 10^{\frac{RSSI_0 - RSSI + X}{10 * n}} \quad (3.7)$$

A Tabela 3.1 evidencia o comportamento dos modelos para a estimativa de distância e equivalência entre eles em caso ideal, ou seja, espaço livre, antenas isotrópicas (ganhos iguais a 0) e desvanecimentos ideais.

Tabela 3.1: Comparação entre os modelos *Frii's Transmission Model*, LDPLM e RPLM

RSSI (dB)	Frii's (m)	LDPLM (m)	RPLM (m)	Distância Real (m)
-56,04	1,00	1,00	1,00	1,00
-62,06	2,00	1,99	1,99	2,00
-65,59	3,00	3,00	3,00	3,00
-68,09	4,00	4,00	4,00	4,00
-70,00	5,00	4,98	4,98	5,00
Erro Médio (m)	0,00	0,00	0,00	0,00

No cálculo para o modelo de Frii's, necessitou-se transformar P_t e P_r de dB para Miliwatts, de acordo com a equação [40]:

$$Power_{mW} = 10^{\frac{Power_{dB}}{10}} \quad (3.8)$$

Onde $Power_{dB}$ representa a potência em dB. Para P_t utilizou-se o valor de -16 dB ficando este igual a $10^{-1,6}$. Para P_r utilizou-se o RSSI recebido, mostrado na Tabela 3.1; f e c assumiram os valores $2,4 * 10^9$ e $3,0 * 10^8$, respectivamente. Nos modelos de LDPLM e RPLM o valor para $RSSI_0$ foi de $-56,04$ dB, valor esperado para o desvanecimento do

signal a 1m de distância do transmissor ideal, encontrado a partir da Equação (3.2), e $n = 2$ [2].

3.5 Ruído do RSSI

Como o mundo real contém diversos obstáculos, o sinal sofre variações, seja passando pelos lados do obstáculo ou sendo refletido por ele, ilustrado na Figura 3.4. Devido a essas variações que não seguem o desvanecimento padrão há a necessidade de implementar um filtro passa-baixa ou similar no sistema. Este filtro pode ser uma simples média ou filtros mais complexos, como Kalman [41].

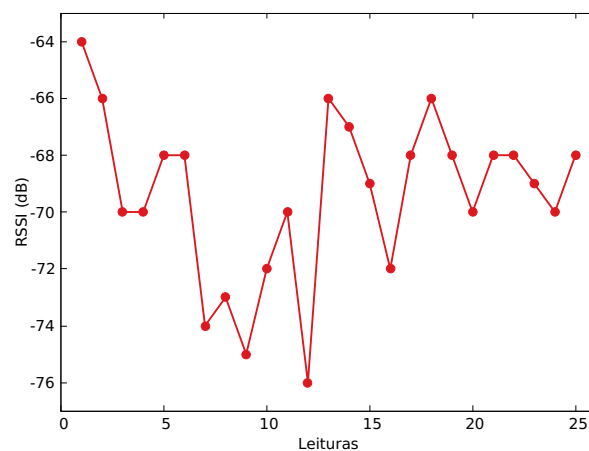


Figura 3.4: Ruído do valor RSSI

1. Média Móvel (MM) - Utiliza-se o conceito de uma média simples somado a movimentação de uma janela de dados, assim ajustando o tamanho da janela pode-se fazer uma média de acordo com a chegada de valores. Portanto:

$$M = \frac{\sum a_i}{k} \quad (3.9)$$

onde k representa o tamanho da janela, e i vai de 1 a k .

2. Kalman (K) [41] - O objetivo de utilizar um filtro como o de Kalman consiste em uma melhor remoção de ruídos e portanto na variação da recepção do sinal. Este tipo de filtro vem se tornando cada vez mais utilizado em filtragens de dados para RSSI de *beacons*, assim como para práticas de localização e posicionamento em ambientes internos, conforme estudos realizados por Bulten [42] e Dalce [17].

O filtro de Kalman se baseia em um estimador de estados que presume o valor de uma variável a partir de entradas com ruído, expressado por um algoritmo que

se fundamenta no histórico das entradas. Além disso, o filtro de Kalman assume um modelo linear, portanto os estados e as estimativas devem ser transformações lineares. O modelo geral apresenta a forma:

$$x_t = A_t x_{t-1} + B_t u_t + E_t \quad (3.10)$$

Define-se o estado x_t como uma combinação do estado anterior x_{t-1} dado uma matriz de transformação A , um controle u_t , como por exemplo o movimento do dispositivo, junto também de uma matriz de transformação B e o ruído processado, proveniente do próprio sistema E .

3. Filtro Duplo (do Inglês, *Double Filter*) (DF) - No estudo por Viswanatlan *et al.* [38], apresenta-se um novo modelo de filtro chamado *Double Filter*. De acordo com os autores, obteve-se um erro abaixo de 1 metro. O filtro fundamenta-se em duas variáveis de controle A e B . A controla o ganho do sinal, sendo um valor de 0,5 a 1, simbolizando as porcentagens de ganho, enquanto B representa um valor de controle para o erro propagado no sistema, variando de 0,05 a 0,5. A fórmula do modelo encontra-se evidenciada a seguir:

$$S_n = (1 - A_n)S_{n-1} + A_n(E(S_n) + (RSSI_n - E(S_n))B_n) \quad (3.11)$$

Onde S_n simboliza o novo valor do sinal filtrado, S_{n-1} , o valor anterior do sinal filtrado e $E(S_n)$, o valor de erro do sistema até o momento calculado através de uma média simples dos valores de RSSI filtrados de 0 a $n - 1$. A e B constituem vetores, se apresentam atualizados por:

$$Se((A_n > A_B) \& \& (B_n > B_b)) := A_n = W_n A_{n-1}, B_n = W_n B_{b-1} \quad (3.12)$$

Onde A_B , B_B e W_n representam valores de limitação, dados como 0,7, 0,05 e 0,9, respectivamente, de acordo com o estudo.

3.6 Trilateração

Este método apresenta como premissa o conhecimento de três posições de referência, fixas, e a distância em que cada uma se encontra com relação a um ponto, de posição desconhecida. Em tese, ocorre a formação de três círculos com raios de tamanho variável, cada um tendo como centro um dos pontos referenciais. Requer, portanto, a distância entre o nó alvo até os nós referência previamente ao cálculo da posição [37].

Suponha N_1 , N_2 e N_3 , representando os nós referências e N o nó alvo que se quer saber a posição. Conhecendo-se as posições em (x,y) dos nós referências e suas distâncias até o alvo, pode-se descrever a relação entre estas informações a partir de:

$$(x - x_1)^2 + (y - y_1)^2 = d_1^2 \quad (3.13)$$

$$(x - x_2)^2 + (y - y_2)^2 = d_2^2 \quad (3.14)$$

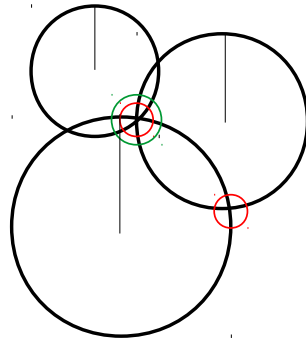


Figura 3.5: Trilateração

Assim fundamentado, geram-se dois círculos $C_1(N_1, d_1)$ e $C_2(N_2, d_2)$. Em que N situa-se na interseção entre estes dois, portanto há duas soluções [17]. Um terceiro círculo pode ser feito a partir de N_3 , seguindo-se a mesma fórmula para saber qual entre as duas soluções será a correta, como pode ser visto na Figura 3.5, em verde a solução certa.

Conforme apresentado na Figura 3.5, com apenas dois pontos de referência criam-se dois pontos prováveis onde o receptor poderia ser encontrado, círculos em vermelho. Porém, ao se utilizar um terceiro círculo, cria-se um ponto de encontro entre os referenciais, círculo em verde, proporcionando maior confiabilidade quanto ao posicionamento do receptor.

3.7 Estado da Arte

A ideia de posicionar objetos ou indivíduos no espaço baseando-se em potência de sinal teve início com a popularização dos padrões de comunicação sem fio disponibilizados pelas famílias IEEE 802.11 (WLAN) e IEEE 802.15 (WPAN). Neste contexto, surgiram projetos como *Rome*, *Guide* e outros mais novos continuam em processo de evolução nos dias atuais. Para embasar teoricamente este estudo, destacam-se projetos relevantes retratados nos artigos a seguir:

- Huang *et al.* [6]

O estudo baseia-se na proposta de que a informação se torna mais útil se apresentada

no momento temporal ou espacial correto e não em sua criação. Por exemplo, uma pessoa ao abrir uma geladeira nota que precisa comprar mais leite, e insere esta informação em seu celular, mas certamente se esqueceria mais tarde. Neste cenário, o interessante seria receber um lembrete novamente ao passar por perto de um mercado.

Desenvolveu-se *Rome*, inspirado no ditado “todos os caminhos levam a Roma”, um programa baseado em *Windows CE* que utiliza o protocolo TCP para serializar e enviar mensagens, chamadas de *Triggers*, a dispositivos portáteis como PDA ou *notebooks*, com regras definidas, como proximidade a um local ou data e horário, como 3 de março às 15 horas, para liberar a mensagem ao usuário.

- GUIDE [43]

Projeto da Universidade de *Lancaster*, com o objetivo de promover um contexto sensível para turistas da cidade de *Lancaster* na Inglaterra.

Utilizando *chips WaveLAN*, precursores ao protocolo *Wi-Fi* - 802.11, como *beacons* espalhados em pontos turísticos da cidade, tornou-se possível trazer informação aos usuários sobre os pontos em que se encontram. Para tal, utilizou-se a conexão da própria universidade, passando cabos para os terminais encontrados nas áreas turísticas. Através de páginas *Web*, tornou-se possível checar se o usuário acessa a Internet por meio de um dos terminais, caso positivo, carrega-se a página referente ao ponto turístico.

- Deordica e Alexandru [44]

Discute-se um sistema baseado na arquitetura cliente-servidor. O servidor consiste em um *Arduino* e um dispositivo BLE, com o propósito de transmitir o sinal a uma certa distância, a fim de que seja detectado e conectado quando o cliente estiver em seu raio de alcance. A Figura 3.6 representa a arquitetura do sistema cliente-servidor. Uma vez estabelecida a conexão, o servidor envia os dados correspondente à propaganda através do BLE ao cliente e encerra a conexão. Para a parte do cliente, foi desenvolvida uma aplicação *Android* instalada em um *Samsung Galaxy Note 2*, utilizado para os testes. A aplicação foi desenvolvida de maneira a captar todos os sinais BLE de dispositivos próximos, exibindo-os em uma lista, ordenada pela intensidade do sinal; a aplicação escolhe o dispositivo com o sinal mais forte (o mais próximo ao usuário), que se conecta ao servidor e envia pelo BLE a propaganda a ser exibida na tela do dispositivo do cliente.

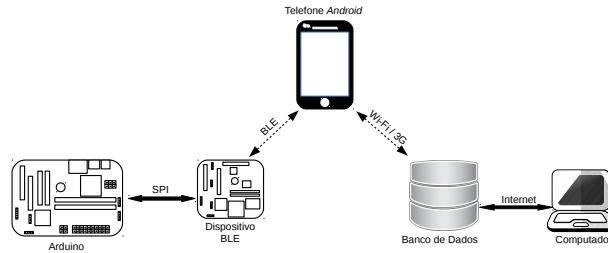


Figura 3.6: Arquitetura do sistema (implementado por Deordica e Alexandru [44])

- Galván-Tejada *et al.* [16]

Apresenta-se técnica para combinar as informações de distância de *Bluetooth* e *Wi-Fi*, assim como um algoritmo de posicionamento por meio de multilateração, baseado em ToA. Utilizou-se métodos de regressão linear, especificamente o de coeficientes de determinação, para ajustar o modelo logarítmico da propagação do sinal para os dispositivos utilizados.

Para o método de multilateração utilizou-se pelo menos 8 pontos referenciais para estimar a posição do nó receptor, fato que gera custos adicionais, porém aumenta consideravelmente a estimativa. De acordo com os resultados do estudo, o método proposto supera outros métodos com uma precisão de 0,87m de erro, contra 2m de, somente, trilateração ou 4m de uma combinação do sinal GPS e *Wi-Fi*.

- Subhan *et al.* [15]

Elabora-se proposta e teste de uma combinação de técnicas de posicionamento, *Fingerprinting* e trilateração em conjunto. Os testes foram realizados em um laboratório de $120m^2$, com dimensões de $10 \times 12m$. Para a emissão dos sinais *Bluetooth*, utilizou-se de *dongles* USB, com diferentes especificações, e como dispositivo móvel, um *Nokia 5130*. Na primeira fase do *fingerprinting* (chamada de aprendizagem), foram utilizadas duas maneiras diferentes para a coleta do RSSI: a primeira consiste de uma técnica orientada a conexão, que faz uso do *hcitool* e *hcidconf*, comandos para realizar medidas de sinal; a segunda maneira abarca o *Inquiry with RSSI*. A leitura do RSSI no primeiro método foi realizada com o dispositivo móvel movendo-se de 30 em 30 centímetros, enquanto no segundo, a distância movida foi de 1 em 1 metro, sendo realizadas mais de 50 leituras a cada passo. Para filtrar as inconsistências das leituras dos valores do RSSI, foi utilizado um Filtro Gradiente, que consegue atenuar tais inconsistências.

Para a estimação da distância, foi utilizado o *Free Space Propagation Model*, sendo os valores de $A = 47,42$ e o de $n = 1,5$ a melhor combinação encontrada. A técnica

híbrida proposta (utilizando *Fingerprinting* e trilateração) consiste em assumir os valores da fase *offline* do *Fingerprinting* na fase *online*, para a aplicação da trilateração. A média de erro da distância estimada, utilizando o RSSI sem nenhum filtro foi de 5,87 metros; já com o uso do Filtro Gradiente, esse erro caiu para 2,67 metros, significando uma melhora de 45% na precisão.

- Filippoupolitis *et al.* [45]

O estudo objetivou detectar a presença de um usuário em determinada área, por meio de sinal BLE e técnicas de aprendizagem de máquina. A abordagem consiste em usar *beacons* BLE instalados em ambientes internos que se comunicam com a aplicação móvel e permite que um sistema infira a localização do usuário. A aplicação móvel recebe pacotes BLE e os filtra, coletando informações apenas dos *beacons* utilizados no projeto. A aplicação envia informações como os valores de RSSI e ID para o servidor localizado em um centro remoto de controle que recebe essas informações e calcula a ocupação do ambiente. Os *beacons* utilizados consistem em minicomputadores *Raspberry Pi 2* equipados com módulos *Bluetooth 4 BLE*, que simulavam *iBeacons*. A aplicação móvel foi desenvolvida para *Android* e utilizada tanto na fase de coleta de dados como na fase normal de testes.

O ambiente de testes compreendeu um laboratório dividido em dois setores. O primeiro com dimensões aproximadas de 12x16 metros e o segundo com 13x16 metros, sendo estes quadriculados em unidades de $1m^2$. Em cada setor foram identificadas áreas sem acesso, obstáculos e espaço livre, sendo que em cada setor foram identificadas 5 áreas possíveis de ocupação. Para a coleta de dados, os autores caminharam pelos setores, passando nos pontos de cada uma das 10 áreas livres, aguardando 10 segundos antes de seguir ao próximo ponto. Para cada pacote BLE recebido foi salvo o ID e o RSSI e atribuído um rótulo identificando a área em que o dispositivo se encontrava. Dos dados coletados foram utilizados 80% para treinar o algoritmo classificador e 20% para testá-lo. Em cada setor, foram aplicados 3 diferentes classificadores e os resultados apresentados por meio de matrizes de confusão. A Tabela 3.2 representa uma das matrizes, onde as linhas denotam a classe atual da instância e as colunas, a classe prevista; a diagonal principal caracteriza a porcentagem de instâncias classificadas corretamente, restando as demais, incorretas. Os resultados obtidos foram satisfatórios, mostrando que a combinação de técnicas de aprendizagem de máquina e BLE pode ser uma abordagem promissora na estimativa de ocupação de um ambiente.

Tabela 3.2: Matriz de confusão para um dos classificadores (retirada de Filippoupolitis *et al.* [45])

	A1	A2	A3	A4	A5
A1	0,94	0,03	0,00	0,03	0,00
A2	0,00	0,93	0,04	0,00	0,04
A3	0,00	0,00	1,00	0,00	0,00
A4	0,14	0,00	0,00	0,86	0,00
A5	0,09	0,09	0,00	0,00	0,83

- Vargas [2]

Retratou a implementação de um sistema de navegação usando um conjunto de *beacons* e uma aplicação *Android*. Os *beacons* encontram-se posicionados em um certo ambiente; a aplicação coleta os dados referentes ao RSSI, estima a distância entre o receptor e os *beacons* e a aplicação ainda faz uso do giroscópio e da bússola integrados ao dispositivo móvel para detectar a direção em que o usuário se move; tanto os *beacons* como a posição do usuário e as distâncias entre eles aparecem em um mapa no dispositivo. O ambiente de testes consistiu em um apartamento de cerca de $50m^2$, com 2 *beacons* posicionados a $5m$ de distância um do outro. O dispositivo móvel utilizado para rodar a aplicação foi um *Samsung Galaxy S5* com o *Android* 5.0. Os *beacons* utilizados foram iBKS BLE *beacons* fabricados pela *Accent Systems*, configurados para enviar o sinal a cada 100 milisegundos, possibilitando coletar o RSSI aproximadamente 10 vezes por segundo. Na aplicação, o posicionamento do *beacon* ocorre manualmente; a distância apresentada na tela segue atualizada a cada 1 segundo, sendo esta a média das distâncias estimadas para as 10 leituras do RSSI, calculada com a fórmula de *Free Space Friis Model*. Para distâncias de até $3m$ o sistema apresentou valores relativamente próximos, como pode ser visto na Tabela 3.3. O estudo revelou que a navegação em um ambiente interno resulta-se possível, porém a tecnologia ainda possui algumas limitações que impossibilitam uma localização precisa, embora aceitável, a depender do propósito.

Tabela 3.3: Comparação entre a distância real e a calculada (retirada de Vargas [2])

Distância Real (m)	Média da Distância Calculada (m)
1,00	1,85
2,00	2,46
3,00	4,47
4,00	6,25
5,00	7,45

Atualmente, percebe-se o propósito de popularizar aplicações de posicionamento nos *smartphones Android e IOS*, no entanto existe um problema de inconsistência na leitura de RSSI, que pode ocorrer conforme o modelo de aparelho, seu invólucro, o ganho da antena, a posição, o gargalo no barramento de dados, entre outros. Isso causa grande infortúnio, pois se faz necessário homologar uma versão calibrada do software para cada aparelho. Por exemplo, em testes dentro de um apartamento, a menos de 2 metros do *beacon* a informação recebida poderia vir a gerar 10m ou mais de erro. Portanto, a calibração torna-se fundamental.

3.8 Discussão

Aborda-se os conhecimentos matemáticos acerca de mecanismos de localização de dispositivos *Bluetooth Low Energy*, para situar o leitor sobre os tópicos necessários para compreensão dos assuntos utilizados no estudo. Explica-se os principais métodos entre eles: *Fingerprinting*, TDoA, AoA e RSSI. Os modelos de propagação do sinal tem como intuito prever o comportamento do desvanecimento da intensidade do sinal de acordo com a distância do receptor ou transmissor, por fim, explana-se a técnica de posicionamento e faz-se a revisão do estado da arte. O Capítulo seguinte retrará à avaliação das técnicas discutidas neste.

Capítulo 4

Avaliação de Técnicas de Localização de Dispositivos BLE

O capítulo apresenta a utilização dos fundamentos e conhecimentos apresentados nas seções anteriores, com foco em alcançar os objetivos propostos no Tópico 1.2. Descreve as metodologias utilizadas para a coleta de dados, o *hardware* e o *software* necessários para a obter-se as informações desejadas, os problemas encontrados e as possíveis soluções, assim como os ambientes onde realizaram-se os testes. Ao final da seção expõe-se os resultados referentes a estimativa de distância, posicionamento e exemplo de aplicação.

4.1 Metodologia de Coleta de Informações

Necessitou-se da aquisição de *hardware* compatível com as tecnologias utilizadas para a realização do projeto. Os *beacons* foram escolhidos devido ao suporte às tecnologias necessárias ao estudo, assim como por motivos econômicos, posto não se tratar de dispositivo comercializado no mercado nacional, constituindo um fator limitante na sua seleção, pois ou o frete se apresentava muito elevado ou a empresa não dispunha de logística para o Brasil. Foram selecionados dois diferentes modelos de *beacons* fabricados pela empresa KST [46], o primeiro modelo Ion [47] e o segundo Particle [48], ambos possuem suporte ao protocolo Eddystone, porém somente o modelo Ion dispõe de pacotes do quadro TLM, isto devido a alguns sensores extras existentes no modelo e além disso, o também conta com um envoltório protetor de classificação IP65, tornando-o mais resistente contra avarias causadas pelo ambiente externo. Ambos os modelos contam com modificação do intervalo de anúncios, porém mesmo trocando os valores de 0,1s a 2s não notou-se mudança na taxa de leituras, esta ocorrendo a cada 2s aproximadamente.

Para o dispositivo móvel, elegeu-se um *smartphone* novo, garantindo as especificações quanto à versão de *Bluetooth*, suporte BLE, e acesso à última atualização da API dispo-

nível para a plataforma *Android*. O modelo Moto Z *Play* XT1635-02 [49], fabricado pela Lenovo, lançado no ano de 2016, ofereceu o devido suporte necessário ao desenvolvimento do projeto. Ressalta-se, contudo, não ter sido possível encontrar as informações quanto ao *chipset* utilizado no *hardware* e constatou-se durante a coleta de informações que a API do *Android* não fornece os requisitos necessários para modificar o valor de taxa de leitura do adaptador *Bluetooth*, mantendo-se na taxa de leitura já citada.

Em relação ao *software*, desenvolveu-se uma interface *Android* preliminar, de maneira a permitir a coleta de informações transmitidas pelo *beacon*, sendo, a mais importante, o RSSI, para tornar a estimativa de distância concebível. Além dessa funcionalidade, o aplicativo permitiu a simulação de um dispositivo anunciante para testes, onde o *beacon* não estivesse disponível. Os dados foram gravados em formato *Comma Separated Values* (CSV) para serem analisados posteriormente.

A Figura 4.1 ilustra o fluxograma do projeto preliminar. O *beacon* envia uma mensagem encapsulada com o protocolo Eddystone por meio do sinal BLE, capturado pelo dispositivo móvel por meio da interface *Android*. Passando pelos seguintes módulos e finalizando com a atualização da interface e repetindo-se o ciclo:

- *Parser*

A informação passa por um *parser* onde aplica-se uma máscara de *bits*, para a extração das informações necessárias, a *Identification* e a *Namespace* explicadas no Tópico 2.3.2. Já, o RSSI e o *Power* possuem métodos para sua obtenção por meio da API *Android*.

- Filtragem

O *RSSI* passa por filtros, explicados no Tópico 3.5, com o intuito de reduzir o ruído do sinal.

- Estimativa de distância

A partir da filtragem, calcula-se a distância com base no modelo RPLM, pois mostrou-se melhor em relação aos demais, como abordado no Tópico 3.4.

- Posicionamento

Passa-se por um nó condicional, onde, verifica-se a existência de pelo menos 3 *beacons* ativos, em caso positivo realiza-se a trilateração.

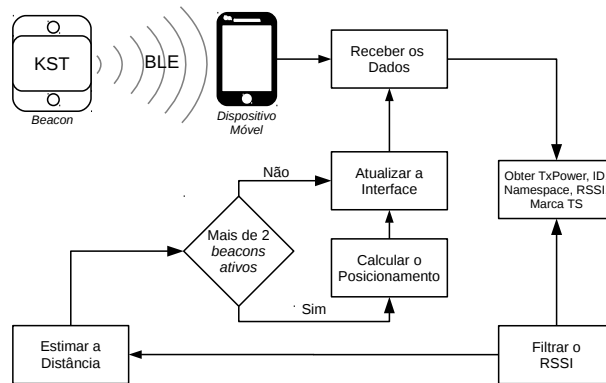
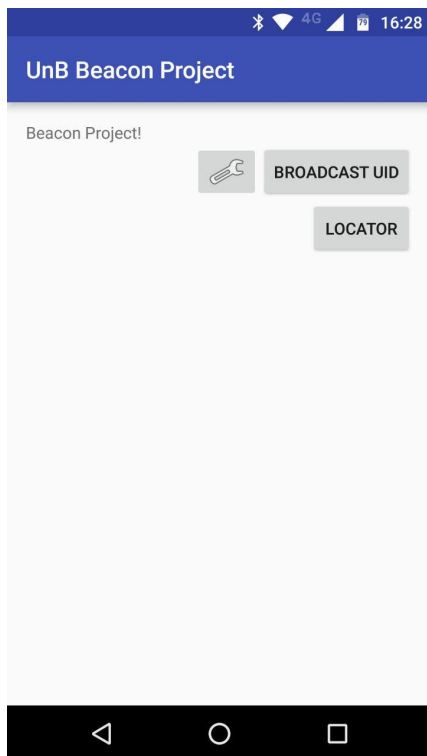


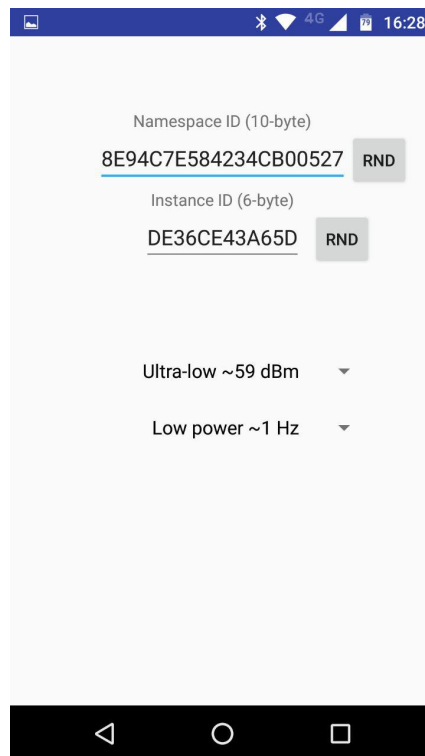
Figura 4.1: Fluxograma da interface preliminar

Algumas imagens do aplicativo podem ser vistas nas Figuras 4.2(a), 4.2(b), 4.2(c) e 4.2(d). A composição da interface se ocorre por:

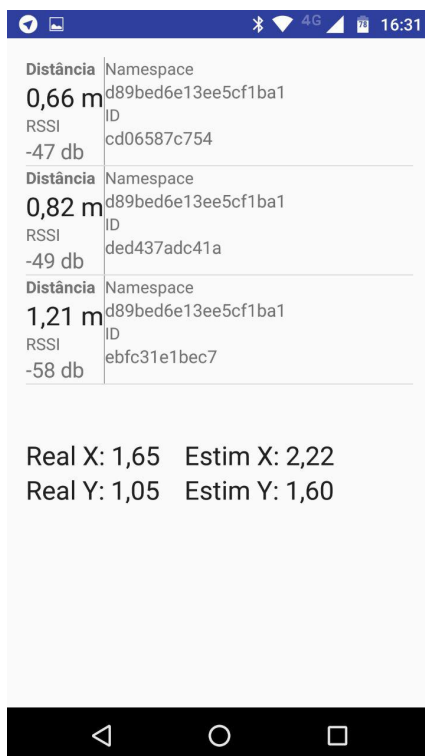
- Tela Inicial
Estrutura com botões para configuração da transmissão, para dar início a mesma e para ativar localização dos *beacons* em atividade.
- Tela de Configuração do Transmissor
Possui áreas para configurar *Namespace* e *Identification* de acordo com o protocolo Eddystone, abordados na Seção 2.3.2, e também para a frequência e potência do sinal, a partir de configurações pré-determinadas pela API *Android*.
- Tela de Localização
Exibe a lista de *beacons* em atividade e também algumas informações referentes a eles, como distância estimada, valor de RSSI, *Namespace* e *Identification*.
- Tela de Detalhamento do *Beacon*
Revela as informações do *beacon* de maneira mais detalhada, além das informações contidas na tela anterior. Disponibiliza um gráfico em tempo real referente às flutuações de RSSI. Por fim, o botão “editar” permite modificação na posição e no raio para o receptor. Os campos de posição devem ser utilizados para obter uma estimativa de posicionamento, já o campo “Raio” serve apenas para rodar o algoritmo de trilateração para a posição real, pois o raio aferido ocorre pela aproximação da distância.



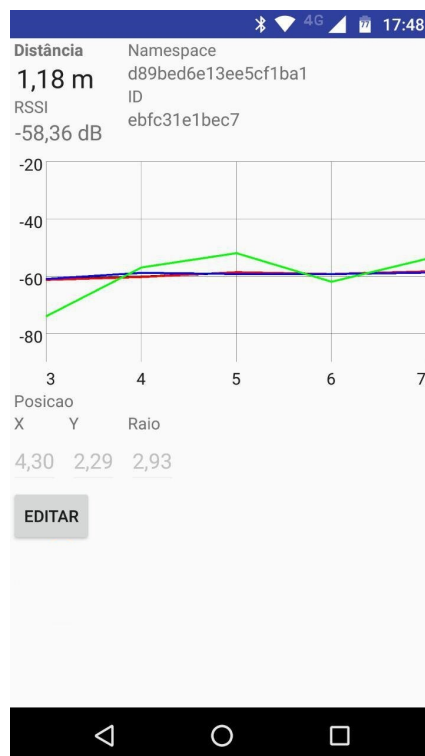
(a) Tela inicial



(b) Tela de configuração do transmissor



(c) Tela de localização



(d) Tela de detalhamento do beacon

Figura 4.2: Telas da interface preliminar

Os testes iniciais foram projetados com base nos problemas experienciados nos fundamentos e nos artigos abordados nas seções anteriores. Em todo os experimentos das Seções 4.2 a 4.5 seguiu-se um padrão, sendo realizadas ao menos 25 leituras, e os que utilizam distância, começa-se em 1m e termina-se em 6m. As variações ao padrão serão explicadas nas seções referentes a cada experimento, se necessárias. Os testes preliminares compreenderam:

- Flutuação da Intensidade do Sinal

Como abordado na Seção 3.5, a intensidade do sinal sofre variação conforme o ambiente em que se encontra.

- Análise de Filtros

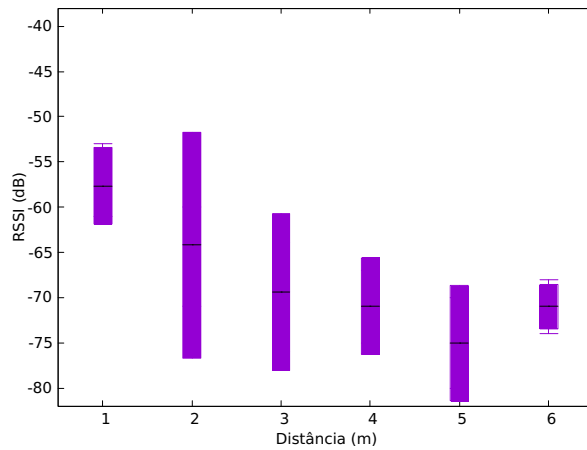
Compara-se os filtros abordados na Seção 3.5 e os modifica conforme a necessidade do estudo.

- Ambientes de Testes

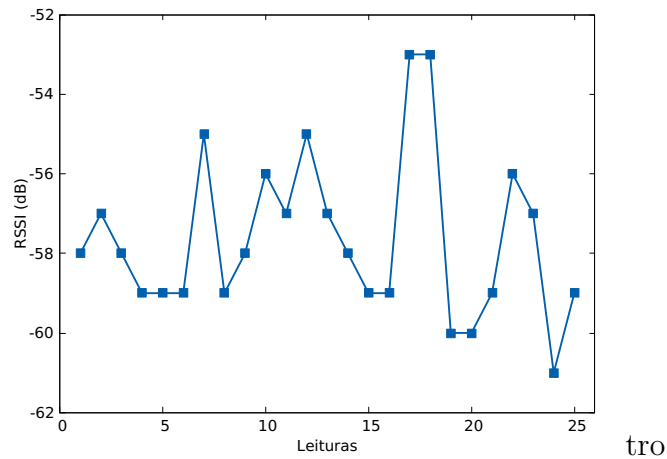
Descreve-se os diferentes cenários em que os experimentos foram realizados.

4.2 Flutuação da Intensidade do Sinal

O indicador de potência de sinal, RSSI, apresentou grande variação ao longo do tempo, mesmo em ambiente estático, sem movimento da fonte ou do receptor. Para um *beacon* colocado a um metro e aumentando a distância a cada metro do receptor, sendo realizadas 25 leituras por metro, pôde-se observar variância média de 7,43dB para o Ion. A Figura 4.3(a) ilustra somente o comportamento para o *beacon* Ion, a marca central na barra do gráfico representa a média dos valores, as marcas das barras simbolizam os valores mínimo e máximo de RSSI. Já, a barra simboliza a área onde a densidade de leituras pode flutuar. Na Figura 4.3(b) observa-se a oscilação entre as leituras, o que evidencia a necessidade de técnicas de filtragem.



(a) Variância do RSSI presente quando não há filtros



(b) Flutuação de RSSI ao longo de leituras sem filtro - Ion

Figura 4.3: Testes de flutuação do sinal de potência

4.3 Análise de Filtros

O valor de RSSI será utilizado como forma de estimativa de distância na Seção 4.5.3, portanto, a filtragem torna-se necessária devido a ruídos no sinal, causados por efeitos físicos como, reflexo do sinal, multipercurso, sombreamento, desvanecimento da intensidade entre outros explicados no Tópico 3.3 e 3.4. Para efeito de comparação entre o desempenho das filtragens, decidiu-se utilizar três filtros, selecionados com base nos artigos abordados no Tópico 3.7. O primeiro, uma simples média móvel, para se avaliar o resultado quase sem custo computacional. O segundo, filtro de Kalman, utilizado por diversos estudos no que tange filtragem de RSSI [2], [17]. O terceiro, o Filtro Duplo, proposto por Viswanathan *et al.* [38], visa minimizar ainda mais o erro médio em relação aos seus concorrentes.

- **Média Móvel**

Definido o tamanho de janela tornou-se possível obter a média de acordo com as leituras, o tamanho adotado foi de 25, pois necessita-se de ao menos 20 para garantir confiabilidade a informação.

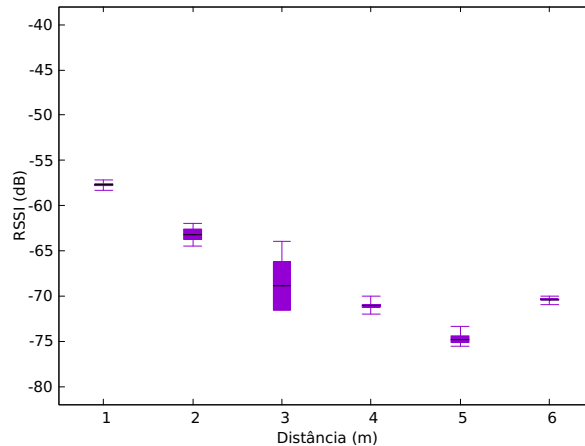


Figura 4.4: Variância do RSSI presente na Média Móvel

Nota-se diminuição da variância na Figura 4.4, porém a convergência para novos valores se torna mais lenta, conseqüentemente necessitando de mais leituras, assunto melhor abordado no Tópico 4.5.2.

- **Kalman**

Filtro de Kalman, já bem estabelecido como boa técnica de filtragem RSSI. Adaptou-se sua equação geral com o intuito de simplificar o seu funcionamento, pois não foi necessário acomodar eventos de controle.

Definido o filtro, seguiu-se a ideia de *Wouter Bulten* [42], que desconsidera a variável de controle u e utiliza A como matriz identidade, tornando a Equação (3.10) mais simples:

$$x_t = A_t x_{t-1} + B_t u_t + E_t \approx x_{t-1} + E_t \quad (4.1)$$

Ainda se tratando do segundo filtro, fez-se necessária a definição de como um estado X resulta em uma medição z . Assim utilizou-se o modelo geral:

$$Z_t = C_t x_t + \delta_t \quad (4.2)$$

Onde C representa a matriz de transformação e δ o ruído.

A etapa de atualização do filtro ocorre pela predição, da seguinte forma:

$$\bar{\mu}_t = \mu_{t-1} \quad (4.3)$$

$$\bar{\Sigma}_t = \Sigma_{t-1} + R_t \quad (4.4)$$

Note que μ denota a predição e X o estado real, já Σ define a certeza da predição e R o ruído causado pelo sistema, pode-se assumir que devido ao ambiente estático o ruído esperado procede, quase exclusivamente, do ambiente. Em ambiente dinâmico, onde há o movimento do receptor, deve-se aumentar o valor desta variável.

A partir da etapa de predição computa-se o ganho de Kalman:

$$K_t = \bar{\Sigma}_t(\bar{\Sigma}_t Q_t)^{-1} \quad (4.5)$$

A influência Q representa o ruído da informação. Para modelagem sobre RSSI, a variância simboliza o valor de influência. Assim, o ganho juntamente a Q servem como peso para avaliar a certeza da predição contra a certeza da medição. O que torna a etapa de atualização completa por:

$$\mu_t = \bar{\mu}_t + K_t(z_t - \bar{\mu}_t) \quad (4.6)$$

$$\Sigma_t = \bar{\Sigma}_t - (K_t \bar{\Sigma}_t) \quad (4.7)$$

Desta forma, quanto maior o ganho K mais a medição se torna integrada ao sistema, e quanto menor este ganho mais a predição se torna integrada.

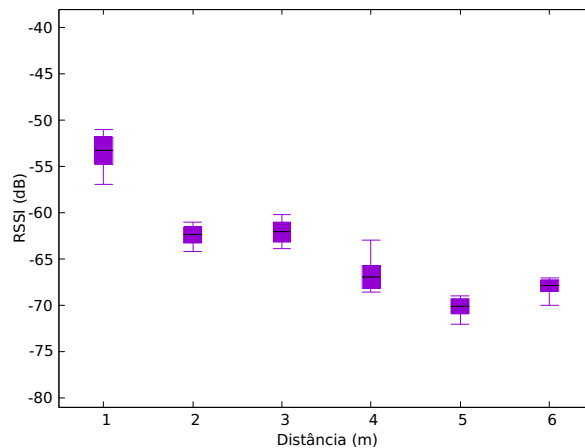


Figura 4.5: Variância RSSI presente no filtro de Kalman

Na Figura 4.5, semelhante a Figura 4.3(a), nota-se menor área de densidade de leituras, o que resulta em uma melhor precisão no recebimento de informação referente ao RSSI e por consequência na estimativa de distância como pode ser visto na Subseção 4.5.3.

- **Filtro Duplo**

Implementou-se uma versão do filtro duplo [38], porém os resultados não se mostraram interessantes em ambientes dinâmicos quando comparados com ambientes estáticos, pois o filtro em questão depende dos primeiros valores para convergir a predição, além disso se mostra resistente a variações bruscas no sinal, raramente divergindo de sua predição inicial. Na Figura 4.6 demonstra-se este comportamento quanto a pequena divergência de valores.

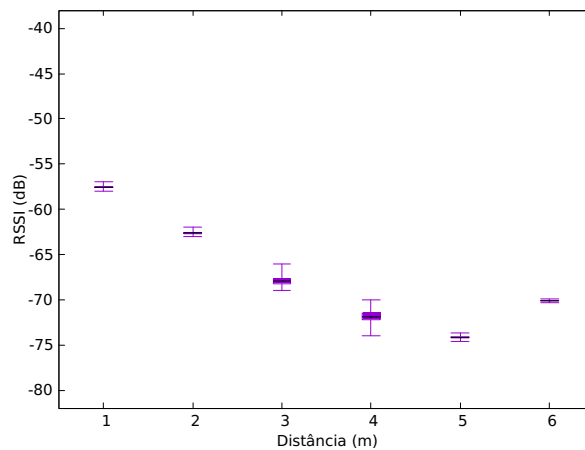


Figura 4.6: Variância RSSI presente no Filtro Duplo

Os resultados dos três filtros podem ser constatados na Figura 4.7 e nota-se grande vantagem em relação à variação não filtrada replicada na Figura 4.3(b). A média móvel se apresenta melhor que o filtro de Kalman, porém manifesta um problema semelhante ao filtro Duplo, no qual se faz necessário maior número de leituras para convergir a um novo valor de RSSI, piorando a precisão em ambientes dinâmicos, ou seja, em que ocorre a movimentação do receptor.

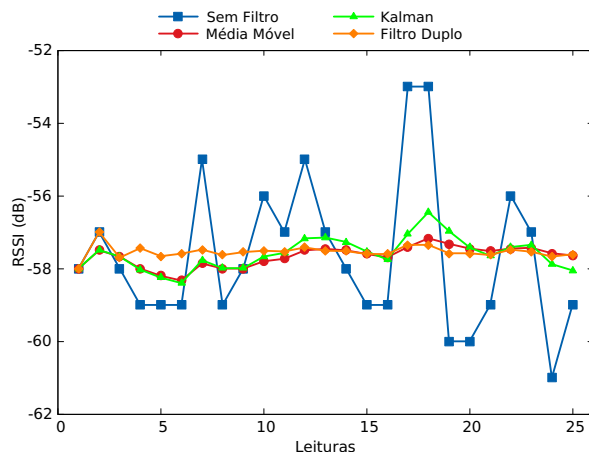


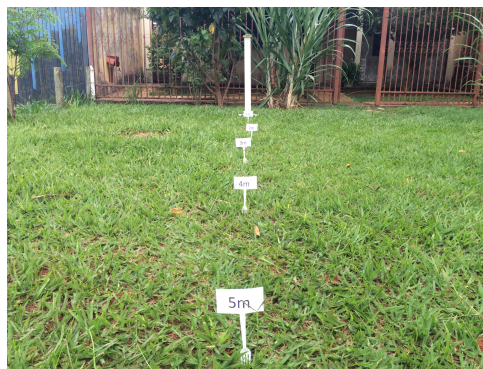
Figura 4.7: RSSI ao longo do tempo observado a 1m de distância

4.4 Ambientes de Teste

Selecionou-se diversos ambientes para cada tipo de teste, primeiramente uma área com bom campo de visão entre os dispositivos para testes de desvanecimento de intensidade e estimativa preliminar de distância, a área em questão situa-se logo atrás do Departamento de Ciência da Computação da Universidade de Brasília (UnB) e pode ser observada na Figura 4.8(a). Outra área com mesmo campo de visão também foi utilizada em frente a uma casa como mostra a Figura 4.8(b).



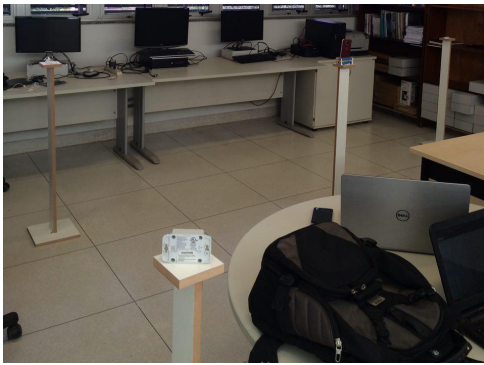
(a) CIC



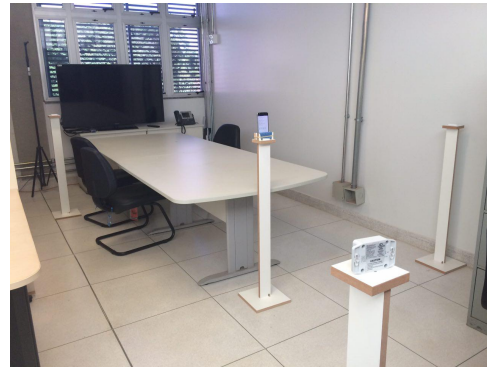
(b) Casa

Figura 4.8: Experimento de desvanecimento de RSSI e estimativa de distância

Já para os testes de posicionamento, a partir de trilateração, elegeu-se áreas internas, devido ao objetivo de se obter uma espécie de localizador interno de maneira similar a um GPS. Para tal, utilizou-se do pátio do Departamento de Computação assim como de um de seus laboratórios, o *Computer Network Laboratory* (COMNET), exibidos nas Figuras 4.9(a), 4.9(b), 4.9(c) e 4.9(d).



(a) Ambiente COMNET 1



(b) Ambiente COMNET 2



(c) Ambiente departamento 1



(d) Ambiente departamento 2

Figura 4.9: Experimentos de posicionamento

4.5 Resultados

Nessa seção apresentam-se os resultados obtidos através dos fundamentos e conhecimentos discutidos nos tópicos anteriores e possuem como base o seguinte conjunto de experimentos:

- Calibração dos Modelos de Propagação
Necessitou-se modificar os parâmetros de alguns modelos devido as diferenças de *hardware* existentes entre os modelos de *beacons* garantindo comportamento similar em ambos.
- Calibração da Convergência de RSSI
Com a necessidade de se reproduzir os experimentos em ambientes dinâmicos, notou-se que a quantidade de leituras para convergência do RSSI encontrava-se demasiadamente alta, e portanto, necessitou-se de ajustes nos parâmetros da filtragem, objetivando-se reduzir o número de leituras para convergência.

- Estimativa de Distância

Com a calibração e a filtragem do RSSI, pôde-se estimar a distância de maneira confiável e para tal utilizou-se a Equação (3.7).

- Posicionamento

Devido ao mesmo motivo da estimativa, garantir maior confiabilidade, pôde-se estimar a posição do receptor utilizando-se trilateração, abordado na Seção 3.6.

4.5.1 Calibração dos Modelos de Propagação

Para a calibração do modelo LDPLM e RPLM necessitou-se encontrar o valor de $RSSI_0$, que representa o RSSI para a distância de 1m entre transmissor e receptor, para tal realizou-se leituras e obteve-se a média, sendo esta de $-57,70\text{dB}$ no modelo Ion e $-53,20\text{dB}$ no modelo Particle. Já o valor X utilizou-se a média das variâncias para o filtro de Kalman, pelos motivos expostos na Seção 4.3, obtendo-se 1,22. Por fim, para o coeficiente de perda do sinal n , obteve-se o valor de 2,55 empiricamente, por meio de experimentos realizados nos ambientes relatados na Seção 4.4. O cálculo da distância para estes parâmetros encontram-se na Tabela 4.1, como esperado, o modelo de Friis não se comporta bem para o BLE e os outros modelos o superam na precisão, devido aos parâmetros de controle.

Tabela 4.1: Comparação entre os modelos *Frii's Transmission Model*, LDPLM e RPLM

RSSI (dB)	Friis (m)	LDPLM (m)	RPLM (m)	Distância Real (m)
-57,64	0,42	0,99	1,11	1,00
-64,20	0,77	1,80	2,00	2,00
-69,40	1,23	2,88	3,21	3,00
-70,96	1,42	3,31	3,69	4,00
-75,04	2,05	4,79	5,33	5,00
Erro Médio (m)	1,82	0,24	0,19	0,00

4.5.2 Calibração da Convergência de RSSI

A partir de um *script Python* gerou-se uma tabela por meio da variação dos valores Q e R do filtro de Kalman. O resumo dessa tabela encontra-se exposto na Tabela 4.2. Modificando estes valores iniciais pode-se chegar à convergência rapidamente, porém isto causa efeitos colaterais, novamente, na variância como consta na Figura 4.10. Portanto, optou-se por utilizar uma convergência em equilíbrio com o seu efeito colateral, para que fosse possível obter um novo valor de convergência, mas sem afetar de forma desfavorável

a estimativa de distância, sendo estes valores $Q = 1,5$ e $R = 0,7$, como expressado na Figura 4.11, este obtém convergência por volta de 15 leituras, contra mais de 25 do modelo inicial.

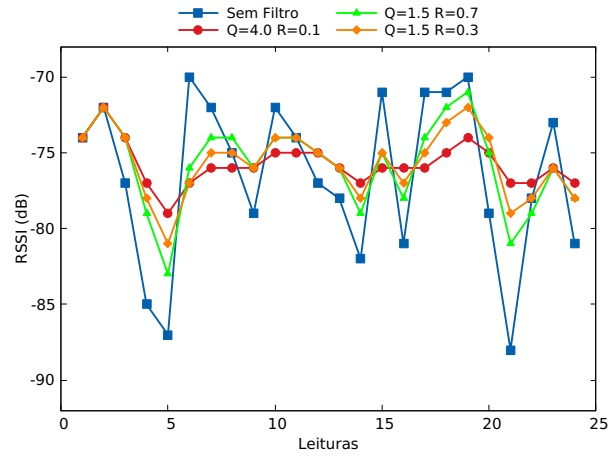


Figura 4.10: Variância no filtro Kalman devido a mudança de valores iniciais

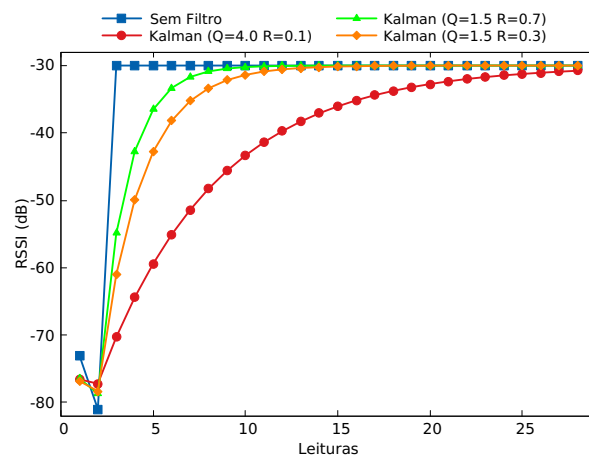


Figura 4.11: Quantidade de leituras para convergência no filtro Kalman

Os valores marcados na Tabela 4.2 simbolizam o ponto onde a convergência completou-se, atingindo a diferença aproximada de duas casas decimais para o valor esperado, no exemplo $-30,00$.

Tabela 4.2: Valores iniciais e leituras necessários para convergência do filtro de Kalman

Medida	Sem Filtro	Q= 4,0 R= 0,1	Q=1,5 R= 0,7	Q= 1,5 R= 0,3
1	-30,00	-70,31	-54,89	-61,01
2	-30,00	-64,42	-42,73	-49,90
3	-30,00	-59,39	-36,51	-42,77
4	-30,00	-55,09	-33,33	-38,20
5	-30,00	-51,43	-31,70	-35,26
6	-30,00	-48,29	-30,87	-33,38
7	-30,00	-45,62	-30,45	-32,17
8	-30,00	-43,34	-30,23	-31,39
9	-30,00	-41,39	-30,12	-30,89
10	-30,00	-39,73	-30,06	-30,57
11	-30,00	-38,30	-30,03	-30,37
12	-30,00	-37,09	-30,02	-30,24
13	-30,00	-36,06	-30,01	-30,15
14	-30,00	-35,17	-30,00	-30,10
15	-30,00	-34,41	-30,00	-30,06
16	-30,00	-33,77	-30,00	-30,04
17	-30,00	-33,22	-30,00	-30,03
18	-30,00	-32,75	-30,00	-30,02
19	-30,00	-32,35	-30,00	-30,01
20	-30,00	-32,00	-30,00	-30,01
21	-30,00	-31,71	-30,00	-30,00
22	-30,00	-31,46	-30,00	-30,00
23	-30,00	-31,25	-30,00	-30,00
24	-30,00	-31,07	-30,00	-30,00
25	-30,00	-30,91	-30,00	-30,00

4.5.3 Estimativa de Distância

A partir dos resultados com filtragem sobre o valor do RSSI, pôde-se estimar a distância de maneira mais confiável devido a baixa flutuação de valores. Para tal, utilizou-se a estimação pelo método de *Rappaport* de desvanecimento do sinal a partir do isolamento da variável de distância como já explicado previamente no capítulo 3.

Os testes foram realizados de 1 em 1 metro até o limite de 6 metros, isto devido a baixa diferença de potência de sinal após 5 metros de distância, portanto os valores para 6 metros em diante não se revelaram tão úteis quanto valores anteriores a esta marca,

como demonstrado na Figura 4.12. O motivo pelo qual isto ocorre se dá pela natureza dos dispositivos transmissores e receptores, que para manter um consumo energético menor, o seu *hardware*, no caso, as antenas, possuem menor capacidade de operação, não distinguindo eficientemente as variações que ocorrem após os 5m. Pode-se observar na Figura 4.12 um padrão, onde há um desvanecimento da intensidade do sinal a medida que se aumenta a distância, o que não ocorre do quinto metro em diante, isso se deve a pequena discrepância do RSSI para distância maiores que o limite encontrado de 5m, o que aumenta o erro da estimativa.

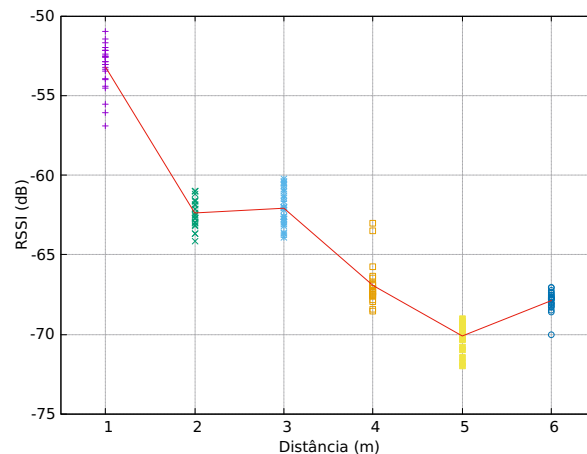


Figura 4.12: Distância máxima devido ao desvanecimento do sinal

Na Tabela 4.3 e Figura 4.14 registram-se os valores reais e estimados para as distâncias testadas. O melhor erro médio foi de 0,12m para o Filtro Duplo no modelo Ion de *beacon*, valor aceitável quando comparado a erros encontrados na literatura, outros valores podem ser consultados na mesma tabela citada. Porém, o Filtro Duplo, assim como a Média Móvel possuem alta rigidez quanto a convergência para novos valores como já abordado no Tópico 4.3, ilustrado no Figura 4.13. A partir do erro de estimativa da distância calculou-se a taxa de melhora em relação ao teste Sem Filtro (SF), onde a Média Móvel obteve 18%, Kalman 11% e o Filtro Duplo 58%.

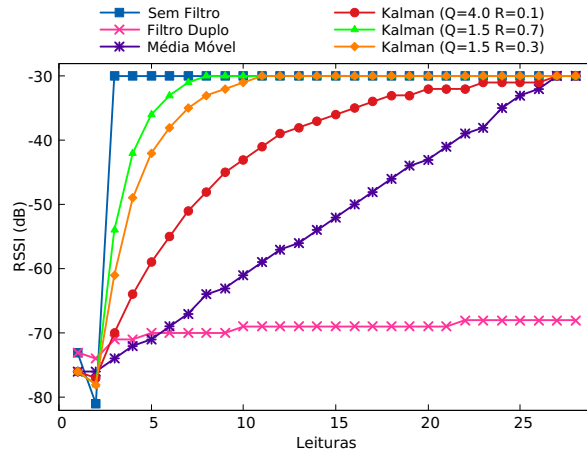


Figura 4.13: Quantidade de leituras para convergências - Média Móvel e Filtro Duplo

Tabela 4.3: Estimativa de distância - Ion

Distância Real (m)	Sem Filtro	Média Móvel	Kalman	Filtro Duplo
1,00	1,11	1,11	1,10	1,10
2,00	2,00	1,83	1,90	1,74
3,00	3,21	3,06	3,08	2,81
4,00	3,69	3,73	3,70	4,00
5,00	5,33	5,21	5,30	4,92
Erro Médio (m)	0,19	0,16	0,17	0,12

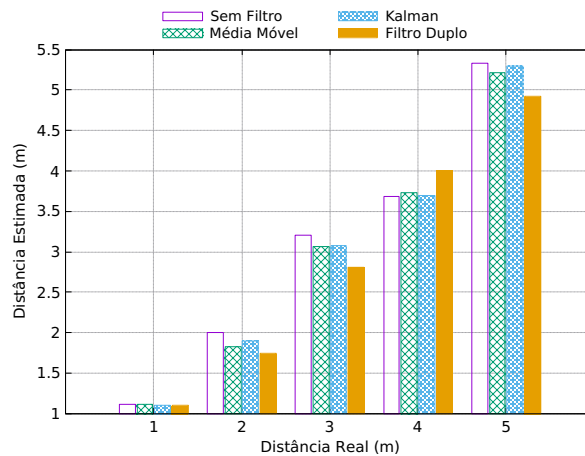


Figura 4.14: Comparação de estimativa de distância - Ion

Considerando somente a Tabela 4.3, pode-se inferir que o valor sem filtragem seja apropriado, porém, vale lembrar que os valores apresentados passaram por uma média

aritmética devido a variação de seus índices, o que faz com que valores sem filtragem se comportem de maneira similar a Média Móvel, a variação se torna evidente na Tabela 4.4 e na Figura 4.3(b), obteve-se média da variância de 7,43db num cenário sem filtragem.

Tabela 4.4: Variação e desvio padrão para cada metro e filtragem

Distância (m)	Ion							
	Variância				Desvio Padrão			
	SF	MM	K	DF	SF	MM	K	DF
1,00	4,24	0,08	0,20	0,03	2,06	0,29	0,44	0,17
2,00	12,50	0,57	1,96	0,07	3,54	0,75	1,40	0,26
3,00	8,67	2,67	3,04	0,25	2,94	1,63	1,74	0,42
4,00	5,37	0,15	0,29	0,38	2,32	0,39	0,54	0,62
5,00	6,37	0,34	0,61	0,05	2,52	0,58	0,78	0,22
Média	7,43	0,76	1,22	0,15	2,68	0,73	0,98	0,34

4.5.4 Posicionamento

Com os valores de distância delimitados e com uma estimativa boa tornou-se possível prever agora a posição do receptor por meio da técnica de trilateração, utilizando-se, portanto, três *beacons*. Para averiguar o algoritmo de posicionamento aplicado mediu-se, a princípio, as posições onde seria colocado o receptor, a fim de verificar a posição real do dispositivo, em seguida, utilizou-se o algoritmo baseado nas coordenadas dos três transmissores juntamente com a estimativa do raio entre os transmissores e o receptor. Devido ao tempo necessário para que a estimativa convirja em novas posições, o teste foi executado de maneira estática, deixando, portanto, o dispositivo imóvel entre um a dois minutos e após esse tempo gravou-se os dados em tabela CSV.

A Tabela 4.5 expõe os valores encontrados para posicionamento em quatro locais diferentes com um erro em torno de 0,39m, o que pode ser considerado bom devido a este valor corresponder a pouco mais de um passo de distância do receptor. A Figura 4.15 evidencia a tabela em forma de gráfico. Calculou-se o erro médio com base na distância Euclidiana entre os pontos, a partir da equação abaixo:

$$E = \frac{1}{N} * \sum \sqrt{t(x_1 - x)^2 + (y_1 - y)^2} \quad (4.8)$$

Onde N representa a quantidade de coordenadas, x , x_1 a posição real e estimada no eixo X, respectivamente, de maneira equivalente para as posições no eixo Y indicados por y e y_1 . Por fim, E caracteriza o erro médio.

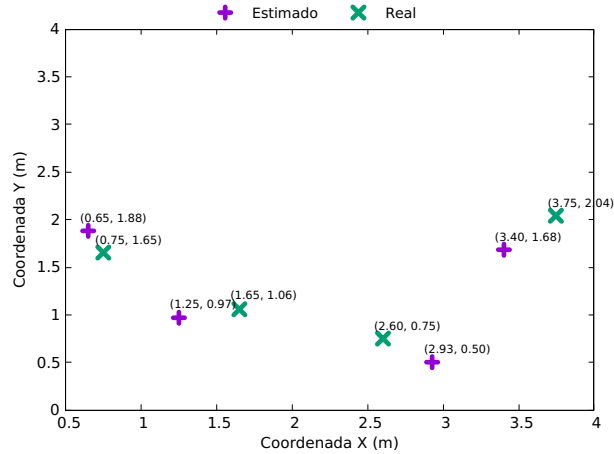


Figura 4.15: Trilateração real e estimada

Tabela 4.5: Posição real e estimada

Real (m)		Estimada (m)		Erro (m)
X	Y	X	Y	
0,75	1,65	0,65	1,88	0,25
2,60	0,75	2,93	0,50	0,41
3,75	2,04	3,40	1,68	0,50
1,65	1,05	1,25	0,97	0,40
Erro Médio (m)				0,39

4.6 Prova de Conceito

Com o intuito de exemplificar uma funcionalidade para os conceitos discutidos no estudo, desenvolveu-se uma prova de conceito, em um cenário onde transmite-se uma informação para o usuário com base na sua proximidade a um objeto físico, no contexto de um estabelecimento *food truck*. A PW promove um novo contexto, por meio de objetos físicos que anunciam dados de interesse à localidade ou preferência pessoal, fazendo com que o usuário não mais precise se dar o trabalho de buscar a informação. Dentre às inúmeras funcionalidades da PW, exemplifica-se um aplicativo para transmitir o cardápio de um *food truck*, produzido no estudo.

A partir da versão preliminar, desenvolvida para a coleta de informações, gerou-se uma nova interface com o intuito de conferir usabilidade ao usuário. As funções existentes foram mantidas por questões de *debugging*, e novas funcionalidades implementadas para proporcionar melhor entendimento.

A Figura 4.16 representa o fluxograma da interface de caso de uso. O funcionamento permanece idêntico à variação anterior, exceto pelos módulos que constam no interior do retângulo, cuja mudança se encontra a seguir: após a extração das informações do *beacon*, um nó condicional verifica se a mensagem recebida pertence a um dispositivo cadastrado, em caso positivo, a interface se conecta a um banco de dados, através de *Wi-Fi* ou redes móveis, para obter informações extras relativas ao *beacon*, sendo essas as informações relacionadas ao local do dispositivo transmissor, tais como, cardápio de um *food truck* ou promoções de uma loja.

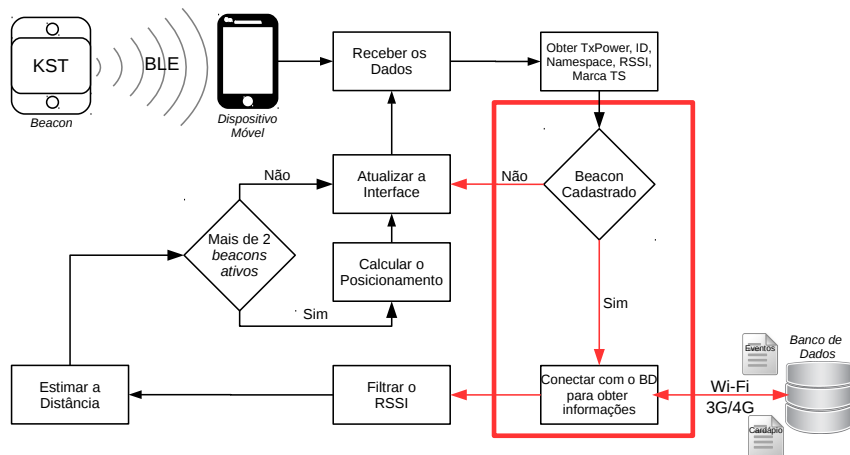


Figura 4.16: Fluxograma para interface caso de uso

Na interface caso de uso, confirmou-se que quando o usuário se aproxima suficientemente de um dispositivo *beacon*, ele pode receber uma lista com produtos ou informações de locais próximos, referentes ao posicionamento do dispositivo. Por exemplo, no caso de um *food truck*, o usuário poderá acessar o cardápio do estabelecimento, ou para outros casos, obter informações referentes a oficinas, eventos ou exposições que acontecerão em algum Centro Empresarial. O exemplo do *food truck* foi escolhido e implementado utilizando *firebase*, banco de dados não relacional em nuvem, oferecido gratuitamente pela *Google*, assim como a ferramenta *Glide* para a parte de gerenciamento de imagens, o fluxograma pode ser visto na Figura 4.16. Um exemplo de informações recebidas de um *beacon* pode ser visto na Figura 4.17.

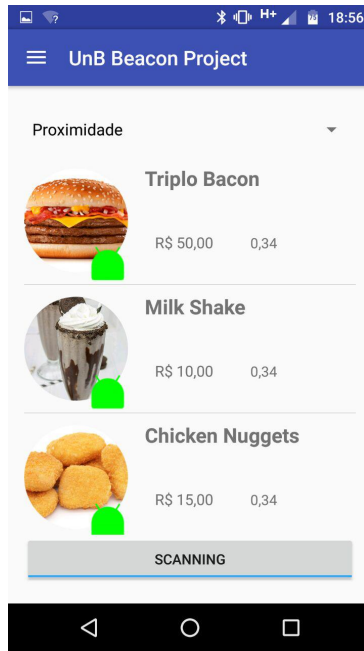


Figura 4.17: Tela da aplicação *food truck*

Para avaliar o desempenho da interface em caso real, foi necessária a checagem dos eventos enviados corretamente para o aplicativo, a partir da aproximação do usuário de um transmissor. No teste, utilizou-se de contato visual para identificação de cada um dos *beacons*, que se encontravam simbolizados pela logotipo *Android* pintada em quatro cores diferentes, uma para cada *beacon*, como pode ser visto na Figura 4.17, no caso apenas um dos *beacons* estava presente.

A certificação do funcionamento ocorreu com análise de todas as possíveis combinações de distância de 0,5 em 0,5m até a faixa limite de 5 metros. Os resultados podem ser observados na Tabela 4.6, onde marca-se 1 em caso de recebimento e 0 caso contrário.

Tabela 4.6: Checagem de desencadeamento de eventos

Distância (m)	0,50	1,00	1,50	2,00	2,50	3,00	3,50	4,00	4,50	5,00
0,50	1	0	0	0	0	0	0	0	0	0
1,00	1	1	0	0	0	0	0	0	0	0
1,50	1	1	1	0	0	0	0	0	0	0
2,00	1	1	1	1	0	0	0	0	0	0
2,50	1	1	1	1	1	0	0	0	0	0
3,00	1	1	1	1	1	1	0	0	0	0
3,50	1	1	1	1	1	1	1	0	0	0
4,00	1	1	1	1	1	1	1	1	0	0
4,50	1	1	1	1	1	1	1	1	1	0
5,00	1	1	1	1	1	1	1	1	1	1

Capítulo 5

Conclusão

A *Physical Web* objetiva uma interação do usuário e de locais em sua proximidade com maior fluidez, por meio de objetos físicos denominados *beacons*, por exemplo, um usuário em um parque gostaria de informações sobre a programação de eventos para o local, sem o trabalho de pesquisar por estes. Portanto, necessita-se dos mecanismos de localização, e além disso de uma tecnologia de baixo custo energético, como o BLE, para viabilizar sua utilização de forma comercial.

Sistemas baseados em GPS, embora amplamente disponíveis, possuem dificuldade em identificar e posicionar receptores em ambientes internos, com boa acurácia. Tendo como objetivo explorar outras tecnologias, por exemplo, *Bluetooth*, mostrando a capacidade destas em se sobressair no contexto de ambientes fechados em relação ao GPS. Para tal desenvolveu-se uma interface preliminar de forma a coletar as informações necessárias para avaliar a estimativa de distância. Testes iniciais denotam grande variação dos valores do RSSI obtido, impactando significativamente na obtenção de uma estimativa de distância e tornou-se necessário utilizar técnicas de filtragem a fim de minimizar este problema. O filtro que melhor respondeu às necessidades foi o de Kalman, devido a capacidade de ajustes tanto para situações estáticas quanto dinâmicas (sem e com a movimentação do usuário), a partir de variáveis Q e R , onde Q representa a influência, ou seja o ruído da informação, e R o ruído causado pelo sistema, responsáveis por regular a filtragem dando mais liberdade para a flutuação ou restringindo-a.

Encontrou-se também, experimentalmente, um limite de estimativa de distância para o *beacon* utilizado, dado em torno dos 5 metros, isto devido à pequena divergência de RSSI, prejudicando a estimativa, desta forma há um limite para o desvanecimento da intensidade do sinal ao se aumentar a distância do receptor ou transmissor. Por causa deste limite e também do quantitativo disponível de dispositivos transmissores não foi possível testar-se amplamente o posicionamento por trilateração, porém ainda assim obteve-se um erro aproximado de 0,39m, o que corresponde a pouco mais de um passo da posição

real, valor aceitável para ambientes internos. Todavia, para aplicações relacionados a apenas um único dispositivo transmissor, como encontrar objetos perdidos, receber informações baseado na proximidade ao transmissor, enviar listas de objetos disponíveis em uma estante, prateleira ou informações quanto a telemetria, como temperatura, estado de funcionamento e bateria, a tecnologia *Bluetooth* se mostra mais que capaz. Com os resultados aqui apresentados, obtendo-se erro médio de 0,17m para a estimativa de distância, argumenta-se ser a intensidade do sinal uma boa opção para localização, e com devido quantitativo de dispositivos, posicionamento para ambientes internos, com acurácia relativamente alta, condizendo, pois, com outros estudos já desenvolvidos nesse cenário.

Trabalhos Futuros

Sugere-se a utilização de maior quantitativo de dispositivos no ambiente, a fim de se comprovar melhor precisão no posicionamento dado o limite de 5m, e de técnicas de minimização de erro como *Least Squares* durante a trilateração, assim como a utilização de multi-lateração, técnica que utiliza mais de 3 dispositivos como base para a estimativa de posicionamento. Além disso, poder-se-ia desenvolver, de fato, uma aplicação *Bluetooth Positioning System* (BPS) que possa guiar o usuário a locais de seu interesse, porém torna-se necessário para tal boa quantidade de dispositivos *beacons* espalhados estrategicamente na área a ser mapeada.

Referências

- [1] Want, Roy, Bill N. Schilit, e Scott Jenson: *Enabling internet of things*. IEEE Computer Society, 2015. 1
- [2] Vargas, Milan Herrera: *Indoor navigation using bluetooth low energy beacons*. Turkey University, 2016. xi, 1, 4, 27, 29, 35, 36, 42
- [3] Namiot, Dmitry e Manfred Sneps-Sneppe: *The physical web in smart cities*. IEEE, 2015. 1
- [4] Kindberg, Tim, John Barton, Jeff Morgan, Gene Becker, Debbie Caswell, Philippe Debaty, Gita Gopal, Marcos Frid, Venky Krishnan, Howard Morris, John Schettino, Bill Serra e Mirjana Spasojevic: *People, places, things: Web presence for the real world*. IEEE Computer Society, 2000. 1, 2
- [5] Google: *Physical web*. <https://github.com/google/physical-web>, acessado em julho 2016. 2
- [6] C, Andrew, Huang, Benjamin C. Ling e Shankar Ponnekanti: *Pervasive computing: What is it good for?* International Workshop on Data Engineering for Wireless and Mobile Access, 1999. 2, 31
- [7] BKON: *Bkon*. <https://bkon.com/>, Acessado em março 2016. 2
- [8] Levine, Barry: *With the physical web, you become the search engine*. <http://marketingland.com/with-the-physical-web-you-become-the-search-engine-165338>, acessado em julho 2016, 2016. 2
- [9] Namiot, Dmitry: *Context-aware browsing – a practical approach*. IEEE, 2012. 2, 3
- [10] Evans, Katie: *Beacons go to school*. <https://www.mobilestrategies360.com/2016/01/14/beacons-go-school>, Acessado julho 2016, 2016. 2
- [11] Trnka, Tomas: *Eddystone beacon technology and the physical web*. <https://mobiforge.com/design-development/eddystone-beacon-technology-and-the-physical-web>, acessado em julho de 2016. 2, 3
- [12] Research, ABI: *New ble beacons coverage*. <https://www.abiresearch.com/pages/ble-beacons/>, Acessado em março 2016, 2016. 2
- [13] Google: *Eddystone format*. <https://developers.google.com/beacons/eddystone>, Acessado em março 2016, 2016. 3

- [14] Apple: *ibeacon*. <https://developer.apple.com/ibeacon/>, Acessado em março 2016. 3
- [15] Subhan, Fazli, Azat Rozyyev, Halabi Hasbullah e Sheikh Tahir Bakhsh: *Indoor positioning in bluetooth networks using fingerprinting and lateration approach*. IEEE, 2011. 4, 27, 33
- [16] Galván-Tejada, Carlos E., José C. Carrasco-Jiménez e Ramon F. Brenna: *Bluetooth-wifi based combined positioning algorithm, implementation and experimental evaluation*. The 2013 Iberoamerican Conference on Electronics Engineering and Computer Science, 2013. 4, 33
- [17] Dalce, Rejane, Thierry Val e Adrien Van Den Bossche: *Comparison of indoor localization systems based on wireless communications*. Wireless Engineering and Technology, 2011. 4, 24, 25, 27, 29, 31, 42
- [18] Kurose, James F. e Keith W. Ross: *Computer Networking A Top-Down Approach*. Pearson, 2013. 7, 9, 11
- [19] Tanenbaum e Wheterall: *Computer Networks*. Pearson, 2011. 7, 8, 13
- [20] Leon-Garcia, Alberto e Indra Widjaja: *Communication Networks: Fundamental Concepts and Key Architectures*. McGraw-Hill, 2000. 7
- [21] IEEE: *Ieee 802.15 working group for wireless specialty networks (wsn)*. <http://www.ieee802.org/15/>, 2017. 13
- [22] IEEE: *Ieee 802.15 wpanTM task group 4 (tg4)*. <http://www.ieee802.org/15/pub/TG4.html>. 13
- [23] IEEE: *Ieee 802.15 wpan task group 1 (tg1)*. <http://www.ieee802.org/15/pub/TG1.html>, 2017. 13
- [24] Group, Bluetooth Special Interest: *Bluetooth*. <https://www.bluetooth.com/>, 2017. 14
- [25] Bray, Jennifer e Charles F. Sturman: *Bluetooth : Connect Without Cables*. Pearson Education, 2001. 14
- [26] Lindh, Joakim: *Bluetooth low energy beacons application report*. Texas Instruments, 2016. 17
- [27] Gutierrez, José A.: *Ieee 802.15.4 enabling pervasive wireless sensor networks*. <https://people.eecs.berkeley.edu/~prabal/teaching/cs294-11-f05/slides/day21.pdf>, acessado em julho de 2016. 17
- [28] Liang, Chieh Jan Mike, Nissanka Bodhi Priyantha, Jie Liu e Andreas Terzis: *Surviving wi-fi interference in low power zigbee networks*. SenSys 2010 ACM, 2010. 17
- [29] Rivero, Armando: *Zigbee and bluetooth: Low energy and smartphone integration*. Universit'a della Svizzera Italiana ALaRI Institute, 2013. ix, 18

- [30] Apple: *Getting Started with iBeacon*. Apple, jun 2014. <https://developer.apple.com/ibeacon/Getting-Started-with-iBeacon.pdf>. 20
- [31] Shen, Guowei, Rudolf Zetik e Reiner S. Thomä: *Performance comparison of toa and tdoa based location estimation algorithms in los environment*. WPNC(Workshop on positioning, navigation and communication), 2008. 23
- [32] Kaune, Regina: *Accuracy studies for tdoa and toa localization*. Fraunhofer FKIE/University of Bonn, 2012. 23
- [33] Hughes, William J.: *Global positioning system (gps) standard positioning service (sps) performance analysis report*. Federal Aviation Administration, 2013. 23
- [34] Blank, Aaron: *The limitations and admissibility of using historical cellular site data to track the location of a cellular phone*. Richmond Journal of Law & Technology, 2011. 24
- [35] Albowicz, Joe, Alvin Chen e Lixia Zhang: *Recursive position estimation in sensor networks*. IEEE, 2001. 24
- [36] Sturdevant, Rick W.: *Societal Impact of Spaceflight : Chapter 17*. NASA, 2007. 24
- [37] Wang, Yapeng, Xu Yang, Yutian Zhao, Yue Liu e Laurie Cuthbert: *Bluetooth positioning using rssi and triangulation methods*. IEEE Consumer Communications and Networking Conference(CCNC), páginas 837–842, 2013. 24, 27, 30
- [38] Viswanathan, Subha e Sreedevi Srinivasan: *Improved path loss prediction model for short range indoor positioning using bluetooth low energy*. IEEE, 2015. 24, 30, 42, 45
- [39] Whitehouse, Kamin e David Culler: *Calibration as parameter estimation in sensor networks*. WSNA, 2002. 24
- [40] Nuevo, Jorge: *A Comprehensible GloMoSim Tutorial*. Université Québec, 2004. 28
- [41] Welch, Greg e Gary Bishop: *An introduction to the kalman filter*. University of North Carolina at Chapel Hill, 2001. 29
- [42] Bulten, Wouter: *Kalman filters explained: Removing noise from rssi signals*. <https://wouterbulten.nl/blog/tech/kalman-filters-explained-removing-noise-from-rssi-signals/>, Acessado em junho 2016, 2015. 29, 43
- [43] Davies, N., K. Cheverst, K. Mitchell e A. Friday: *‘caches in the air’: disseminating tourist information in the guide system*. WMCSA’99, 1999. 32
- [44] Deordica, Bianca e Marian Alexandru: *Advertisement using bluetooth low energy*. Review of the Air Force Academy, 2014. ix, 32, 33
- [45] Filippoupolitis, Avgoustinos, William Oliff e George Loukas: *Bluetooth low energy based occupancy detection for emergency management*. IEEE, 2016. xi, 34, 35
- [46] KST: *Kst*. <https://kstechnologies.com/>, Acessado em março 2016. 37

- [47] KS Technologies, Revision A: *Device Specification: Ion*, December 2014. <https://kstechnologies.com/wp-content/uploads/2015/04/Ion-Device-Specification-Rev-A.pdf>. 37
- [48] KS Technologies, Revision B: *Device Specification: Particle*, April 2015. <https://kstechnologies.com/wp-content/uploads/2015/04/Particle-Device-Specification-Rev-B.pdf>. 37
- [49] Lenovo: *Moto z play xt1635-02 specification*. http://www.gsmarena.com/motorola_moto_z_play-8310.php, Acessado em setembro 2016, 2016. 38