



PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

CAIO CÉSAR LIMA LACERDA FERREIRA

**SISTEMA DE LOCALIZAÇÃO REGIONALIZADO DE DISPOSITIVO DE ACESSO
A VEÍCULO AUTOMOTOR BASEADO EM *BLUETOOTH LOW ENERGY***



Universidade Federal de Pernambuco
pgraduacao@cin.ufpe.br
www.cin.ufpe.br/~posgraduacao

Recife

2018

CAIO CÉSAR LIMA LACERDA FERREIRA

**SISTEMA DE LOCALIZAÇÃO REGIONALIZADO DE DISPOSITIVO DE ACESSO
A VEÍCULO AUTOMOTOR BASEADO EM *BLUETOOTH LOW ENERGY***

Dissertação apresentada à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de concentração: Engenharia da Computação

Orientador: Abel Guilhermino da Silva Filho

Recife

2018

Catálogo na fonte
Bibliotecária Monick Raquel Silvestre da S. Portes, CRB4-1217

F383s Ferreira, Caio César Lima Lacerda
Sistema de localização regionalizado de dispositivo de acesso a veículo automotor baseado em *bluetooth low energy* / Caio César Lima Lacerda Ferreira. – 2018.
126 f.: il., fig., tab.

Orientador: Abel Guilhermino da Silva Filho.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2018.
Inclui referências.

1. Engenharia da computação. 2. Aprendizagem de máquina. I. Silva Filho, Abel Guilhermino (orientador). II. Título.

621.39 CDD (23. ed.) UFPE - CCEN 2020 - 116

CAIO CÉSAR LIMA LACERDA FERREIRA

**SISTEMA DE LOCALIZAÇÃO REGIONALIZADO DE DISPOSITIVO DE ACESSO
A VEÍCULO AUTOMOTOR BASEADO EM BLUETOOTH LOW ENERGY**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Aprovado em: 12/09/2018

BANCA EXAMINADORA

Prof. Dr. Carlos André Guimarães Ferraz
Centro de Informática /UFPE

Prof. Dr. Victor Wanderley Costa de Medeiros
Departamento de Estatística e Informática UFRPE

Prof. Dr. Abel Guilhermino da Silva Filho
Centro de Informática / UFPE
(Orientador)

AGRADECIMENTOS

Agradeço ao Prof. Dr. Abel Guilhermino da Silva Filho por ter me dado a oportunidade de realizar este trabalho, por ter me orientado de uma forma bastante paciente, demonstrando sempre entusiasmo a cada passo dado e me incentivando nos momentos de dificuldade.

Agradeço à minha família, meus pais pela dedicação e firmeza no cuidado da família, sempre sendo um porto seguro. Às minhas irmãs pela solidariedade, palavras de incentivo e refúgio nos momentos de dificuldade. Às minhas avós, tias, primas e primos pela presença marcante na minha vida.

Agradeço ao grupo do LIVE, Bianca, Maria Luiza, Maria, Eron, Camarotti, Hilson, Rafael, Marcus, Rodolfo, Victor, Henrique, Hélio, Ricardo, Prof. Carlos Ferraz, Prof. Marcílio, Prof. George Darmington pelo companheirismo e exemplo de profissionais que me inspiraram e me ajudaram nesta jornada.

Agradeço aos meus sócios, Maria e Hugo, e a toda a equipe Udon, em especial Regina, pela paciência e por levar a empresa com incrível competência na minha ausência durante esse período dividido entre as obrigações de empreender e de fazer pesquisa.

Obrigado a todos!

RESUMO

Com a redução de custo de fabricação de dispositivos eletrônicos, ampla disseminação de *smartphones* e dispositivos eletrônicos vestíveis (*smartwatches*, *smartbands*) e um mercado automotivo global competitivo, vários sistemas para prover comodidade e auxílio ao usuário têm surgido. Um desses sistemas, de entrada e partida passiva ao veículo, já está no mercado há pelo menos duas décadas, no entanto, não apresenta qualquer conectividade com o ambiente externo. Para suprir essa deficiência e aproveitando crescimento no mercado dos dispositivos *smart*, há uma tendência natural de convergir esses dispositivos ao ambiente automotivo aproveitando o surgimento e disseminação cada vez mais acentuada da tecnologia de conexão *Bluetooth Low Energy* (BLE). Este trabalho apresenta um sistema de localização de dispositivo BLE que faz o papel de chave veicular. Apresenta o *hardware* utilizado para construção do sistema, topologias de antenas posicionadas no veículo (*dongles* BLE), arquitetura e fluxo do sistema de localização, apresentando os algoritmos de aprendizagem de máquina utilizados, usando máquinas em duas camadas e em cascata. Apresenta as regiões de decisão escolhidas e apresenta uma discussão sobre o parâmetro de entrada do sistema, o RSSI. Mostra a não-linearidade desse parâmetro e seu impacto na qualidade dos resultados obtidos.

Palavras-chave: Sistemas de acesso passivo a veículo. *Bluetooth Low Energy*. *Fingerprinting*. RSSI. Aprendizagem de máquina.

ABSTRACT

By reducing the cost of manufacturing electronic devices, the wide spread of smartphones and wearable electronic devices (smartwatches, smartbands) and a competitive global automotive market, several systems to provide convenience and user assistance have emerged in vehicles. One of these systems, passive entry and start, has been on the market for at least two decades, however, it has no connectivity with the external environment. To meet this gap and to take advantage of the growing market for smart devices, there is a natural tendency to converge these devices to the automotive environment by taking advantage of the ever-increasing emergence and spread of Bluetooth Low Energy (BLE) technology. This work presents a device localization system based on BLE that plays the role of a vehicular key. It presents the hardware used for system design, antenna topologies positioned in the vehicle (BLE dongles), architecture and flow of the localization system, presenting the machine learning algorithms used, using two layers and cascaded machines. It presents the chosen decision regions and presents a discussion about the system input parameter, RSSI. It shows the non-linearity of this parameter and its impact on the quality of the results obtained.

Keywords: Vehicle passive entry and start systems. Bluetooth Low Energy. Fingerprinting. RSSI. Machine learning.

LISTA DE FIGURAS

Figura 1 – Diagrama simplificado de sistema de comunicação RKE.....	21
Figura 2 – Diagrama de blocos de sistema PKE.....	22
Figura 3 – Esquema de troca de mensagens em sistemas PKE.....	23
Figura 4 – Campo das antenas LF externas. [16].....	23
Figura 5 – Campo das antenas LF internas. [16].....	23
Figura 6 – Força de campo magnético versus distância. [16].....	25
Figura 7 – Representação da evolução das chaves veiculares. [17].....	26
Figura 8 – Chave secundária do Model 3 da Tesla.....	27
Figura 9 – Total de remessas de dispositivos Bluetooth (em bilhão de unidades) [19]	28
Figura 10 – Pilha do protocolo BLE apresentando controlador e host. [21].....	30
Figura 11 – 40 canais do protocolo BLE e canais Wifi que se superpõem à faixa de operação do BLE. [22].....	32
Figura 12 – Máquina de estados da camada de enlace. [23].....	32
Figura 13 – Estrutura do pacote de dados da camada de enlace, com detalhamento para pacote de advertising. [24].....	35
Figura 14 – Exemplos de comunicação no estado de advertising, para modos ativo e passivo de scanning. [26].....	37
Figura 15 – Diagrama de blocos do rádio do SoC nRF52832 com destaque para o módulo RSSI. [27].....	38
Figura 16 – Diagrama de blocos de algoritmo baseado em fingerprinting. [32].....	44
Figura 17 – Regiões de Voronoi nos espaços bi e tridimensional.....	47
Figura 18 – Exemplo de classificação com kNN.....	47
Figura 19 – Exemplo de árvore de decisão. [33].....	49
Figura 20 – Representação do perceptron. [34].....	53
Figura 21 – Esquerda: representação simplificada de um perceptron, a rede neural mais simples. Direita: MLP com duas camadas escondidas. [34].....	54
Figura 22 – Vista superior da placa nRF52.....	69
Figura 23 – Vista inferior da placa nRF52.....	69
Figura 24 – Diagrama de blocos dos módulos utilizados no experimento.....	71

Figura 25 – Exemplo de configuração de antenas (losangos) e posição de captura de sinais de RSSI (círculos).....	72
Figura 26 – Regiões de localização da chave.....	72
Figura 27 – Geração da base de dados.....	72
Figura 28 – Diagrama generalizado do sistema de acesso veicular.....	73
Figura 29 – Diagrama de blocos expandido do controlador de acesso.....	74
Figura 30 – Módulo de desenvolvimento nRF52 DK.....	74
Figura 31 – Medição de corrente em dispositivo em scanning.....	78
Figura 32 – Corrente para scan com janela fixa em 50ms.....	79
Figura 33 – Medição de tensão consumida durante advertising.....	80
Figura 34 – Medição de tensão em advertising com foco na área de transmissão efetiva.....	80
Figura 35 – Corrente consumida em advertising.....	80
Figura 36 – VN8912 como gateway entre BCM e RFHM do Renegade.....	82
Figura 37 – Módulo VN8912 com placa VN8970 da Vector.....	82
Figura 38 – Arquitetura de software do VN.....	83
Figura 39 – Parte do painel traseiro do VN8912.....	84
Figura 40 – Diagrama de blocos do sistema PASE da Continental retirado de manual do produto submetido à ANATEL.....	85
Figura 41 – Pinagem dos conectores da RFHM extraído de manual do produto submetido à ANATEL.....	85
Figura 42 – Fios do conector da RFHM ligado à rede veicular alterados.....	86
Figura 43 – Componentes de hardware para desenvolvimento e teste do sistema de localização.....	87
Figura 44 – Diagrama simplificado do fluxo de comunicação entre os dispositivos..	88
Figura 45 – Fluxo de treinamento da máquina de aprendizagem.....	89
Figura 46 – Fluxo do sistema de localização.....	89
Figura 47 – Formato do pacote de advertising BLE.....	90
Figura 48 – Comportamento do RSSI com a distância.....	91
Figura 49 – Valores de RSSI obtidos com distância de 1m entre dispositivos.....	92
Figura 50 – Curva de return loss de antena utilizada no espectro de Wifi.....	92
Figura 51 – Curva de VSWR da antena utilizada no espectro de Wifi.....	93
Figura 52 – Comportamento do sinal de RSSI variando tamanho da janela.....	95
Figura 53 – Exemplo de dados normalizados.....	96

Figura 54 – Arquitetura de 2 níveis da máquina de aprendizagem.....	97
Figura 55 – Fluxo do sistema proposto.....	98
Figura 56 – Metodologia dos experimentos.....	99
Figura 57 – Antenas LF e módulo RFHM do sistema PASE do Jeep Renegade....	100
Figura 58 – Exemplo de grid de pontos de captura para o método de fingerprinting.	102
Figura 59 – Matriz de confusão considerando maior diferenciação entre as regiões.	103
Figura 60 – Representação de base de dados para topologia de 3 antenas.....	104
Figura 61 – Exemplo de base capturada em topologia de 3 antenas, dados normalizados e filtrados com janela de tamanho 15.....	106
Figura 62 – Máquina de aprendizagem utilizada nos experimentos.....	106
Figura 63 – Acurácia dos algoritmos de aprendizagem.....	107
Figura 64 – Topologia 1.....	109
Figura 65 – Matriz de confusão do treinamento da Topologia 1 com janela de tamanho 15.....	109
Figura 66 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 15.	109
Figura 67 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 20.	109
Figura 68 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 25.	109
Figura 69 – Topologia 2.....	110
Figura 70 – Matriz de confusão de treinamento da Topologia 2 com janela de tamanho 15.....	111
Figura 71 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 15.	111
Figura 72 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 20.	112
Figura 73 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 25.	112
Figura 74 – Topologia 3.....	112
Figura 75 – Matriz de confusão de treinamento da Topologia 3 com janela de tamanho 15.....	112

Figura 76 – Matriz de confusão de teste da Topologia 3 com janela de tamanho 15.	113
Figura 77 – Matriz de confusão de teste da Topologia 3 com janela de tamanho 20.	113
Figura 78 – Topologia 4.....	114
Figura 79 – Matriz de confusão de treinamento da Topologia 4 com janela de tamanho 15.....	114
Figura 80 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 10.	114
Figura 81 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 15.	114
Figura 82 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 20.	115
Figura 83 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 25.	115
Figura 84 – Topologia 5.....	116
Figura 85 – Matriz de confusão de treinamento da Topologia 5 com janela de tamanho 15.....	116

LISTA DE TABELAS

Tabela 1 – Tabela comparativa de tecnologias de localização <i>indoor</i>	57
Tabela 2 – Tabela comparativa dos parâmetros usados em sistemas de localização <i>indoor</i>	58
Tabela 3 – Comparativo de erro médio em metros entre modelos de [36].....	59
Tabela 4 – Tabela comparativa de sistemas de localização <i>indoor</i>	62
Tabela 5 – Divisão dos pontos do <i>grid</i> entre as regiões de decisão.....	102
Tabela 6 – Exemplo de valores de RSSI capturados.....	104
Tabela 7 – Exemplo de características de entrada para máquina de aprendizagem.	104
Tabela 8 – Comparativo entre topologias testadas; janela de tamanho 15.....	117

SUMÁRIO

1	INTRODUÇÃO	14
1.1	MOTIVAÇÃO E CONTEXTO.....	14
1.2	OBJETIVO GERAL.....	16
1.3	OBJETIVOS ESPECÍFICOS.....	16
1.4	ESTRUTURA DA DISSERTAÇÃO.....	17
2	SISTEMAS DE ACESSO VEICULAR REMOTO E <i>BLUETOOTH LOW ENERGY</i>	19
2.1	INTRODUÇÃO.....	19
2.2	SISTEMAS RKE E PKE.....	21
2.3	PAAK (PHONE AS A KEY) OU PRÓXIMO PASSO NA EVOLUÇÃO DE SISTEMAS DE ACESSO.....	25
2.4	INTRODUÇÃO AO PROTOCOLO BLE.....	27
2.5	ARQUITETURA DO PROTOCOLO BLE.....	29
2.5.1	Camada física	30
2.5.2	Camada de enlace	32
2.5.2.1	Interface <i>host</i> controlador (HCI).....	34
2.5.2.2	Pacotes de <i>advertising</i> e de dados.....	34
2.6	PARÂMETROS DE COMUNICAÇÃO DE <i>ADVERTISING</i>	36
2.7	RSSI.....	37
3	SISTEMAS DE LOCALIZAÇÃO	39
3.1	INTRODUÇÃO.....	39
3.2	<i>FINGERPRINTING</i>	42
3.3	MÁQUINAS DE APRENDIZAGEM.....	45
3.3.1	<i>k</i>-Nearest Neighbors	45
3.3.2	Árvore de decisão	48
3.3.3	Redes neurais	51
3.3.4	Sistema proposto	55
4	TRABALHOS RELACIONADOS	56
4.1	TECNOLOGIAS, PARÂMETROS E SISTEMAS DE LOCALIZAÇÃO <i>INDOOR</i>	56
4.2	PATENTES.....	63
5	SISTEMA PROPOSTO	67

5.1	INTRODUÇÃO.....	67
5.2	CONFIGURAÇÃO DO EXPERIMENTO.....	68
5.3	ARQUITETURA DO SISTEMA DE ACESSO VEICULAR.....	72
5.3.1	Funcionamento geral.....	73
5.3.2	Chave veicular – módulo BLE na função advertiser.....	74
5.3.3	Dongle BLE – módulo BLE na função scan.....	76
5.3.4	Consumo de potência.....	77
5.3.5	Módulo de localização – Arduino.....	80
5.3.6	Módulo controlador – Vector VN8900.....	81
5.3.7	Montagem do sistema e periféricos.....	84
5.4	DESCRIÇÃO DO SISTEMA DE LOCALIZAÇÃO.....	86
5.4.1	Captura de RSSI.....	89
5.4.2	Filtro – janela passante.....	93
5.4.3	Normalização.....	95
5.4.4	Máquina de aprendizagem.....	96
6	RESULTADOS.....	99
6.1	METODOLOGIA.....	99
6.1.1	Topologia das antenas.....	99
6.1.2	Captura de sinais de RSSI.....	100
6.1.3	Máquina de aprendizagem.....	104
6.2	RESULTADOS.....	107
6.2.1	Topologia 1.....	108
6.2.2	Topologia 2.....	110
6.2.3	Topologia 3.....	112
6.2.4	Topologia 4.....	113
6.2.5	Topologia 5.....	115
6.3	DISCUSSÃO.....	116
7	CONCLUSÕES E TRABALHOS FUTUROS.....	118
7.1	CONSIDERAÇÕES FINAIS E CONTRIBUIÇÕES.....	118
7.2	TRABALHOS FUTUROS.....	119
	REFERÊNCIAS.....	121

1 INTRODUÇÃO

Esta seção tem como objetivo apresentar as principais motivações para realização deste trabalho, assim como a contextualização do problema atual na descrição do protocolo de comunicação *Bluetooth Low Energy* e sistemas atuais de acesso e partida passivos em veículos. Em seguida são apresentados os objetivos gerais e específicos da dissertação. Por fim, será apresentada a estrutura geral do documento.

1.1 MOTIVAÇÃO E CONTEXTO

A rápida disseminação de dispositivos eletrônicos, diminuição dos seus custos e a massificação de dispositivos inteligentes de uso pessoal tem modificado a relação dos usuários com o ambiente em seu entorno. Cada vez mais, o uso de *smartphones* e outros dispositivos *smart* tem se tornado presentes em atividades cotidianas, desde com ajuda na escolha de locais para fazer compras de bens, ingressos, programar viagens, *player* de música, leitura digital, comunicação com familiares, equipes de trabalho, monitoramento e armazenamento de dados vitais (como batimentos cardíacos), monitoramento de atividades físicas, acesso a ambientes, realização de pagamentos, entre outra infinidade de aplicações.

Com o avanço no uso destes dispositivos, houve um aumento crescente no interesse em determinar a localização do usuário [1] para melhorar a experiência baseada no contexto e empresas que dominam o tráfego da Internet podem usar essa informação para aumentar a qualidade dos anúncios de publicidade. Portanto, a demanda comercial por serviços de proximidade vem aumentando de forma exponencial, com mercado chegando a U\$4,4 bilhões em 2019 [2].

A conectividade proporcionada pela proliferação dos dispositivos *smart* também causa impactos e mudanças no mercado automotivo. Em 2016, foi criado o *Car Connectivity Consortium* com o objetivo de criar padrões para comunicação de dispositivos com veículos. Um dos projetos em andamento é o *Digital Key*. É um esforço de padronização para uso dos dispositivos *smart* como chaves veiculares.

Essa iniciativa vem a partir de uma tendência crescente das montadoras para inserir seus modelos nessa realidade, como exemplos, há projetos da Tesla, Ford [3], Volvo. Esses projetos têm como características a garantia da segurança e autenticidade na troca de informações. Mas, além disso, há a necessidade também de permitir comodidade ao usuário na execução das tarefas para uso do veículo, como, por exemplo, permitir que o dispositivo *smart* apresente passividade na abertura do carro, sem necessidade de ação específica exceto aproximação. Essa característica faz parte dos sistemas PKE (*Passive Keyless Entry*) ou PASE (*Passive Start and Entry*) já existentes. Desse modo, métodos de localização *indoor* têm papel fundamental no desenvolvimento de sistemas de acesso veicular baseados em dispositivos *smart*, para permitir que o veículo identifique a posição relativa da nova chave.

Nesse sentido, uma que surgiu, dentre várias tecnologias de comunicação de baixo consumo, voltadas para aplicações em Internet das Coisas, oferecendo características como conectividade entre vários dispositivos simultaneamente, segurança, autenticidade, larga presença em dispositivos *smart*, baixo consumo de potência e determinação de posicionamento da chave em relação ao veículo é *Bluetooth Low Energy* (BLE).

Sistemas de localização *indoor* (IPS – *Indoor Positioning Systems*) que usam BLE como tecnologia de comunicação vêm sendo propostos [4][5][6][7]. Em sua maioria, esses sistemas fazem uso do dado de potência de sinal recebido (RSSI – *received signal strenght*) como parâmetro de entrada de algoritmo de cálculo da posição. Esse algoritmo pode ser baseado em método geométrico (trilateração, por exemplo), método bayesiano, de minimização de função de custo ou *fingerprinting*, sendo este último o mais utilizado nos sistemas IPS baseados em BLE. Normalmente esses sistemas tem apresentando erro médio de posicionamento ainda elevado, normalmente acima de 1m em ambientes fechados controlados. O problema de determinação da posição para o caso automotivo, posição da chave em relação ao veículo, não encontra muitos estudos na literatura [8], mas uma profusão de patentes sobre o tema.

Atualmente, a indústria automobilística vem presenciando um notável desenvolvimento de sistemas e dispositivos destinados a auxiliar os condutores na realização de certas tarefas, tais como, por exemplo, abertura e fechamento de portas, acionamento de vidros, ignição e acionamento de alarmes, entre outras. Não

obstante tais avanços, a possibilidade de atuação do condutor distante do veículo gera inúmeros problemas relacionados com a segurança dos mecanismos de comunicação entre veículo e condutor.

Em particular, são conhecidos na literatura diversos dispositivos, sistemas e plataformas destinadas a promover uma interação do usuário com o veículo, de modo a permitir que algumas tarefas sejam realizadas de forma autônoma pelo veículo a partir da detecção de um sinal proveniente do usuário, ou de uma ação deste usuário.

O presente trabalho apresenta uma plataforma, ou genericamente um sistema, para a localização, em uma região previamente determinada de um dispositivo destinado a comandar uma ou mais funções automotivas à distância. Compreende um sistema apto a determinar a presença de um dispositivo de comando, tal como um dispositivo de acesso, em uma dada região extra ou intra veicular, a partir da emissão e recepção de *beacons*. A determinação das regiões veiculares é obtida a partir da posição relativa entre as antenas receptoras. No sistema proposto, não serão considerados experimentos e testes realizados em tempo real, ou seja, dinamicamente, com chave sendo movimentada ao redor do veículo, mas sim em pontos em um grid previamente definido e minimizando o máximo possível interferências de componentes externos, inclusive do corpo humano.

1.2 OBJETIVO GERAL

O objetivo geral deste trabalho é apresentar um sistema de acesso e partida passivos (PASE – *Passive Entry and Start*) veicular baseado em plataforma embarcada de desenvolvimento para a tecnologia *Bluetooth Low Energy* (BLE), focando na apresentação de um sistema de localização que usa aprendizagem de máquina para construção do algoritmo de decisão.

1.3 OBJETIVOS ESPECÍFICOS

Para alcançar o objetivo geral deste trabalho, foram definidos alguns objetivos específicos e características do sistema:

1. Sistema de localização proposto deve prover acesso passivo e partida passiva;
2. Sistema proposto deve utilizar tecnologia inovadora, não aplicada nos sistemas de acesso atuais, provendo conectividade com dispositivos *smart* presentes no mercado;
3. Limiar de abertura do veículo deve estar numa região distante em torno de um metro e meio do veículo;
4. Desenvolvimento do sistema proposto deve ter em consideração limite de consumo de potência para sistemas de acesso veiculares presentes no mercado;
5. Tempo de resposta do sistema para tomada de decisão não deve passar do limite dos sistemas atuais, em torno de 300ms;
6. Sistema proposto deve ser testado em ambiente controlado, evitando efeitos causados por interferência, inclusive do corpo humano, com pontos de coleta de dados estáticos, sem variação de posição ao longo de uma medição e definidos previamente dentro de um grid ao redor e no interior do veículo.

1.4 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação encontra-se estruturada da seguinte forma: na Seção 2 é apresentado breve histórico de sistemas de abertura de veículos com sistemas como RKE e PKE. Na Seção 3 é apresentado o protocolo BLE e suas características, como troca de mensagens de *advertising* e comportamento do valor de RSSI referente a esse protocolo. Na Seção 3, algoritmos de localização *indoor* e os parâmetros que esses protocolos utilizam são apresentados brevemente dando ênfase à técnica de *fingerprinting* e aos algoritmos de aprendizagem de máquina. Na Seção 4 é feito o estudo do estado da arte associado aos sistemas de acesso passivo a veículo, apresentando algumas patentes recentes relacionadas ao uso de BLE para a resolução desse problema. Na Seção 5 é apresentado o sistema proposto, com descrição do ambiente experimental criado e utilizado ao longo do desenvolvimento do estudo, os componentes de *hardware* utilizados, como foram conectados e os papéis que desempenharam no fluxo do sistema. Também neste capítulo, é apresentado o comportamento do parâmetro utilizado como entrada do

sistema de localização, o RSSI. Ainda são apresentados os módulos e fluxo do sistema de localização particularmente. Na Seção 6 são apresentados os resultados dos experimentos para algumas topologias de antenas testadas. Na Seção 8 são apresentadas as conclusões e trabalhos futuros. E finalmente no último capítulo há uma lista dos trabalhos que serviram de base para o desenvolvimento desta dissertação.

2 SISTEMAS DE ACESSO VEICULAR REMOTO E *BLUETOOTH LOW ENERGY*

Esta seção apresenta uma visão geral sobre sistemas de abertura e partida passivos e uma pequena fundamentação teórica para o protocolo *Bluetooth Low Energy* (BLE). Apresenta arquitetura de rede do protocolo, o indicador, *Received Signal Strength Indicator* (RSSI), utilizado como parâmetro para sistemas de localização baseados em BLE. Apresenta os conceitos de serviços e camadas disponibilizados pelo protocolo, detalhamento da camada física e uso do RSSI como parâmetro de distância entre dispositivos.

2.1 INTRODUÇÃO

A introdução de sistemas de acesso remoto ou, em inglês, *remote entry systems*, no mercado automotivo foi iniciado na década de 1980 com a introdução de sistemas, por exemplo, que usavam teclado numérico, como o sistema SecuriCode da montadora Ford, ou chaves que usavam tecnologia de infravermelho para troca de mensagens com veículo, existente, por exemplo, em modelos de veículo da montadora Mercedes-Benz ou do primeiro veículo a apresentar sistema de acesso remoto, o Renault Fuego [9]. No mercado brasileiro, até anos recentes, a tecnologia dominante para acesso e partida de veículos ainda era a simples chave mecânica, sem o uso nem mesmo de sistemas de alarme instalados fora da concessionária, os chamados acessórios *after-market*. No entanto, com a abertura do mercado automotivo para importação de veículos no início dos anos 1990 [10], esse cenário começou a mudar. E seguindo uma tendência global de mercado com consumidores mais exigentes junto com ampla disseminação e barateamento de sistemas eletrônicos, sistemas de acesso utilizando comunicação via rádio são hoje bastante comuns.

Os sistemas atuais de *Remote Keyless Entry* (RKE) permitem, além de travamento e destravamento das portas nos sistemas mais básicos, partida do motor do veículo, abertura ou fechamento de janelas, acionamento do sistema de ar condicionado do veículo em sistemas mais completos, entre outras comodidades.

Todas as funcionalidades desses sistemas, no entanto, ainda dependem do acionamento de botões presentes nas chaves, exigindo uma ação do usuário.

Para permitir ainda maior comodidade ao usuário, o mercado automotivo introduziu o conceito de *Passive Keyless Entry* (PKE), também denominado na literatura de *Passive Entry Passive Start* (PEPS ou PASE) ou *smart key*. Nesse sistema, o usuário não precisa tomar nenhuma ação além de se aproximar do veículo com a chave próxima ao corpo. Através de um protocolo de comunicação por rádio o veículo troca mensagens com a chave e, a depender da distância entre eles, aciona abertura das portas ou partida do veículo. O primeiro sistema desse tipo foi desenvolvido pela Siemens em 1995 [11] e lançado pela Mercedes-Benz em 1998 com o nome de *Keyless Go*. Segundo relatório produzido pela companhia de pesquisa de mercado *Research and Markets*, a previsão de crescimento anual na adoção dos sistemas PKE no período de 2016 a 2020 é de 10,6% [12], com aumento das vendas especialmente nos mercados emergentes. Para isso, as montadoras têm se adaptado. Na faixa de veículos de luxo, a adoção desses sistemas foi mais acelerada, com 88% dos veículos desse segmento produzidos com sistemas PKE no ano de 2014. Desde 2008, as montadoras vêm simplificando e barateando esses sistemas, por exemplo, adotando o uso de botões nas maçanetas, para integrar esses sistemas no segmento de carros médios [13].

Desde a introdução desses sistemas, questões de segurança surgiram e tem causado discussões sobre a ampla adoção desses sistemas, inclusive com impactos, por exemplo, no mercado de seguros de veículos, nos quais há casos de seguradoras aumentando valores ou mesmo rejeitando seguros para veículos com sistemas RKE/PKE comprovadamente falhos e facilmente burlados por ladrões com acesso a sistemas de ataque cada vez mais baratos e efetivos. Questiona-se, assim, se o caminho para a tecnologia para acesso veicular é uma volta para uso de chaves mecânicas. Sabe-se que há também facilidade de cópia dessas chaves e, portanto, não se vislumbra esse cenário. O caminho que se apresenta como tendência, e isso não somente no mundo automotivo, é o da integração e convergência de tecnologias e funcionalidades em dispositivos móveis, os *smartphones*, deixando a chave mecânica cada vez mais numa posição de *backup* em caso de falhas dos novos sistemas. Como cada vez mais a presença de *smartphones* é item indispensável no cotidiano, esse item – como outros que estão

surgindo e tornando-se *smart*, como *smartwatches*, *smartbands* – apresenta-se como alternativa potencial para transformar-se em chave veicular.

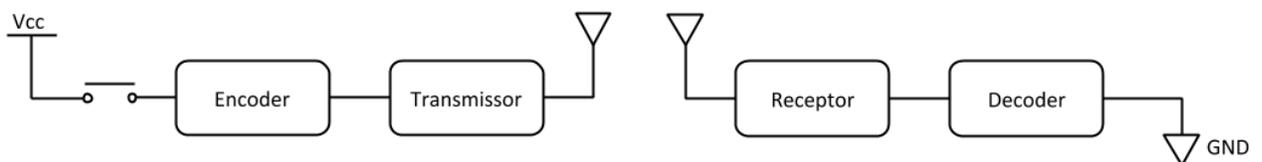
Neste capítulo serão discutidos os sistemas atuais de acesso veicular via rádio e através de qual tecnologia de comunicação esses novos dispositivos *smart* podem ser usados como chave veicular, apresentando introdução à tecnologia mais promissora, o *Bluetooth Low Energy* (BLE).

2.2 SISTEMAS RKE E PKE

Tanto o sistema RKE como PKE apresentam sistemas de comunicação que operam na em faixas de frequência idênticas e não sofreram mudanças significativas desde suas introduções, a não ser em relação a modificações originadas por falhas de segurança desses sistemas, permitindo diversos ataques para acessos não permitidos, tais como *scan attack*, *playback attack*, *two-thief attack* ou *relay attack*, uso de *jammers* de sinal, causando interferência, cópia de códigos de chave através de comunicação via OBD (*On Board Diagnostics*) do veículo [14]. As modificações foram no sentido de deixar os sistemas seguros, introduzindo números de identificação únicos para cada chave e veículo, *rolling codes*, uso de criptografia, *immobilisers*, entre outros.

No entanto, o princípio de funcionamento da comunicação não foi modificado. Em sistemas mais simples, a comunicação é unidirecional e o acesso se baseia em manter iguais os códigos de acesso, sendo codificados no transmissor e decodificados no receptor [15].

Figura 1 – Diagrama simplificado de sistema de comunicação RKE



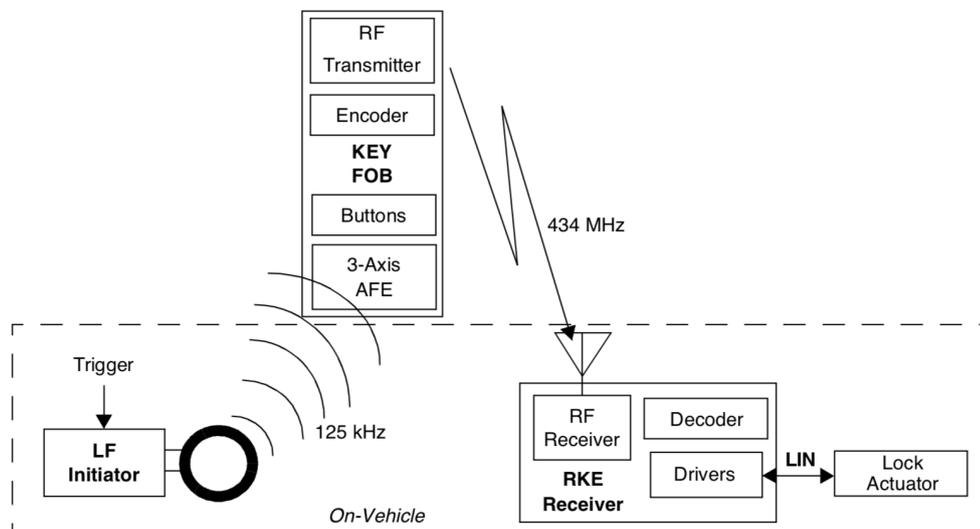
Como mostrado na Figura 1, os sistemas RKE necessitam de um ou mais botões, *switches* ou contatos para usuário escolher a ação a ser realizada. O encoder detecta o botão pressionado e converte em um pacote de dados digitais, convertido em sinal de rádio pelo transmissor e enviado para o receptor, no caso, o veículo. O decodificador traduz o sinal de rádio recebido e determina se o pacote de

dados é válido para realização da ação desejada, enviando para o circuito ou rede veicular responsável pela ação.

Para obedecer a questões regulatórias dos diferentes países onde os rádios dos sistemas RKE operam, normalmente são usados nas frequências 315MHz ou 422,96MHz, na faixa de frequência UHF. A diferença entre os sistemas RKE e PKE na camada física é a presença nos sistemas PKE um componente adicional de comunicação operando na faixa de baixa frequência em, tipicamente, 125KHz. A Figura 2 mostra diagrama de blocos de sistema PKE desenvolvido pela fabricante Microchip, a partir de sistema RKE já existente [16]. Como discutido, novas funcionalidades e melhorias nos protocolos de segurança foram adotadas ao longo do tempo, mas o princípio básico de funcionamento permanece.

Portanto, as tecnologias em que se baseiam os sistemas PKE são geração e medição de sinais de LF e troca de mensagens via sinais de UHF. Sistemas PKE tem comumente as mesmas funções de sistemas RKE, com botões na chave para o usuário acionar funcionalidades quando necessitar.

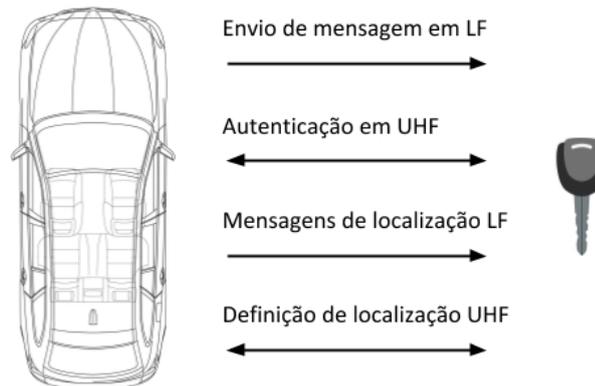
Figura 2 – Diagrama de blocos de sistema PKE



A forma de abertura das portas do veículo nesses sistemas varia entre sistemas, mas basicamente, pode operar através de proximidade – quando a chave se aproxima da distância configurada ocorre destravamento das portas – ou através de acionamento de algum sensor ou botão localizado na maçaneta. Nos dois casos, a chave está em modo *idle* ou *sleep*, ou seja, está no estado de menor consumo de potência, estendendo a vida útil da bateria. Seja no caso da proximidade ou

acionamento por sensor, o sistema PKE do veículo envia um sinal de *wake-up* em LF para a chave – continuamente no primeiro caso e quando acionado no segundo. A chave, ao receber o sinal LF com a potência e formato corretos, sai do modo *sleep* e inicia a comunicação em UHF para troca de mensagens de autenticação. Essa troca de mensagens é mostrada simplificada na Figura 3.

Figura 3 – Esquema de troca de mensagens em sistemas PKE



A localização se inicia após a autenticação da chave pelo veículo. Normalmente, sistemas PKE utilizam de quatro a seis antenas para definir o posicionamento da chave, se dentro ou fora do carro. Essas antenas produzem um campo magnético ao redor e no interior do carro, conforme as Figura 4 e Figura 5, retiradas de sistema PKE desenvolvido pela fabricante Atmel. Devido a propagação de sinal e para evitar oclusão por obstáculos, como o corpo humano, as antenas são comumente dispostas no teto do veículo.

Figura 4 – Campo das antenas LF externas. [16]

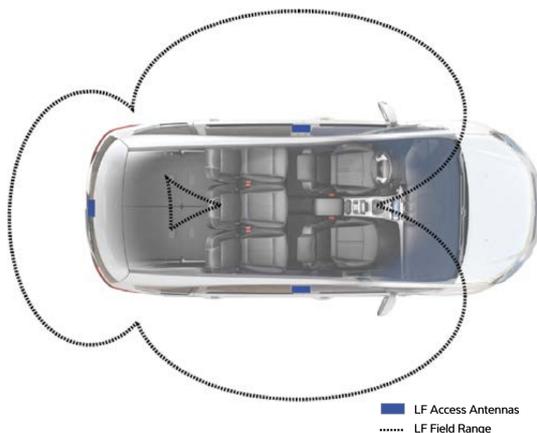
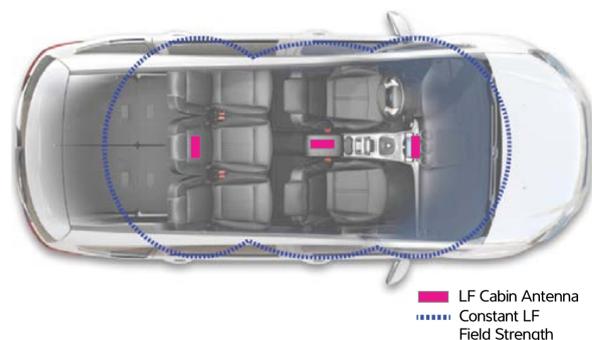


Figura 5 – Campo das antenas LF internas. [16]



O campo gerado pelas antenas é recebido na chave por um sistema de 3 antenas discretas ou uma única bobina que captura o sinal nos três eixos, x , y , z . O indicador de potência do sinal recebido, RSSI (*Received Signal Strength Indicator*), é medido na chave e enviado para o veículo via mensagens em UHF. A partir desses valores, a posição da chave é determinada.

O princípio de funcionamento do sistema de localização se baseia na recepção do campo magnético gerado pelas antenas do veículo pela chave. No espaço livre, a propagação desse campo e, em seguida, recepção na chave tem um comportamento previsível e depende do fluxo de corrente na antena e do número de voltas com que é feita a antena. No entanto, esse campo pode ser distorcido pela presença de materiais ferrosos. O gráfico presente na Figura 6 apresenta força do campo magnético dependente da distância para duas antenas com raios diferentes.

Como apresentado, o campo magnético decai em uma taxa exponencial com a distância, a uma relação de $1/d^3$. Essa propriedade é importante para o funcionamento do sistema, pois limita o alcance dos sistemas PKE para até cerca de 5 metros.

Outros fatores que implicam na qualidade dos sistemas PKE são afinar as bobinas tanto do veículo como da chave para a mesma frequência, através das equações mostradas, compensações do sistema para fatores externos, como variações na temperatura, arquitetura do conversor A/D escolhido, como função de transferência utilizada e resolução escolhida.

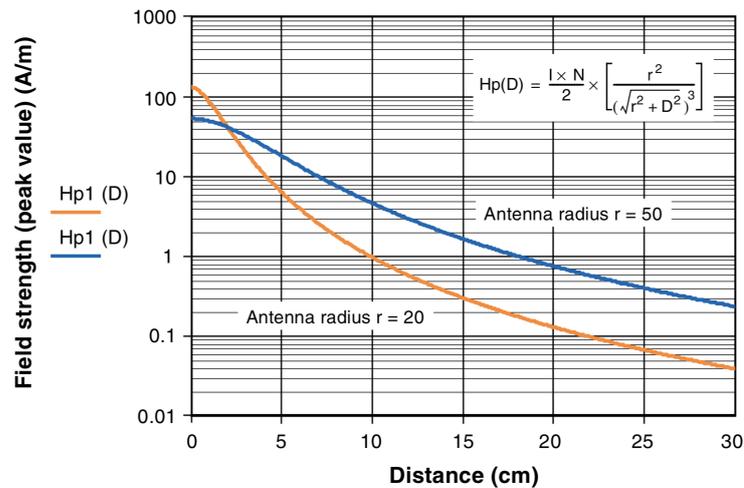
$$Z_{bobina} = R + j(X_L - X_C)$$

$$X_C = \frac{1}{2\pi fC} \quad X_L = 2\pi fL$$

$$f_{reson} = \frac{1}{2\pi\sqrt{LC}}$$

Essas dificuldades técnicas hoje são conhecidas e os fabricantes de sistemas PKE (Atmel, NXP, Siemens, Continental, Hella, ZF TRW, Valeo, entre outros) oferecem sistemas com precisão centimétrica de localização da chave.

Figura 6 – Força de campo magnético versus distância. [16]



2.3 PAAK (PHONE AS A KEY) OU PRÓXIMO PASSO NA EVOLUÇÃO DE SISTEMAS DE ACESSO

Os sistemas de acesso veicular ou, simplesmente, as chaves de carro, vêm apresentando uma evolução ao longo dos anos. Algumas dessas chaves são apresentadas na Figura 7, representando essa evolução que trouxe novas funcionalidades e, talvez, apresente o fim da chave mecânica utilizada até hoje. Poucas montadoras estão convencidas ainda em adotar definitivamente o uso de *smartphones* ou quaisquer outros dispositivos *smart*. No entanto, a montadora Tesla apresentou este ano o modelo Model 3, com uso preferencial do *smartphone* como chave de acesso, tendo um cartão NFC como chave secundária, Figura 8.

Alguns dos pontos citados pelas montadoras na negativa da adoção de *smartphones* ou dispositivos eletrônicos vestíveis (*wearables*) são: mudança drástica de uma tecnologia já estabelecida e aceita pelo público para outra radicalmente diferente, podendo gerar lacunas de segurança e conveniência; limitações das tecnologias utilizadas, por exemplo, duração da bateria desses equipamentos insuficiente para uso prologando e usuário ter que depender de chaves secundárias; no caso de *smartphones* especificamente, há ainda questões envolvendo tempo de vida útil do dispositivo, ao longo da qual há atualizações de sistema operacional que podem criar brechas de segurança ou provocar problemas de funcionamento no aplicativo de autenticação e ainda questões relacionadas à repasse do veículo para outros usuários. Por outro lado, pontos positivos apontados são preços abusivos cobrados por montadoras para reposição das chaves PKE atuais em casa de perda

ou roubo e que as atualizações de *software* em *smartphones* para retirar falhas de segurança são mais frequentes e eficientes que as atualizações de *hardware* que as montadoras atualmente realizam [18].

Figura 7 – Representação da evolução das chaves veiculares. [17]



Apesar desse impasse enfrentado pelas montadoras mais tradicionais do mercado de automóveis, o cenário aponta para a adoção de dispositivos *smart* também como chave de acesso veicular, acompanhando uma tendência global de centralização de diversos outros dispositivos com as mais variadas funções em dispositivos únicos, sendo os maiores candidatos os *smartphones* e *smartwatches*. Esses dispositivos estão no centro do surgimento de uma quantidade cada vez maior de aplicações agrupadas sob o título de Internet das Coisas ou *Internet of Things* (IoT). Uma das tecnologias, e talvez uma das mais centrais, no desenvolvimento das aplicações IoT é o novo padrão de comunicação *wireless* de baixo consumo de potência que surgiu a partir do Bluetooth clássico, o *Bluetooth Low Energy* (BLE).

Figura 8 – Chave secundária do Model 3 da Tesla.

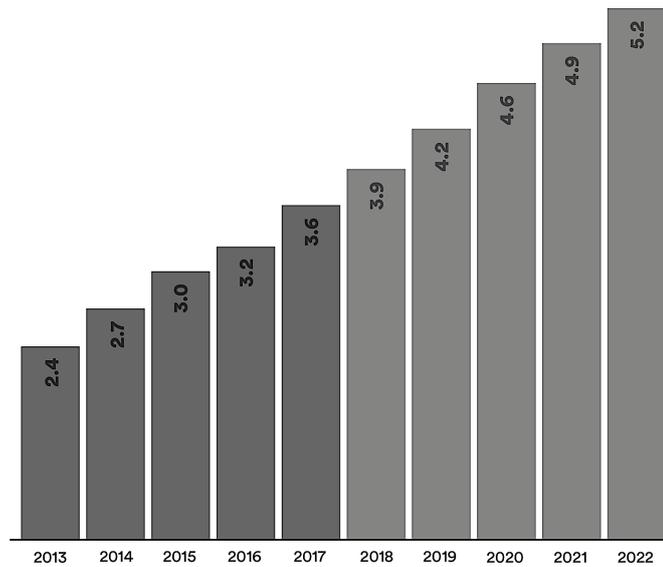


Como será discutido nos próximos capítulos, essa tecnologia é a que tem possibilitado o uso de *smartphones* e *wearables* como dispositivos de acesso veicular, pelas suas características de segurança, baixo consumo de potência e uso em protocolos de autenticação e localização. Nas próximas seções, uma visão geral da tecnologia BLE será apresentada.

2.4 INTRODUÇÃO AO PROTOCOLO BLE

Bluetooth Low Energy, também conhecido como *Bluetooth Smart*, é uma evolução do *Bluetooth* clássico e foi introduzido como parte do padrão *Bluetooth 4.0*. Apesar de haver algum tipo de sobreposição com o *Bluetooth* clássico, o padrão BLE, na verdade, veio com uma proposta bem diferente e foi inicialmente desenvolvido pela Nokia sob um projeto interno de nome *Wibree* antes de ser adotado como padrão pelo *Bluetooth SIG (Specific Interest Group)*. Em períodos recentes, o padrão vem sendo adotado largamente pela indústria de tecnologia (Figura 9 mostra crescimento da produção de dispositivos *Bluetooth*), seja em *smartphones*, sistemas embarcados para uso pessoal (caixas de som, pulseiras inteligentes, relógios inteligentes), acessórios para computadores (mouses, teclados) e sistemas de som automotivos, por exemplo. Isso torna o padrão BLE altamente atrativo para desenvolvimento de projeto de novos sistemas eletrônicos embarcados, principalmente aqueles que fazem parte do ecossistema de IoT, devido à disponibilidade tanto de *softwares* como de *hardwares* já desenvolvidos pela indústria e pela comunidade de desenvolvedores para a tecnologia BLE.

Figura 9 – Total de remessas de dispositivos Bluetooth (em bilhão de unidades) [19]



Algumas características do padrão BLE [20] são:

- **Menor consumo de potência:** um dos maiores apelos do protocolo é a redução do consumo. O máximo pico de consumo durante operação do rádio BLE consome 15mA e o consumo médio, usando métodos onde o dispositivo entre em modo *sleep*, é de 1uA. O consumo corresponde a cerca de 10% do *Bluetooth* clássico. A depender do tempo de trabalho do dispositivo, uma bateria moeda, com capacidade de 240mAh, pode chegar a ter mais de 5 anos de vida;
- **Compatibilidade e redução de custos:** para garantir compatibilidade com *Bluetooth* clássico há dois tipos de chipset disponíveis no mercado – *dual-mode* com BR/EDR (*Bluetooth Basic Rate/Enhanced Data Rate*) e BLE e *stand-alone* BLE, otimizado para baixo consumo e baixo custo;
- **Segurança e confiabilidade:** BLE utiliza da mesma tecnologia de *adaptive frequency hopping* (AFH) do *Bluetooth* clássico. Isso implica em maior confiabilidade e robustez na transmissão em ambientes ruidosos. Para diminuir consumo, BLE apresenta somente 40 canais de largura de 2MHz (especificação do *Bluetooth 5*), enquanto no *Bluetooth* clássico são 79 canais de 1MHz;
- **Co-existência com outros protocolos sem fio:** como opera na banda ISM de 2,4GHz, o protocolo BLE divide essa banda de frequência com outros protocolos como, Wifi, IEEE 802.15.4/ZigBee e outros

protocolos proprietários que podem compartilhar essa mesma banda. Essa divisão com tantos protocolos pode gerar interferências causando redução da performance de comunicação, sendo necessário uso de códigos corretores de erro e retransmissão, garantidos pelo protocolo BLE;

- **Alcance do sinal:** BLE apresenta um tipo de modulação um pouco diferente do utilizado no *Bluetooth* clássico, o que permite alcance de até 300m na potência de transmissão máxima permitida pelas chipsets, de 10dBm;
- **Facilidade de desenvolvimento e integração:** a rede BLE é baseada na relação mestre-escravo, onde o mestre controla quando os escravos podem trocar mensagens, através de requisições. Em comparação ao *Bluetooth* clássico, o protocolo BLE apresenta uma nova funcionalidade chamada de *advertiser*, na qual um dispositivo atuando como escravo pode anunciar através de pacotes de *advertising* a intenção de se conectar a um mestre ou mesmo usar essa mensagem para enviar dados.

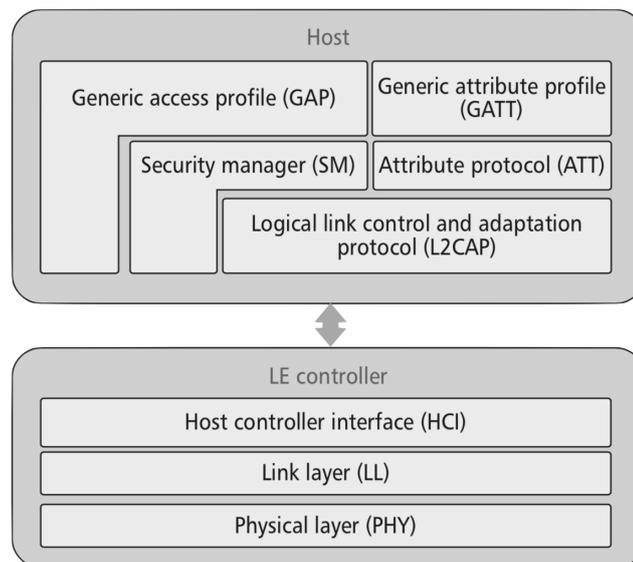
2.5 ARQUITETURA DO PROTOCOLO BLE

A arquitetura do protocolo BLE é fundamentalmente bastante simples. Como mostra a Figura 10, ela é dividida em três partes básicas: controlador, *host* e aplicação.

A pilha do protocolo BLE inclui a camada física (PHY – *Physical Layer*) (que transmite os bits de dados), camada de enlace (LL – *Link Layer*) (que define a estrutura dos pacotes de dados e controle) e interface *host* controlador (HCI – *Host Controller Interface*). Essas três camadas em conjunto são chamadas de *Bluetooth Low Energy Link Controller* ou controlador. Basicamente essa camada é um dispositivo que recebe e transmite sinais de rádio e interpreta esses sinais como pacotes transportando informações. Acima do controlador, a camada *host* incorpora a camada *Logical Link Control and Adaptation Protocol* (L2CAP) que provê uma abstração baseada em canal para serviços e aplicações. Essa camada executa fragmentação e de-fragmentação de dados de aplicação e multiplexa e demultiplexa múltiplos canais através de um link lógico compartilhado. Normalmente, essa

camada é tipicamente uma pilha de *software* que gerencia como dois ou mais dispositivos podem estabelecer comunicação e como uma variedade de serviços podem ser oferecidos simultaneamente pela mesma camada física, compartilhando recursos. A camada *host* inclui ainda a *Security Manager Protocol* (SMP) e a *Attribute Protocol* (ATT). A SMP usa a um canal fixo da L2CAP para implementar funcionalidades de segurança entre dispositivos. A ATT fornece um método de comunicação em pequenas quantidades de dados por um canal fixo da L2CAP. A camada *Generic Attribute* (GATT) *Profile* especifica a estrutura na qual dados são trocados. Essa estrutura define elementos básicos, como serviços e características, usados em aplicações. A camada *Generic Access Profile* (GAP) define os requisitos básicos de um dispositivo *Bluetooth*. A camada de aplicações permanece sobre essa pilha, oferecendo uma interface padrão para gerenciamento de dados BLE. Baseado no tipo de produto um dispositivo BLE pode incluir um ou mais desses serviços para o usuário.

Figura 10 – Pilha do protocolo BLE apresentando controlador e host. [21]



2.5.1 Camada física

A camada mais abaixo da pilha da BLE é a camada física (PHY). Ela é a responsável pela comunicação através da antena de rádio acoplada ao circuito, enviando e recebendo pacotes através da banda de frequência ISM de 2,4GHz. No protocolo BLE os sinais de rádio são modulados no esquema de modulação chamado *Gaussian Frequency Shift Keying* (GFSK), onde zeros e uns são

representados por um deslocamento positivo ou negativo de 185kHz da frequência da portadora do sinal, respectivamente. O *Bluetooth* clássico utiliza índice de modulação de 0,35 enquanto no protocolo BLE esse índice é de 0,5, dessa forma, diminuindo requisitos de consumo de potência. Também contribui para aumentar alcance e aumentar robustez do protocolo. A modulação GFSK do BLE usa um filtro gaussiano para suavizar a transição entre zeros e uns. Quando ocorre essa transição, um pulso de energia é espalhado por uma maior banda de frequência, além da designada para o canal em que os dados estão sendo propagados. Então, o filtro opera de forma a diminuir essas influências fora da banda do canal.

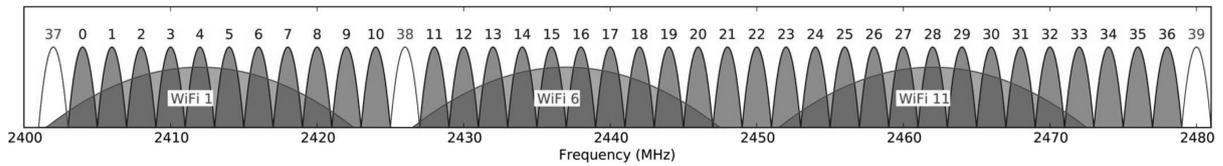
O protocolo BLE opera, como citado anteriormente, na banda ISM (*Industrial, Scientific and Medical*). Essa faixa é dividida em 40 canais todos com largura de banda de 2MHz e a potência de transmissão pode variar entre -20dBm e 10dBm. A frequência central dos canais BLE são dados pela Equação (1).

$$f_c = 2402 + 2k \quad (1)$$

Os canais da camada física são divididos em 37 canais de dados e três canais de *advertising*. Tais canais são utilizados para transmissão de pacotes de *advertising* que permitem que um dispositivo escravo realize *broadcast* de mensagens para todos os dispositivos ao alcance do sinal de rádio transmitido. Apesar de operar como canais de *broadcast* para descoberta de dispositivos, os pacotes de *advertising* apresentam um *payload* de dados, que podem ser usados para transmissão de informações, como, por exemplo, dados de sensores. Os canais de dados são utilizados para troca de informações entre dispositivos após o estabelecimento de conexão. A Figura 11 mostra uma representação desses canais, com destaque para os canais 37 (frequência central em 2402MHz), 38 (2426MHz) e 39 (2480MHz) usados para *advertising* e canais de Wifi que compartilham a mesma faixa de frequência BLE.

Figura 11 – 40 canais do protocolo BLE e canais Wifi que se superpõem à faixa de operação do BLE.

[22]

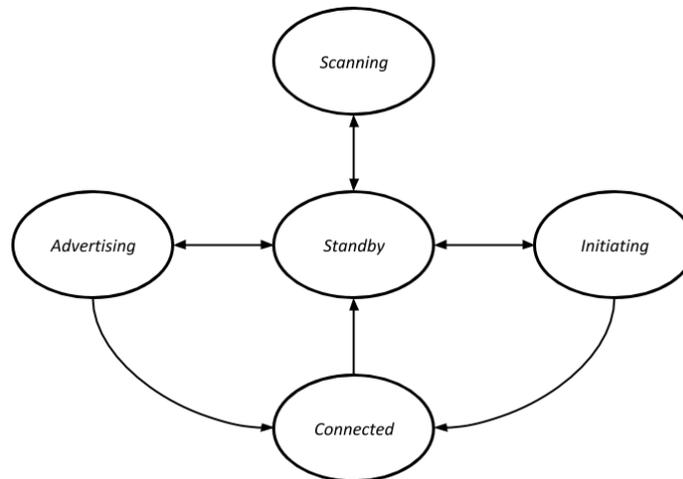


2.5.2 Camada de enlace

A camada de enlace, como mostrado na Figura 10 está diretamente acima da camada física. Pode-se dizer que esta camada é o núcleo do protocolo BLE, sendo a parte mais complexa da pilha. Ela é responsável pelas funções de *advertising*, escaneamento de pacotes, criação e manutenção de conexões. Também responsável por garantir que os pacotes transmitidos sejam corretamente estruturados.

Essa camada define cinco estados para um dispositivo BLE: *Standby*, *Advertising*, *Scanning*, *Initiating* e *Connected*. Os estados são relacionados conforme a Figura 12.

Figura 12 – Máquina de estados da camada de enlace. [23]



O estado *Standby* é aquele no qual o dispositivo permanece inativo (*idle*). A partir dele, dispositivo pode entrar em qualquer outro estado.

O estado de *Advertising* é onde a camada de enlace transmite pacotes de *advertising*. Um dispositivo que queira ser descoberto ou conectado deve ter implementado este estado. A partir deste estado, o dispositivo pode mudar para o

estado *Connected* se receber pacote de requisição de conexão de outro dispositivo no estado *Initiating*. Se houver parada no envio de mensagens de *advertising*, o dispositivo volta para o estado de *Standby*. A camada de enlace de dispositivo no estado de *Advertising* também pode receber e responder mensagens de outro dispositivo no estado de *Scanning*. O dispositivo que está em *Advertising* pode ser chamado de *advertiser*.

No estado de *Scanning*, o dispositivo está à espera de mensagens de *advertising* geradas por dispositivos no estado de *Advertising*. Há dois modos de operação para esse estado: *active scanning* e *passive scanning*. No último, o dispositivo somente recebe mensagens de *advertising*. No modo ativo, pode enviar mensagens chamadas *scan requests* e receber do *advertiser* mensagens do tipo *scan response* com os dados adicionais requisitados. Ao parar de realizar *scanning*, o dispositivo volta para o estado de *Standby*. Dispositivo no estado de *Scanning* é também chamado de *observer*.

Para iniciar uma conexão com outro dispositivo, o estado da camada de enlace deve ser o *Initiating*. Nesse estado, o dispositivo espera receber pacote de *advertising* do dispositivo com quem ele deseja se conectar. Ao receber a mensagem, o dispositivo envia uma requisição de conexão e depois vai para o estado *Connected*. Dispositivo no estado de *Initiating* é chamado de *initiator*.

O estado de *Connected* pode ser acessado via estado de *Advertising* ou de *Initiating*. Esse é o único estado que utiliza os canais de dados e não os canais de *advertising*, conforme Figura 11. Quando dois dispositivos estão neste estado eles podem transferir entre si pacotes de dados pelos canais de dados e para isso o protocolo utiliza um padrão pseudo-randômico de troca entre canais chamado *Adaptive Frequency Hopping (AFH)*, de forma a diminuir interferências e aumentar robustez na troca de mensagens. Usando de conexão, os dados podem ser enviados, reconhecidos (*acknowledged*) e, se necessário, retransmitidos. Pacotes trocados usando os canais de dados podem ainda ser criptografados e autenticados. Dispositivos que entram no estado *Connected* através do estado *Initiating* são os mestres e aqueles que entram pelo estado de *Advertising* são os escravos. Ao finalizar a conexão, os dispositivos voltam para o estado de *Standby*.

Esses estados são definidos através do uso do serviço GAP (*Generic Attribute Profile*) da camada superior ao controlador, *host*, com a definição de papéis (*roles*) bem definidos para cada dispositivo, como indicado a seguir:

- *Broadcaster*: dispositivo que envia somente eventos de *advertising*;
- *Observer*: dispositivo que somente recebe eventos de *advertising*;
- *Peripheral* (periférico): dispositivo que aceita o estabelecimento de uma ligação via camada física da pilha BLE usando o processo de conexão;
- *Central*: dispositivo que suporta o início de uma conexão física BLE.

2.5.2.1 Interface *host* controlador (HCI)

A interface *host* controlador (HCI) é responsável pela comunicação entre camada controlador e demais serviços presentes na camada de enlace. O *host* e o controlador podem ser implementados tanto no mesmo chip como em chips separados. Em dispositivos mais complexos e maiores, como em *smartphones*, é possível que o controlador esteja em um controlador comunicando através de uma interface como SDIO, UART ou USB. No caso de sensores ou dispositivos menores, de menor consumo de potência, a HCI normalmente não está presente pois todas as camadas estão integradas no mesmo chip, normalmente chamado de SoC (*System on Chip*).

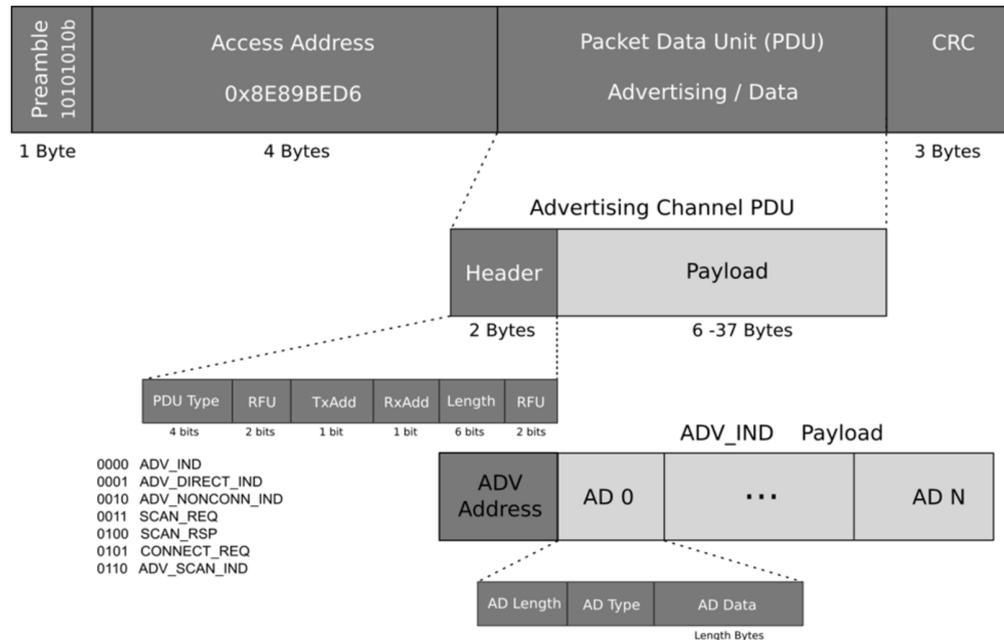
2.5.2.2 Pacotes de *advertising* e de dados

Para enviar dados através de qualquer dos canais do protocolo BLE (de dados ou de *advertising*), pequenos pacotes ou datagramas foram definidos. Esses pacotes encapsulam uma pequena quantidade de dados, trocados em intervalos reduzidos de tempo.

O pacote inicia com uma sequência de treinamento chamada preâmbulo, conforme Figura 13. Esse byte é o suficiente para permitir sincronização entre receptor e transmissor e configurar o controle de ganho automático do rádio. Em seguida, segue um campo de 4 bytes, *Access Address*, que é fixo para a mensagem de *advertising* mas completamente aleatório e privado para pacotes de dados. O campo *Header* é utilizado para descrever o conteúdo do pacote e em seguida aparece o campo *Length* que indica o tamanho do pacote dados, *Data* mostrado na Figura 13. Apesar do campo *Length* ser de 1 byte de tamanho, ele não chega a ser usado completamente, já que o campo *Data* pode ser de no máximo 37 bytes. Esse campo em particular, portanto, é de tamanho variável e carrega dados utilizados

pela aplicação ou pelo *host* da pilha BLE do dispositivo. Finalmente, o campo CRC (*cyclic redundancy check*) de 3 bytes, é usado para garantir que não há erros nos bits do pacote recebido.

Figura 13 – Estrutura do pacote de dados da camada de enlace, com detalhamento para pacote de advertising. [24]



O pacote de *advertising* tem a mesma estrutura do pacote de dados, como citado, no entanto, uma das particularidades desse pacote é o valor fixo do campo *access address* com 0x8E89BED6. Em conjunto com o preâmbulo, esses valores fazem com que o dispositivo realizando *scanning* possa identificar quando outros dispositivos iniciam a transmissão de mensagens de *advertising*. O primeiro dispositivo realiza isso ao armazenar os últimos 40ms e correlacionando os bits recebidos com o preâmbulo e *access address* conhecidos [25].

O campo *header* do pacote de *advertising* indica qual o tipo de pacote está sendo transmitido e cada um desses pacotes tem comportamentos diferentes dentro da pilha BLE, com conjunto de dados em formatos diferentes. Os tipos de pacotes relacionados às atividades de *advertising* disponíveis são ADV_IND (*general advertising indication*), ADV_DIRECT_IND (*Direct connection indication*), ADV_NONCONN_IND (*nonconnectable indication*), ADV_SCAN_IND (*scannable indication*), SCAN_REQ (*active scanning request*), SCAN_RSP (*active scanning*

response) e CONNECT_REQ (*connection request*), sendo estes três últimos relacionados ao *active scanning* [23].

As demais camadas da pilha BLE, *host* e aplicações, com o conjunto de serviços que o compõem, não são relevantes para a descrição do sistema aqui proposto, dessa forma, não serão aqui discutidos.

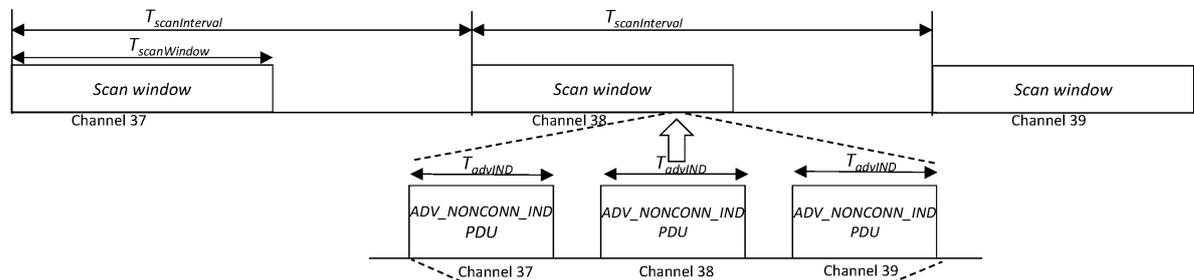
2.6 PARÂMETROS DE COMUNICAÇÃO DE ADVERTISING

O esquema de comunicação de pacotes *advertising* e recepção pelo dispositivo em modo *scanning* segue a representação mostrada na Figura 14.

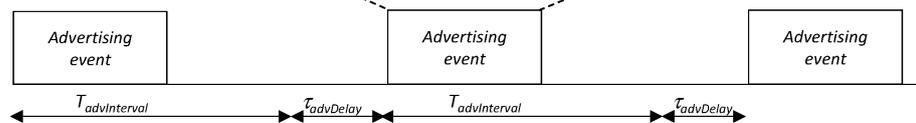
Tanto para o modo de *scanning* em modo ativo como passivo, o estado de *scanning* obedece ao mesmo padrão de comunicação via canais de rádio frequência, escutando em sequência os canais 37, 38 e 39 do padrão BLE. Essa sequência não pode ser modificada, por exemplo, para que o dispositivo somente realize o *scanning* em um desses canais, sendo essa sequência fixa do padrão. Como mostrado na representação da comunicação, os parâmetros de tempo para os eventos de *scanning* são $T_{scanInterval}$ e $T_{scanWindow}$, intervalo de *scan* e janela de *scan*, respectivamente. A cada passagem de $T_{scanInterval}$, o dispositivo está escaneando em um dos canais de *advertising* e no próximo ciclo pula para o canal seguinte. Para evitar consumo de potência excessivo e interferência da operação do rádio com outros processos em andamento na pilha de controle do BLE e reduzir consumo de potência é recomendável, configurar $T_{scanWindow}$ no valor mínimo e ajustar $T_{scanInterval}$ para atingir requisitos de consumo do produto. Para a pilha BLE, o valor máximo configurável de $T_{scanInterval}$ é de 10,24s e o mínimo para $T_{scanWindow}$ é tipicamente 7,5ms. Configurando os dois parâmetros para o mesmo valor, faz com que o rádio esteja ligado 100% do tempo.

Figura 14 – Exemplos de comunicação no estado de advertising, para modos ativo e passivo de scanning. [26]

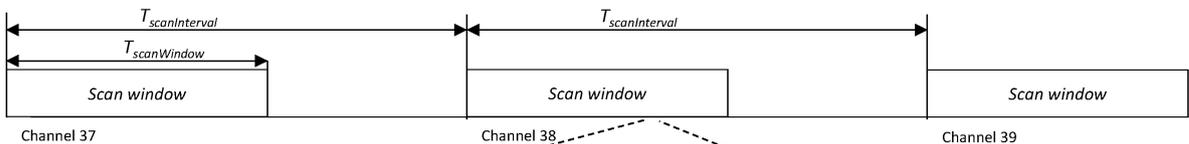
Passive scanner



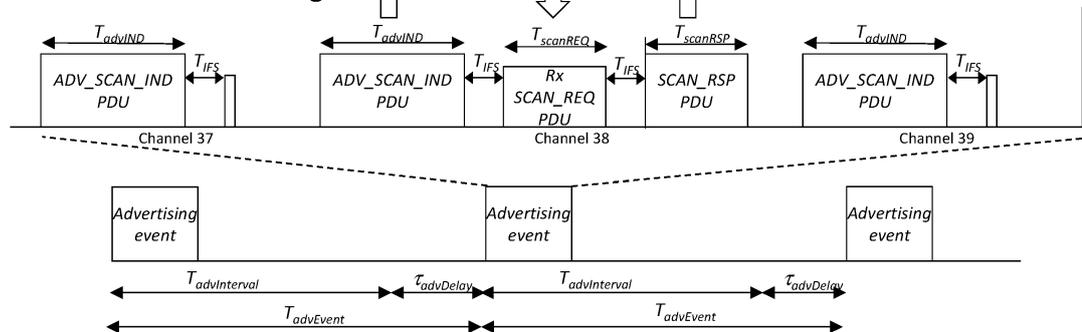
Non connectable undirected advertising



Active scanner



Scannable undirected advertising



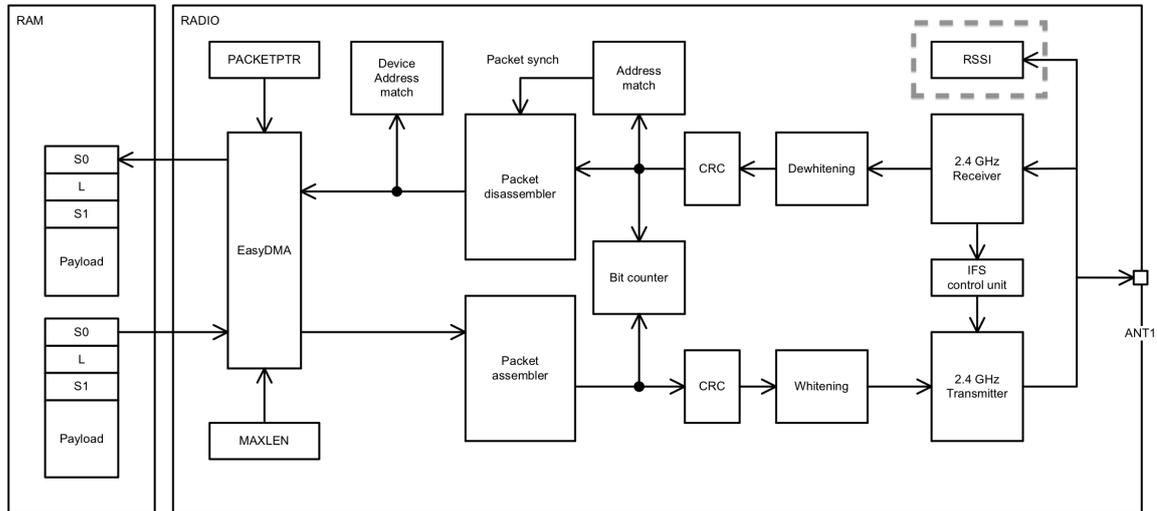
2.7 RSSI

Importante parâmetro para serviços de proximidade e localização utilizando o protocolo BLE, o RSSI (*Received Signal Strength Indicator*) é uma medida da potência presente no sinal de rádio recebido. Dispositivo realizando a função de *scan* pode ler o RSSI de uma mensagem de *advertisement*.

Esse parâmetro não é obtido como informação no pacote de *advertising*, como visto na Figura 13. Ele é informado à aplicação do dispositivo por acesso a registrador de 8 bits (varia de -127 a 20 dBm) que grava o valor medido a partir de

um circuito no próprio chip. Um exemplo é mostrado na Figura 15, do circuito do SoC nRF52832 da Nordic Semi.

Figura 15 – Diagrama de blocos do rádio do SoC nRF52832 com destaque para o módulo RSSI. [27]



Como discutido nas seções anteriores, o protocolo BLE trouxe como inovação a implantação do estado de *advertising*, no qual o dispositivo pode enviar e/ou receber mensagens sem a necessidade de estar conectado a outro dispositivo. Isso possibilitou o uso desse novo recurso para aplicações as mais diversas, como formação de redes de sensores e *beacons* que permitem troca de informações em modo *broadcast*, por exemplo, para monitoramento de temperatura de ambientes ou equipamentos (ou outras grandezas físicas), aplicações em IoT e automação doméstica.

Outra aplicação importante advinda da funcionalidade de *broadcast* do modo de *advertising* foi a de localização e de serviços de proximidade do dispositivo em relação a um conjunto de *beacons* ou receptores BLE [23], como o sistema aqui proposto. No próximo capítulo alguns fundamentos de sistemas de localização utilizando RSSI serão mostrados.

3 SISTEMAS DE LOCALIZAÇÃO

Nesta seção são apresentadas técnicas e parâmetros de localização relacionados ao método conhecido como *fingerprinting*. Fundamento matemático desse método é apresentado de forma sucinta, mostrando o fluxo necessário para obtenção de localização. São apresentadas algumas máquinas de aprendizagem mais comumente utilizadas no processo de localização baseado em *fingerprinting*. O método e as máquinas apresentadas são utilizados posteriormente para desenvolvimento do sistema proposto.

3.1 INTRODUÇÃO

O problema da localização da chave usando sinais de baixa frequência (LF) já é resolvido em aplicações para acesso remoto a veículos como visto na Seção 2. No entanto, há uma nova tendência de integração de veículos a dispositivos *smart* e a tecnologia de comunicação em rádio frequência (UHF e LF) em uso nas chaves de acesso remoto atuais não tem suporte nesses dispositivos. Como visto, o padrão *Bluetooth Low Energy*, lançado no documento do *Bluetooth 4.0*, tem sido utilizado de forma cada vez maior, em grande quantidade de dispositivos e praticamente em todos os novos dispositivos *smart*. Com as características de baixo consumo e alta taxa de troca de mensagens, a tecnologia BLE tem sido foco de pesquisa para uso em sistemas de localização *indoor*.

Sistemas de localização *indoor* são usados para localizar objetos ou dispositivos em ambientes fechados, nos quais sinais do *Global Positioning System* (GPS) não são detectáveis, por depender de linha de visada entre satélite e dispositivo. Além disso, sistemas GPS tem alto consumo e precisão baixa, de até cinco metros. Para aplicações *indoor* e com dispositivos com restrições de bateria, o uso de GPS não é viável. Nesses sistemas, normalmente é desejável precisão menor do que um metro. Nesse caso, o uso de outras tecnologias como base para sistemas de localização *indoor* é necessário. Além da tecnologia BLE, outras são

avaliadas na literatura como Wifi, Zigbee, *Ultra Wideband* (UWB), luz visível, sinais acústicos, ultrassom [28]. Todas essas tecnologias sofrem de algumas características não desejáveis, algumas são a não-linearidade na propagação dos sinais devido a interferências como móveis, paredes, pessoas, causando efeitos de *fading* ou por propagação em caminhos múltiplos. Ainda, em algumas dessas tecnologias, como Wifi e BLE, ocorre interferência na mesma banda de frequência por outros dispositivos, devido a grande densidade de dispositivos em alguns ambientes. Pelas características citadas na Seção 2, a tecnologia BLE foi utilizada no sistema proposto.

Independente da escolha entre estas tecnologias, o problema de decisão da posição precisa do dispositivo não foi resolvido em definitivo. Não há modelo ou sistema consagrado para uso em geral. Alguns dos parâmetros nos sistemas descritos na literatura são: *Angle of Arrival* (AoA), *Time of Arrival* (ToA), *Time Difference of Arrival* (TDoA) e RSSI [29]. Sistemas que usam AoA precisam de um conjunto de antenas para determinar o ângulo de recepção do sinal. Nesse caso, o método utilizado é de triangulação para determinar geometricamente o ângulo do triângulo formado pela recepção de, no mínimo, três receptores ou transmissores. Essas técnicas normalmente requerem conjunto complexo de *hardware* e precisam de calibração de forma que uma posição precisa seja fornecida. ToA é um dos parâmetros mais precisos disponíveis. Através do uso de *clocks* sincronizados, o tempo de propagação do sinal entre transmissor e receptor pode ser obtido. ToA usa o tempo em que o pacote foi enviado com informação enviada no próprio pacote em conjunto com o tempo em que o pacote foi recebido para definir o tempo de propagação do pacote. Para implementar sistemas que usam ToA, no entanto, é necessário *hardware* adicional para sincronizar relógios. TDoA é similar ao ToA no sentido de necessitar de sincronização entre *clocks*. Mas no caso do TDoA, são usados diferentes tempos de propagação de diferentes receptores para calcular um tempo preciso de propagação. RSSI é o método mais popular e mais simples para o cálculo da localização de dispositivos. A razão é que não é necessário nenhum *hardware* adicional e é parâmetro informado em praticamente todas as tecnologias de comunicação sem fio, por isso, muitas vezes é o único parâmetro disponível para realizar localização, a depender da infraestrutura do sistema. O contraponto para o uso do RSSI é a suscetibilidade dos sinais propagados que sofrem influências do ambiente, como destacado, causando variação do RSSI e de maneira não-linear,

sendo difícil encontrar modelo analítico para descrever seu comportamento. Tal modelo teria que ser adequado a cada cenário em que o dispositivo está sendo utilizado.

O uso desses parâmetros define normalmente qual o tipo de base matemática o sistema de localização utilizará e, a partir dessa definição, pode-se definir tipos de sistemas, definidos em [30]. Pode-se falar em métodos de base geométrica, método de minimização de função de custo, método *fingerprinting* e método bayesiano. Os métodos geométricos são usados quando alcances, diferença de alcances ou ângulos são medidas disponíveis, como as já mencionadas TDoA ou ToA. Esses métodos são computacionalmente eficientes mas bastante dependentes das posições dadas às estações de medições e pouco robustos a *outliers* devido a caminhos de propagação que não sejam de visada direta (múltiplos caminhos), principalmente quando se utilizam tecnologias nas quais esse tipo de comportamento é comum, como baseados em sinais de RF. Os métodos baseados na minimização da função de custo, Equação (2) (onde r é o conjunto de medições e a função $h(x)$ contém as posições das estações de medição x), procuram maximizar a probabilidade condicional $p(r \vee x)$. Novamente, para sistemas de propagação baseados em RF, esse método não apresenta robustez pelo impacto da existência de propagação em NLOS (*non line of sight*) em grande escala, fazendo com que a probabilidade de erro p_e chegue a ter mesma ordem de grandeza da função $h(x)$.

$$V(x) = \log p_e(r - h(x)) \quad (2)$$

O método bayesiano pode ser visto como uma extensão do método de *fingerprinting*. Nele, o modelo de observação $p(r \vee x)$ é gerado também na fase de calibração e a probabilidade de erro pode ser incrementada para levar em consideração propagação NLOS, como mostra Equação (3). Esse modelo, portanto, depende fortemente da quantidade de informações obtidas na fase empírica de medição de dados, com separação entre sinais de linha de visada direta e de propagação NLOS.

$$p_e(e) = p_{LOS} N(0, \sigma_r^2) + (1 - p_{LOS}) p_e^{NLOS}(e) \quad (3)$$

Dessa forma, RSSI foi o parâmetro escolhido no sistema proposto como base para sistema de localização e o método de *fingerprinting* foi o utilizado. Neste capítulo, será dado um fundamento para a técnica de localização utilizada,

fingerprinting, e pequena introdução às máquinas de aprendizagem usadas no sistema.

3.2 FINGERPRINTING

Métodos de *fingerprinting* são normalmente utilizados em sistemas de localização *indoor* (IPS – *Indoor Positioning Systems*) baseados em medições de RSSI e consistem de duas fases.

Na fase de calibração, a estação móvel (EM) de teste (aquela que vai ter sua posição definida) é deslocada por um conjunto de posições (x_j que formam o *grid* que deve ser suficientemente denso para cobrir as regiões e gravar os dados de potência de sinal ($\rho_i(x_j)$) das diferentes estações base (EB). Na fase de localização, o sistema captura um novo conjunto de sinais (r_i que são comparados ao conjunto previamente capturado e a posição que mais se aproxima dessa medição é a escolhida como posição estimada pelo sistema. Sistemas baseados no método de *fingerprinting* utilizam técnicas de aprendizagem de máquina (podem também ser chamadas de aprendizagem estatística) para encontrar a estimativa ótima da posição. Não se faz distinção entre medições em relação a sinais vindos de visada direta (LOS – *line of sight*) ou por NLOS e, por isso, apresentam maior robustez quanto a esse tipo de fenômeno em relação aos métodos geométricos ou de minimização da função de custo. Como todos os métodos aqui descritos, o de *fingerprinting* é altamente susceptível a variações no ambiente e ao encontrar um sistema que resolva um problema de localização *indoor* para um ambiente ou situação específicos, não há garantia que esse mesmo sistema será útil se, por exemplo, uma parede seja levantada ou derrubada nesse mesmo ambiente. No entanto, há estudos que indicam que a presença de pessoas no ambiente pode afetar positivamente a assertividade do sistema [31].

Segundo [32], algoritmos baseados em *fingerprinting* associam uma ou várias grandezas a uma determinada posição no espaço (conjunto R). Uma característica específica é escolhida para criação do *fingerprinting*. A característica do sinal é representada por S e pertence ao espaço de características S . A característica do sinal pode ser uma combinação de sinais reais, gerados por múltiplos pontos de captura de sinal. m observações consecutivas da característica do sinal

$S=(S_1, \dots, S_m) \in S^m$ é um vetor aleatório relacionado à posição da EM, u , através da probabilidade condicional $P_{S \vee u}$. *Fingerprints* são construções baseadas em observações S para cada posição do *grid* de captura. Portanto, podemos definir que *fingerprint* é uma função que mapeia $S^m \rightarrow X^n$ e que atribui para as observações S um elemento X chamado de *fingerprint* na posição de captura u .

Como a construção do mapa de posições para a técnica de *fingerprint* é uma escolha estritamente empírica, uma primeira preocupação é a definição dessas posições e como ela pode afetar a robustez da técnica. Seja, por exemplo, que para duas localizações u_1 e u_2 , as probabilidades $P_{X \vee u_1}$ e $P_{X \vee u_2}$ são muito semelhantes. A semelhança entre as probabilidades é dada pela métrica d . O requisito de robustez dos *fingerprints* implica que quando $d(P_{X \vee u_1}, P_{X \vee u_2}) \leq L$, onde d é a métrica e $L > 0$, então u_1 e u_2 também devem estar próximos, com $\|u_1 - u_2\| \leq s(L)$, onde $s(L) \rightarrow 0$ para $L \rightarrow 0$. Essa relação pode ser usada para definir o grau de robustez do *fingerprint*. Através de demonstração conduzida em [32], no caso de *fingerprinting* baseado em RSSI, o *grid* que apresenta melhor robustez é aquele que minimiza a distância euclidiana entre as posições de captura mantendo a cobertura da área maximizada. Além disso, outra característica importante para levantamento dos *fingerprints* é sua invariância ao longo do tempo, ou seja, realizando duas capturas em momentos distintos, manter-se as probabilidades $P_{X \vee u_1}^{t_0}$ e $P_{X \vee u_1}^{t_1}$ semelhantes.

Após a seleção da característica e da função de criação do *fingerprint* realizada, a próxima etapa é a definição do algoritmo de *fingerprinting*. Primeiro passo é a criação da base de dados, consistindo na criação de pares de localização e seus *fingerprints*. Essa base de treinamento pode ser construída a partir de medições extensivas, baseados em simulações ou combinação de ambos. Essa fase é a chamada fase de treinamento. O conjunto de posições de treinamento é denominado de *grid* de treinamento, denotado por Λ , onde $\Lambda \subset R^d$, d normalmente maior ou igual a 3. Esse *grid* pode ser definido matematicamente por coordenada em relação a um ponto fixo ou de forma não-uniforme. Depois de escolhidos os pontos de captura na fase de treinamento, uma característica específica é escolhida para criar o *fingerprint*. A base de dados é criada medindo-se a característica do sinal S nas posições de treinamento dentro do *grid* Λ . A cada localização v , são

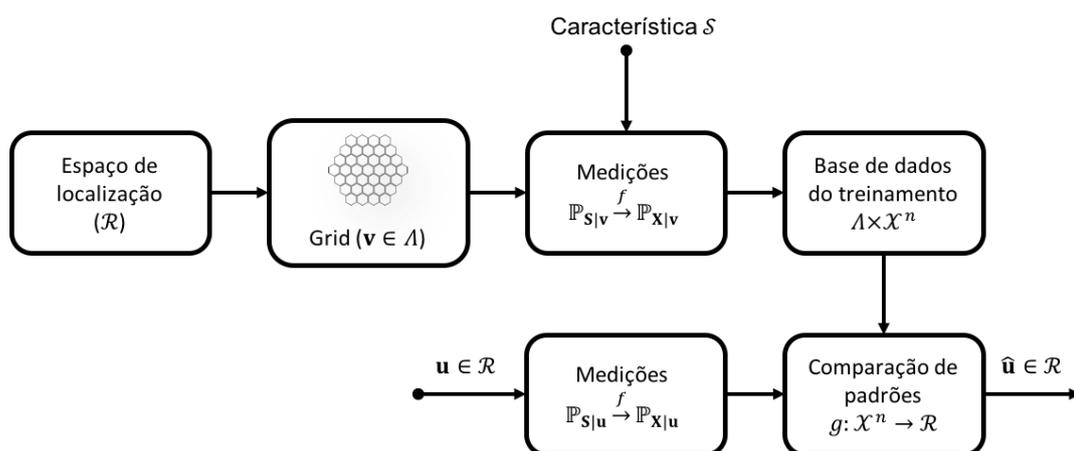
realizadas múltiplas capturas e os *fingerprints* vão sendo construídos. A base de dados D , que consiste de pares (v, X) é um subconjunto de $\Lambda \times X^n$.

Em seguida, o algoritmo consiste em estimar a localização de um ponto numa localização u , ou seja, essa é a fase da localização. Nessa fase do algoritmo, há m' medições nesse ponto, onde se cria um *fingerprint* $X_u \in X^{n'}$. A quantidade de medições na fase de treinamento, n , e na fase de localização, n' , normalmente são diferentes. Após a aquisição dos *fingerprints*, uma função de comparação de padrões g é utilizada para estimar a posição \hat{u} baseado no *fingerprint* X_u obtido e nos *fingerprints* da base de dados da fase de treinamento. Essa função g normalmente é uma máquina de aprendizagem, seja *kNN*, árvore de decisão ou rede neural. Todas elas procuram minimizar o erro $\|u - \hat{u}\|$.

A quantidade de medições necessárias para ter-se uma base que represente adequadamente a região no espaço onde será aplicado o algoritmo de localização não obedece a uma regra conhecida, mas sim a fatores como tempo disponível para realização dos experimentos, quantidade de memória do *hardware* utilizado, entre outros fatores, como observação da base gerada através de representações gráficas dos parâmetros medidos.

Um diagrama de blocos do algoritmo baseado no método de *fingerprint* é mostrado na Figura 16.

Figura 16 – Diagrama de blocos de algoritmo baseado em fingerprinting. [32]



Nas próximas seções, algumas máquinas de aprendizagem, utilizadas nos experimentos para testes do sistema proposto, serão detalhadas.

3.3 MÁQUINAS DE APRENDIZAGEM

Máquinas de aprendizagem buscam por padrões em quaisquer conjuntos de dados fornecidos e tem tido cada vez mais aplicações em diferentes campos do conhecimento. Eles automaticamente aprendem com os dados generalizando a partir de exemplos dados. À medida que mais dados são disponibilizados e com melhor qualidade e representatividade do problema a ser resolvido, mais complexas são as tarefas para as quais as máquinas de aprendizagem são utilizadas [33]. Os algoritmos são normalmente implementados em duas fases. Na primeira, chamada de fase de treinamento, como discutido rapidamente na seção anterior, dados são coletados, rotulados e fornecidos ao algoritmo, de forma que ele possa apreender os padrões contidos no conjunto e criar um modelo para classificar os dados ou estimar propriedades dos mesmos. Na segunda fase, chamada de fase de teste, um novo conjunto de dados é testado usando o modelo construído na fase de treinamento e a efetividade do modelo é obtida. Algoritmos de aprendizagem nos quais essas duas fases estão presentes são chamados algoritmos de aprendizagem de máquina supervisionados. Algoritmos que não se utilizam da fase de treinamento são chamados de não supervisionados. Eles usam dados não rotulados para formar *clusters* de dados divididos em diferentes classes. Máquinas de aprendizagem podem ser usados para classificação ou regressão. Na classificação, o algoritmo faz a divisão dos dados em diferentes classes, atribuindo um rótulo ao dado testado, enquanto que na regressão se estima valores contínuos através de aprendizagem de um conjunto de dados. A seguir, serão apresentados três tipos de máquinas de aprendizagem utilizadas nos experimentos do sistema proposto.

3.3.1 *k*-Nearest Neighbors

O algoritmo de *kNN* é um algoritmo simples não-paramétrico e baseado em instâncias, porém robusto e versátil que normalmente é utilizado como *benchmark* para classificadores mais complexos como redes neurais ou SVM (*Support Vector Machines*). Como um algoritmo de aprendizagem supervisionada, de forma que dado um conjunto de dado de entrada e correspondente classe, (x, y) , o algoritmo retorne a relação entre x e y , $h: X \rightarrow Y$, de forma que de um dado x , a função $h(x)$ possa retornar y correspondente.

Métodos não-paramétricos não fazem nenhuma suposição acerca do formato da função $h(x)$, evitando o erro no modelamento da relação entre classes e conjunto de dados. Baseado em instâncias pois não diretamente aprende um modelo, ou seja, o algoritmo grava na memória todo ou parte do conjunto de dados de treinamento e quando solicitado a decidir sobre um dado de teste, usa os dados armazenados para informar sua estimativa. O algoritmo kNN pode ser usado tanto para regressão como para classificação. É um algoritmo supervisionado, mas com uma fase de treinamento muito reduzida, portanto a maior parte do esforço neste método é dispendida na fase de teste.

Na fase de teste, o algoritmo kNN essencialmente usa um sistema de voto majoritário entre os k dados mais semelhantes ou próximos do novo dado observado. Essa similaridade é definida matematicamente como uma métrica de distância entre os pontos. No caso do sistema proposto foi utilizada a distância euclidiana, Equação (4), mas outras métricas podem ser utilizadas com de Manhattan, Minkowski, Chebyshev ou Hamming.

$$d(x, x') = \sqrt{\sum_{i=1}^n (x_i - x'_i)^2} \quad (4)$$

Dado um valor inteiro positivo k , um novo dado observado x e uma métrica de similaridade d , o classificador kNN irá realizar os seguintes passos:

- Calcula a métrica d entre o novo dado x e todos os dados da base. A partir desses valores escolhe os k pontos da base de menor métrica, portanto mais próximos de x , formando o conjunto A . Usualmente escolhe-se k ímpar para evitar empate na decisão;
- Em seguida, calcula a probabilidade condicional de cada classe representada pelos pontos do conjunto A , seguindo a seguinte relação da Equação (5), onde ω_m é a classe de uma das k amostras. Nesse caso as classes do k são variáveis aleatórias que assumem os valores ω_i com probabilidades $P(\omega_j|x), j=1,2,\dots$

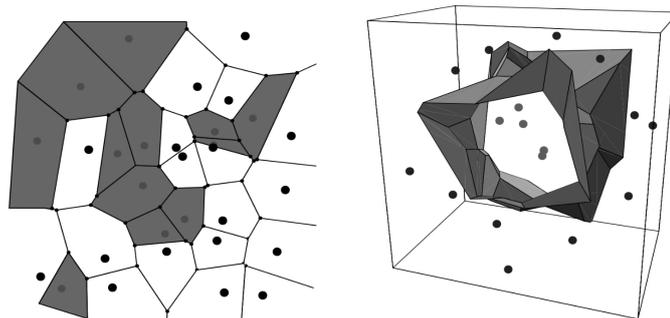
$$\sum_{i=(k+1)/2}^k \binom{k}{i} P(\omega_m \vee x)^i [1 - P(\omega_m|x)]^{k-i} \quad (5)$$

Normalmente, se a maioria das k amostras apresentarem como classe ω_m , então maior a probabilidade de ser a classe escolhida pelo classificador.

Uma outra maneira de entender o algoritmo *kNN* é que ele calcula regiões delimitadas que são usadas para classificar novos pontos a partir da base de dados. Para o caso em que o algoritmo tem $k=1$, podemos ver na Figura 17, as regiões delimitadas no espaço da base de dados a partir da regra de decisão da Equação (6).

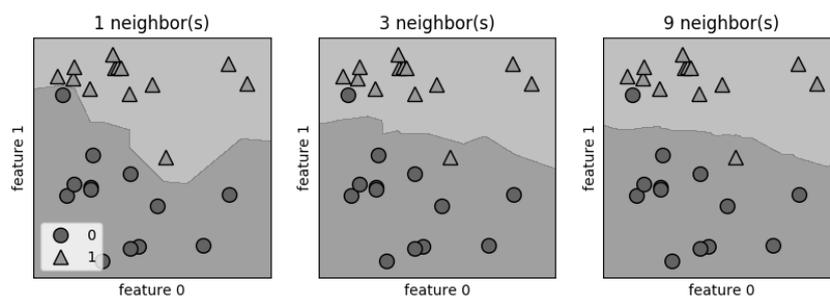
$$P(\omega_m|x) = \max_i P(\omega_i \vee x) \quad (6)$$

Figura 17 – Regiões de Voronoi nos espaços bi e tridimensional.



A escolha do hiper-parâmetro k pelo projetista da máquina de aprendizagem depende basicamente de como o conjunto de dados se comporta com a distribuição das classes. Então, escolhe-se k de forma que o classificador melhor se adeque ao conjunto de dados. Pode-se pensar nesse parâmetro como definindo o formato das regiões de decisão mostradas anteriormente.

Figura 18 – Exemplo de classificação com *kNN*.



A Figura 18 apresenta o mesmo conjunto de dados sendo classificados pelo método *kNN* com diferentes valores de k . Vê-se que para um k pequeno o limite da região de decisão representa melhor os dados, sem viés, mas apresenta maior variação. Para k cada vez maior, o classificador é mais robusto a *outliers*, apresenta limite da região de decisão cada vez mais suavizado, com menor variância, mas com classificação mais enviesada, gerando maior erro de classificação.

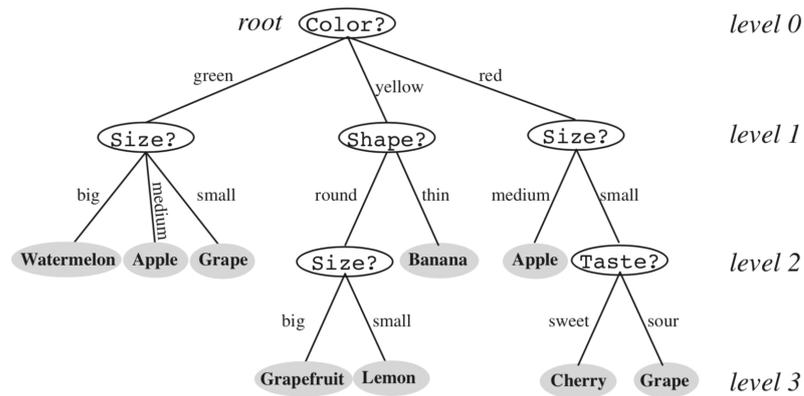
3.3.2 Árvore de decisão

Árvores de decisão são algoritmos de aprendizagem supervisionada, não-paramétricos usados tanto para classificação como para regressão. O objetivo é a criação de um modelo de decisão para prever o valor de uma variável através da aprendizagem de regras de decisão simples inferidas do conjunto de dados. Essas regras de decisão comumente são na forma de declarações do tipo *if-then-else*.

São algoritmos que não necessitam da noção de métrica para criar regras de decisão. Ou seja, serve muito para conjuntos de dados onde o conceito de métrica de similaridade não se aplica, por exemplo, na classificação de grupos de animais ou de características similares entre mamíferos. São os chamados conjuntos de dados nominais. Então o que se usa como entrada para esse algoritmo são listas de atributos para os dados, usando valores específicos de um conjunto fixo de propriedades em uma *d-tupla*. Por exemplo, para classificar frutas usando os seguintes atributos ou propriedades: cor, textura, paladar e tamanho. Tem-se assim uma 4-tupla para usar como classificação de frutas.

A árvore de decisão tem a propriedade clara a de legibilidade, de ser facilmente interpretável pela forma gráfica com que pode ser representada. Funciona como uma série de perguntas (normalmente de resposta binária), representadas como uma árvore com nós e ramificações, como mostrado na Figura 19. O primeiro nó ou nó raiz é sempre colocado no topo da árvore e conectado aos demais nós por ramificações. Novos nós são conectados a outros níveis abaixo até os últimos nós ou folhas da árvore. Essas ramificações são geradas a partir de perguntas sobre o valor de certa propriedade do conjunto de dados. Diferentes nós são ligados ao nó superior a depender da resposta sobre essas propriedades e segue-se a essas perguntas até finalizarem as propriedades ou que os ramos não sejam mais mutuamente distintos.

Figura 19 – Exemplo de árvore de decisão. [33]



Para definir uma regra de crescimento da árvore de decisão, uma metodologia genérica chamada CART (*Classification and Regression Trees*) é utilizada. CART fornece um método geral que pode ser instanciado de diferentes formas para gerar diferentes tipos de árvores de decisão. A estrutura do CART funciona a partir de seis perguntas genéricas:

1. Propriedades restritas a respostas binárias ou multi-valores? Ou seja, quantas ramificações permitidas a cada nó?
2. Qual propriedade deve ser testada em determinado nó?
3. Quando um nó deve ser declarado uma folha?
4. Se a árvore crescer demais, como seria possível torná-la menor, ou seja, ser podada?
5. Se uma folha é impura, como a categoria desse nó deveria ser nomeada?
6. Como dados não categorizados devem ser tratados?

Outros algoritmos existem para criação de árvores de decisão, como ID3, C4.5, MARS, CHAID (*CHI-squared Automatic Interaction Detector*) e *Conditional Inference Trees*. Todos operam, no entanto, sob o conceito de maximização do ganho de informação na escolha da propriedade a ser testada em determinado nó. Em última instância, o que se pretende é escolher decisões (e propriedades sobre as quais decidir) que levem a uma árvore de decisão com o menor número de nós possível, mais compacta, seguindo o princípio da navalha de Occam de que o modelo mais simples a descrever um conjunto de dados é preferível.

Para definir o conceito de ganho de informação é preciso descrever o conceito de impureza do nó. São várias as métricas usadas para o cálculo da impureza, mas todas possuem o mesmo comportamento. Seja $i(N)$ a impureza do nó N . Para todos os casos, o valor de $i(N)$ deve ser zero se todos os dados que chegam àquele nó possuem o mesmo valor para a propriedade e valor alto se propriedades são igualmente representadas. A métrica mais comum é a entropia, dada pela Equação (7), onde $\hat{P}(x \in \omega_j | N)$ é a fração de dados no nó N que estão na classe ω_j .

$$i(N) = - \sum_j \hat{P}(x \in \omega_j | N) \log_2 \hat{P}(x \in \omega_j | N) \quad (7)$$

Nesse caso, como esperado do conceito de entropia, se todas as amostras do nó pertencerem à mesma classe, então a impureza é zero, caso as classes presentes naquele nó estejam igualmente representadas, o valor da impureza será o mais alto.

Outra definição de impureza é a variância, útil para o caso de somente duas classes, dada pela Equação (8). A generalização desse conceito é o índice de Gini, dado pela Equação (9). Essa métrica apresenta um pico de valor maior para igualdade na representação das classes do que a entropia.

$$i(N) = \hat{P}(x \in \omega_1 | N) \hat{P}(x \in \omega_2 | N) \quad (8)$$

$$i(N) = \sum_{i \neq j} \hat{P}(x \in \omega_i | N) \hat{P}(x \in \omega_j | N) = 1 - \sum_j \hat{P}^2(x \in \omega_j | N) \quad (9)$$

Há ainda a métrica de *misclassification* dada pela Equação (10). Essa métrica é a que apresenta maior pico para o caso de representação igual para classes presentes no nó, mas possui uma derivada descontínua, que pode dificultar na busca por uma decisão ótima sobre uma variável contínua.

$$i(N) = 1 - \max_j \hat{P}(x \in \omega_j | N) \quad (10)$$

A partir destas métricas, pode-se definir que a escolha da propriedade a ser testada é feita escolhendo aquela que provoca a maior diminuição possível na impureza do nó. Essa relação pode ser expressa na Equação (11), onde N_L e N_R

são os nós descendentes da esquerda e direita e P_L é a fração de amostras do nó N que irão para o nó N_L quando uma propriedade T for testada, sob o valor s . Essa é uma definição para o ganho de informação obtido da escolha da propriedade T . Então, o melhor valor de s para T é aquele que maximiza $\Delta i(T)$.

$$\Delta i(s) = i(N) - P_L i(N_L) - (1 - P_L) i(N_R) \quad (11)$$

Para o caso de árvores não binárias, permitindo mais de duas ramificações do mesmo nó, pode-se escalonar essa relação para a mostrada na Equação (12), chamada de razão de ganho (*gain ratio impurity*), com P_k sendo a fração de

amostras sendo ramificadas para o nó N_k e $\sum_{k=1}^B P_k = 1$. Portanto, a melhor divisão para o nó é aquela que maximiza $\Delta i_B(s)$.

$$\Delta i_B(s) = \frac{\Delta i(s)}{-\sum_{k=1}^B P_k \log_2 P_k} \quad (12)$$

A escolha de quando parar a divisão nos nós e transformá-los em folhas é empírica e pode ser feita definindo parâmetros de parada, como número mínimo de amostras naquele nó ou número máximo de níveis da árvore ou, ainda, ganho de informação mínimo para criação de novo nó. Essa escolha é importante, pois no pior caso, a árvore pode crescer indefinidamente até que cada nó tenha somente uma amostra. No outro caso, de parar o crescimento da árvore muito cedo, o erro de classificação pode ser alto, com a árvore não classificando o conjunto de dados de forma adequada. A escolha de como a árvore deve crescer pode ser feita automaticamente usando técnicas como *cross validation* (validação cruzada), onde uma parte das amostras (90%, por exemplo) é usada para treinamento e o restante usado para testes.

Árvores de decisão apresentam acurácia como classificadores comparável a outros métodos, especialmente em problemas nos quais informações prévias sobre qual a melhor maneira de classificar os dados não está clara. Como no método *kNN*, é de fácil interpretação como esse classificador opera e importante para ganhar conhecimento e entender inicialmente os padrões que regem o conjunto de dados.

3.3.3 Redes neurais

Considere um problema de aprendizagem supervisionado no qual tem-se acesso a um conjunto de exemplos classificados $(x^{(i)}, y^{(i)})$. Redes neurais apresentam um método de construção de hipóteses $h_{\omega,b}(x)$ complexas e não-lineares, onde através dos parâmetros ω e b pode-se representar os exemplos.

A unidade básica na construção das redes neurais artificiais é chamada de *perceptron*, que é a rede neural mais simples possível e está representada na Figura 20. Então, *perceptrons* são unidades computacionais que recebem dados de entrada

x_1, x_2, \dots, x_m e calcula na saída $h_{\omega,b}(x) = f(\omega^T x) = f\left(\sum_{i=1}^m \omega_i x_i + bias\right)$, onde $f: R \rightarrow R$ é chamada de função de ativação, na Figura 20, a função de ativação representada é a função *step*.

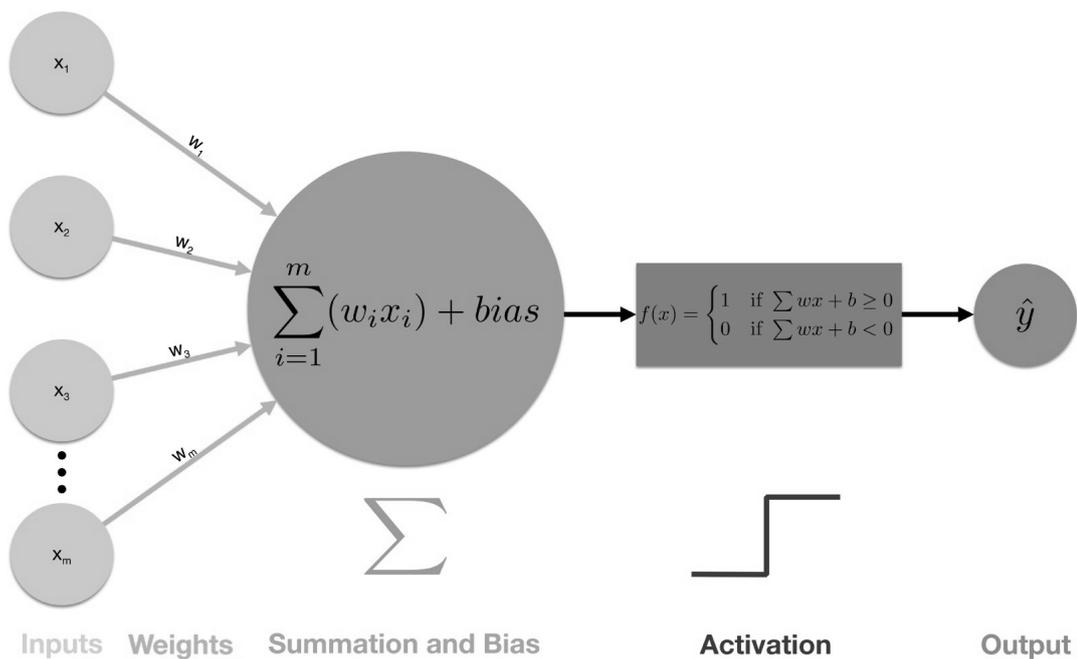
Seguindo as indicações da Figura 20, pode-se descrever a estrutura do *perceptron* como segue:

- 1) Os dados x_1, x_2, \dots, x_m são as entradas do *perceptron*, carregando informações;
- 2) Os valores $\omega_1, \omega_2, \dots, \omega_m$ são os pesos de cada sinapse do *perceptron*, que multiplicam os dados de entrada;
- 3) Etapa que adiciona um valor pré-determinado aplicando viés ao conjunto de entrada, *bias*;
- 4) Aplicação da função de ativação ao termo $\sum_{i=1}^m \omega_i x_i + bias$. A função de ativação representada é a função *step*, mas outras funções de ativação utilizadas são sigmóide, tangente hiperbólica (*tanh*), função linear retificada (*ReLU*) etc;
- 5) Sinal de saída \hat{y} .

O *perceptron* ou rede neural de camada única opera sob a escolha da função de ativação, dos pesos ω_i e do termo *bias*, que será definido como b . Uma das principais limitações que impossibilitaram a rápida adoção do *perceptron* foi o fato dele, usando a função *step*, não ser capaz de aprender a problemas do tipo

resolvidos pela porta XOR. Mais tarde, através da criação das redes neurais de múltiplas camadas (ou *Multilayer Perceptrons* – MLP), esses classificadores foram capazes de resolver o problema da porta XOR, usando o método de *backpropagation* (retropropagação) para treinamento da rede, de forma que a rede aprende a partir de tentativa e erro, corrigindo os valores dos pesos de forma a minimizar a diferença entre a saída inicialmente calculada e aquela desejada, a partir da comparação com o conjunto de dados. Além disso, as funções de ativação hoje mais adotadas são a função sigmoide (Equação (13)), tangente hiperbólica (Equação (14)), que é usada como uma forma de reescalonamento da função sigmoide, e a ReLU (mais utilizada em *deep learning*). As duas primeiras funções são mais utilizadas em MLP por não apresentarem descontinuidades (portanto, deriváveis), de modo que não variação abrupta na saída devido a pequenas mudanças nos pesos, durante o treinamento da rede.

Figura 20 – Representação do perceptron. [34]



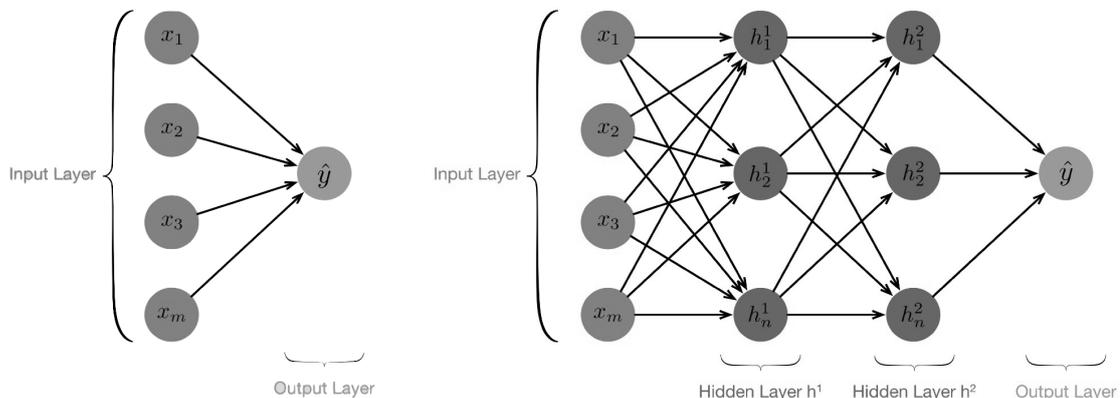
$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (13)$$

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \quad (14)$$

Uma MLP é uma rede neural formada conjugando perceptrons, onde a saída de um serve como entrada para outro. Seguindo o exposto na Figura 21, usando a representação da direita, tem-se a camada mais à esquerda como a camada de entrada e a camada mais à direita como camada de saída, que pode ser múltipla. Os nós intermediários compõem as chamadas camadas escondidas (hidden layers). Portanto, pode-se dizer que a rede neural representada possui m dados de entrada (x_1, x_2, \dots, x_m) e as camadas escondidas são representadas por h . Importante notar que cada nó h_j^i das camadas escondidas são perceptrons, seguindo a estrutura da Figura 20.

Figura 21 – Esquerda: representação simplificada de um perceptron, a rede neural mais simples.

Direita: MLP com duas camadas escondidas. [34]



Para facilitar a compreensão, vamos supor que a rede neural possui somente uma camada escondida. Então, pode-se descrevê-la a partir do conjunto de parâmetros $(\omega, b) = (\omega^{(1)}, b^{(1)}, \omega^{(2)}, b^{(2)})$, onde $\omega^{(l)}$ é formado pelos pesos $\omega_{ij}^{(l)}$ associados à unidade j da camada l , conectados à unidade i da camada $l+1$. E, da mesma forma, $b^{(l)}$ é formado pelos *bias* $b_i^{(l)}$ associados à unidade i da camada $l+1$. Nesse caso, $\omega^{(1)} \in R^{n \times m}$ e $\omega^{(2)} \in R^{1 \times n}$. A rede neural calcula para cada nó da camada escondida a saída da função de ativação correspondente, que torna-se entrada para o *perceptron* da camada de saída, como mostrada na Equação (15), $a_j^{(l+1)}$ é a saída do nó $h_j^{(l)}$.

$$a_j^{(2)} = f \left(\sum_{i=1}^m (\omega_{ji}^{(1)} x_i) + b_j^{(1)} \right) \quad (15)$$

Então, a saída da rede neural é apresentada na Equação (16). Essa propriedade da rede neural de propagação das entradas através das camadas

escondidas até a saída é chamada de *forward propagation*. Organizando os parâmetros em matrizes e usando operações vetorial de matrizes, pode-se usar rotinas de álgebra linear para otimizar o cálculo da rede.

$$\hat{y} = f\left(\sum_{i=1}^n (\omega_{1i}^{(2)} a_i^{(2)}) + b_1^{(2)}\right) = f\left(\sum_{j=1}^n \left(\omega_{1j}^{(2)} f\left(\sum_{k=1}^m (\omega_{jk}^{(1)} x_k) + b_j^{(1)}\right)\right) + b_1^{(2)}\right) \quad (16)$$

A partir do formato de rede neural apresentado, pode-se montar variadas arquiteturas ou topologias de redes neurais com múltiplas camadas escondidas com diferentes quantidades de nós e com variados números de conexões entre eles. Para uma rede neural com l camadas, as equações apresentadas são generalizadas para calcular as saídas da primeira camada que alimentarão os nós da segunda camada e assim por diante, até a camada de saída. Esse tipo de rede neural, onde não há ciclos, retornos ou *loops* entre as camadas é chamada de rede neural que usa *feedforward*.

3.3.4 Sistema proposto

Então, o sistema proposto foi baseado no método de *fingerprinting*. Ou seja, foi desenhado um grid de pontos na região em torno do veículo e no seu interior para realizar medições de RSSI e montar uma base de dados como discutida neste capítulo. A base de dados, com os pontos sendo agrupados nas regiões do espaço, é usada como entrada para os classificadores. Eles são escolhidos para uso na fase de treinamento, na qual os parâmetros da máquina de aprendizagem são calculados para posterior uso na fase de testes ou de localização propriamente dita. Os classificadores aqui apresentados foram utilizados na máquina de aprendizagem proposta na Seção 5. Como será mostrado, uma arquitetura de dois níveis foi proposta, na qual esses classificadores foram utilizados como blocos na sua construção. Suas performances, nesse caso, com a acurácia sendo o parâmetro mais relevante, foram comparadas para definir qual desses classificadores seriam usados.

4 TRABALHOS RELACIONADOS

Nesta seção serão discutidos artigos e patentes relacionados ao tema do sistema proposto nesta dissertação. Os artigos serão analisados comparativamente segundo tecnologias utilizadas, consumo de potência dos dispositivos, alcance em metros do sistema de localização e vantagens e desvantagens comparativas, como complexidade da implementação, imunidade a ruído, custos de *hardware* e penetração no mercado. Foram escolhidas algumas patentes relacionadas para apresentar um panorama de como está o mercado de pesquisa e desenvolvimento na área de sistemas de acesso veicular, sem análise comparativa em relação ao sistema proposto.

4.1 TECNOLOGIAS, PARÂMETROS E SISTEMAS DE LOCALIZAÇÃO *INDOOR*

Sistemas de localização *indoor* (*Indoor Positioning Systems – IPS*) se tornaram uma tendência de pesquisa global com o advento de dispositivos vestíveis (*smartwatches*, *smartbands* etc) e da proliferação do uso de *smartphones* dotados das mais variadas tecnologias de comunicação e posicionamento. Além de diferentes tecnologias, os sistemas de localização também variam conforme uso de parâmetros de localização disponíveis nessas tecnologias de comunicação.

Algumas tecnologias disponíveis e em uso em sistemas de localização *indoor* são mostradas na Tabela 1, com tecnologias levantadas em resumo realizado em [28]. Algumas características de vantagens e desvantagens foram adicionadas para levar em consideração possibilidade de uso de dispositivos eletrônicos de uso pessoal de grande uso no mercado.

Tabela 1 – Tabela comparativa de tecnologias de localização *indoor*.

TECNOLOGIA	ALCANCE MÁXIMO	CONSUMO	VANTAGENS	DESVANTAGENS
IEEE 802.11	35 m <i>indoor</i>	Moderado	Grande disponibilidade, alta precisão, não necessita de <i>hardware</i> adicional, presente em larga escala	Vulnerável a ruído, necessário algoritmo complexo
UWB	10-20 m	Moderado	Imune a interferências, apresenta alta precisão	Alcance reduzido, necessário <i>hardware</i> adicional para dispositivos presentes no mercado, alto custo
Acústico	Poucos metros	Baixo a moderado	Pode ser usado em aplicações personalizadas, pode fornecer alta precisão	Afetado por poluição sonora no ambiente, necessária infraestrutura adicional
RFID	200 m	Baixo	Baixo consumo de potência e grande alcance	Precisão baixa na localização e necessário <i>hardware</i> adicional
Bluetooth	100 m	Baixo	Alta taxa de transferência, alcance alto de recepção, baixo consumo de potência, presente em larga escala	Baixa precisão de localização, vulnerável a ruído
Luz visível	1,4 km	Alto	Pode ter alta disponibilidade, potencial de alta precisão, não vulnerável a múltiplos caminhos	Comparativamente maior consumo, alcance afetado por obstáculos, necessária visada direta
Ultrassom	10 a 100 m	Baixo a moderado	Comparativamente menor absorção de sinal	Alta dependência no posicionamento de sensores, necessário <i>hardware</i> adicional
Sigfox	50 km	Muito baixo	Grande alcance, baixo consumo de potência	Distância elevada entre estação base e dispositivo, grande atenuação pela presença de paredes no caminho do sinal, necessário <i>hardware</i> adicional
LoRA	15 km	Muito baixo	Grande alcance, baixo consumo de potência	Distância elevada entre estação base e dispositivo, grande atenuação pela presença de paredes no caminho do sinal, necessário <i>hardware</i> adicional

[28] também apresenta resumo dos parâmetros usados atualmente em sistemas de localização, com a Tabela 2 mostrando suas vantagens e desvantagens.

Tabela 2 – Tabela comparativa dos parâmetros usados em sistemas de localização *indoor*.

PARÂMETRO	VANTAGENS	DESVANTAGENS
RSSI (<i>Received Signal Strength Indicator</i>)	Fácil implementação, de baixo custo, pode ser usado na maioria das tecnologias	Vulnerável a efeitos de múltiplos caminhos de propagação, como desvanecimento, ruído no ambiente, baixa precisão e necessário realizar <i>fingerprinting</i>
CSI (<i>Channel State Information</i>)	Robusto a múltiplos caminhos e ruído	Não facilmente disponível na maioria dos <i>hardwares</i>
AoA (<i>Angle of Arrival</i>)	Pode fornecer alta precisão de localização, não necessário fazer <i>fingerprinting</i>	Pode ser preciso uso de antenas direcionais e <i>hardware</i> de maior complexidade, como algoritmos mais complexos e performance se degrada com aumento da distância entre dispositivos
ToF (<i>Time of Flight</i>)	Pode fornecer alta precisão de localização, não necessário fazer <i>fingerprinting</i>	Necessária sincronização de tempo entre transmissor e receptor, uso de dados de tempo presentes nas mensagens, múltiplas antenas nos dispositivos e visada direta é obrigatória para boa performance
TDoA (<i>Time Difference of Arrival</i>)	Não necessário fazer <i>fingerprinting</i> , não necessário realizar sincronização de <i>clock</i> entre dispositivo e base	Necessária sincronização de tempo entre transmissor e receptor, uso de dados de tempo presentes nas mensagens
RToF (<i>Return Time of Flight</i>)	Pode fornecer alta precisão de localização, não necessário fazer <i>fingerprinting</i>	Necessária sincronização de tempo entre transmissor e receptor, tempo de processamento pode afetar performance para curtas distâncias
PoA (<i>Phase of Arrival</i>)	Can be used in conjunction with RSS, ToA, TDoA to improve the overall localization accuracy	Performance degradada sem visada direta
<i>Fingerprinting</i>	Fácil implementação e uso	New fingerprints are required even when there is a minor variation in the space

Dentre essas tecnologias, aquelas presentes em larga escala em dispositivos eletrônicos de uso pessoal são *Bluetooth* e *Wifi* (IEEE 802.11). Considerando características como disponibilidade da tecnologia em diferentes dispositivos, precisão, alcance, custo, escalabilidade, consumo de potência e segurança, características de performance sugeridas por [35], a tecnologia BLE foi escolhida para o desenvolvimento do sistema aqui proposto. [29] apresenta estudo comparativo onde sob as mesmas configurações de transmissão, potência em -10dBm e intervalo de transmissão de 0,5s. Nesse estudo, o erro médio obtido com

Wifi foi de 88% do obtido com BLE, mas potência média consumida pelo dispositivo Wifi foi de 217mW e para BLE 0,367mW. O consumo de Wifi inviabiliza seu uso em dispositivos à base de baterias disponíveis hoje para dispositivos vestíveis e, portanto, para chaves veiculares. Alguns estudos [42][62] apresentam a possibilidade de uso de BLE como tecnologia de acesso veicular em particular.

Sistemas de localização baseados em BLE podem usar como parâmetros RSSI, AoA e ToF, mas comumente esses sistemas utilizam o RSSI como parâmetro básico, deixando assim os sistemas mais simples, com alcance de até 100 metros e baixo consumo de potência. A seguir, alguns sistemas de localização baseados em BLE ou Wifi e RSSI apresentados na literatura são discutidos.

[36] apresenta um sistema de localização e rastreamento baseado em BLE e RSSI cuja contribuição é a apresentação de um modelo de observação desenvolvido especialmente para *fingerprinting* de RSSI. Esse modelo é uma interpolação baseada na métrica de distância de Wasserstein (AWC). Esse modelo é usado para interpolar sinais de RSSI mapeados no ambiente testado, de forma que o conjunto de pontos nos quais o sinal é capturado seja reduzindo, diminuindo tempo e esforço da etapa de *fingerprinting*. O ambiente de teste é uma sala de estar, com múltiplos *beacons* espalhados no espaço e dispositivo de teste a 1,5m de altura com seus sinais sendo capturados a cada ponto definido. Para teste do modelo, foram feitos mapeamentos com 50, 32, 21, 15 e 8 pontos. E para comparação do modelo de observação proposto, foram testados os modelos para rastreamento *Nearest Fingerprint* (NF), *k-Nearest Fingerprint Combination* (kNF) e *Artificial Neural Network* (ANN). A Tabela 3 mostra os erros médios para cada modelo testado, variando os pontos do mapa de rádio obtido. Vê-se que para maior densidade de pontos capturados menor o erro de predição associado e o modelo mais simples, NF, apresenta bom desempenho, por vezes sendo o ótimo diminuindo-se a densidade do *grid*.

Tabela 3 – Comparativo de erro médio em metros entre modelos de [36].

	50 PONTOS	32 PONTOS	21 PONTOS	15 PONTOS	8 PONTOS
NF	0,663	1,266	1,602	1,414	2,012
kNF	1,033	1,359	1,561	1,445	1,908
ANN	1,661	1,515	2,098	2,094	2,395
AWC	0,932	1,098	1,662	1,294	1,910

[37] apresenta método de localização via smartphone baseado em *fingerprinting* que utiliza coeficiente de correlação de Kendall Tau (KTCC). Comenta que sistemas baseados em RSSI de WiFi apresentam erro de posicionamento que variam entre 2 e 3 metros, enquanto o sistema proposto, baseado em BLE, apresenta erro médio de 0,87m. Também discute interferência do corpo humano, por absorção no espectro de BLE, e impacto de orientação do smartphone na geração da base de dados.

[38] apresenta novo método de localização baseado em *fingerprinting* que utiliza técnica de *deep learning* em conjunto com informações de estado de canal, sistema nomeado de DeepFi. Utiliza sinais de WiFi com parâmetro de entrada para sistema de decisão o CSI (*Channel State Information*). Na fase offline, o algoritmo de *deep learning* é utilizado para encontrar os pesos para os dados de *fingerprinting*. Na fase online, é utilizado um modelo probabilístico baseado em RBF para obter a localização.

[39] propõe sistema de localização via Bluetooth que utiliza múltiplas redes neurais (MNN – *multiple neural network*), orientada pela experiência do usuário. São usados *beacons* para medição a cada 20 metros e usada janela para cálculo de média de RSSI contendo 8 amostras. Esse sistema usa um modelo de árvore de navegação, no qual ele armazena dados de posição possíveis a partir da anterior. Se essa nova posição não for possível a partir do antigo valor, a captura é realimentada no sistema de redes neurais. Ainda, a nova posição só é aceita pelo sistema após a saída da árvore de navegação for n vezes repetida. Para testes, ele define um conjunto de posições fixas no espaço de testes e calcula o erro de predição dessas posições, comparando com sistemas com 1 camada de rede neural e com múltiplas e com e sem árvore de navegação. Para múltiplas redes a precisão foi em média de meio metro e identificou as posições com acurácia de 89%, em contrapartida com rede neural simples precisão foi de 1,4m e acurácia de 55%.

[40] apresenta sistema de localização via BLE. Devido aos efeitos não lineares sofridos pelo sinal BLE, tais como múltiplos caminhos, desvanecimento, ruído, reflexão, foi adicionado um módulo de pós-processamento do sinal RSSI, realizado em tempo real. Também utilizado um módulo de calibração dos dados RSSI antes do processamento. Essa calibração é baseada no modelo de propagação do sinal que leva em consideração um fator de perda de sinal na

propagação. São usados pontos a 1 metro de distância da estação de captura. No sistema proposto, o fluxo do algoritmo tem etapas de inicialização ou calibração, captura dos sinais RSSI, cálculo das médias, suavização, uso das distâncias físicas do *grid* e finalmente predição da localização usando a técnica de trilateração.

[22] investiga impacto e funcionamento de dispositivos BLE para localização indoor baseadas em *fingerprinting*. Experimentos demonstraram que a largura de banda dos canais BLE são a causa de erros de medição maiores em comparação com os observados em WiFi, considerando ainda o uso de três canais para *advertising*. Proposto esquema para mitigar influência de propagação em múltiplos caminhos. Foi encontrado que uma frequência de *advertising* de 10Hz e janela de processamento de 1s proporcionam performance ótima. Foi também encontrado que incremento na performance de sistema de localização chega ao máximo com uso de entre 6 e 8 *beacons*. Artigo apresenta estudo sobre impacto de uso concorrente de transceiver WiFi no smartphone. O sistema de localização foi baseado em um estimador bayesiano e os valores de RSSI filtrados usando um kernel gaussiano com pesos atribuídos aos valores, obtendo um mapa de probabilidade dos pontos. O experimento foi realizado em um espaço de 45x12m contendo 19 *beacons*. Erro médio ficou em torno de 2,6m em 95% do período de testes.

Em [41] é apresentado um sistema de localização baseado em BLE e *fingerprinting*. É proposta a separação dos canais de *advertising* e sua diferenciação no sistema de predição da posição. O experimento ocorreu em sala de cerca de 15x14m com 4 *beacons* BLE. Segundo os autores, melhora na precisão obtida no experimento teve um incremento de 12% em relação ao sistema que não diferencia os canais de *advertising*, aumentando a precisão de 49% para 55%.

[2] apresenta sistema de localização que também utiliza de diversidade de canais de *advertising*. É baseado no método de trilateração utilizando pesos e faz uso de filtro de Kalman para pré-processamento do sinal de RSSI. Como *hardware* para transmissão dos sinais BLE utiliza o SensorTag CC2650 da Texas Instruments. Para recepção utiliza um conjunto de três Raspberry Pi, cada uma com pelos menos três *sniffers* BLE. Os pesos do método de trilateração são usados para corrigir casos nos quais os raios calculados para posição do transmissor aos receptores apresentam disparidades, como não intersecção, por exemplo. O sistema é testado em três cenários distintos. Primeiramente é apresentado levantamento de qual melhor sinal a ser utilizado para trilateração, sendo eles: uso unicamente de RSSI de

cada canal de *advertising* separadamente, uso do maior valor entre os canais, uso da média ou do MRC (*maximal-ratio combining*). Foi concluído que o uso do maior valor apresentou melhor performance. Também são comparados os erros dos algoritmos de localização, sendo eles o utilizando dados crus, utilizando diversidade do canal, diversidade do canal e filtro de Kalman e, finalmente, com diversidade, filtro e pesos na trilateração. No último caso, o de melhor performance, a depender do cenário houve ganho de 43% em relação ao primeiro método. Autor aponta que precisão pode ser melhorada aumentando número de *beacons* utilizados, mas, por outro lado, aumentando custo do sistema.

[42] é apresentado um sistema que usa BLE para acesso veicular, tendo como chave um *smartphone*, usando como veículo de testes um modelo da fabricante Volvo. O trabalho é desenvolvido para a empresa de conectividade veicular ACTIA Nordic AB. O sistema é desenvolvido usando redes neurais com conjunto de 6 a 8 antenas nas topologias testadas. Usa como parâmetro de entrada para o sistema de localização o RSSI, pondera sobre o comportamento desse parâmetro, concluindo que para mesmos dispositivos, na mesma posição, ocorre variação acentuada em tempos de captura diferentes, e sua implicação na qualidade do sistema de localização. Conclui que pontos de captura, ou seja, granularidade do *grid* de aquisição de pontos são mais importantes que número de antenas.

A seguir apresenta-se uma tabela comparativa com levantamento de algumas características de sistemas de localização aqui explanados e ainda outros encontrados na literatura.

Tabela 4 – Tabela comparativa de sistemas de localização *indoor*.

REFERÊNCIA	CENÁRIO	BEACONS	PRECISÃO	TECNOLOGIA/MÉTODO DE LOCALIZAÇÃO/ ALGORITMO
[36]	6,66x5,36 m	-	0,932m	BLE/ <i>Fingerprinting</i> /Wasserstein
[37]	-	-	0,87m	BLE/ <i>Fingerprinting</i> /kNN-KTCC
[38]	6x9m	1	2,3m	Wifi/ <i>Fingerprinting</i> /CSI- <i>deep learning</i>
[39]	-	5	89%	Wifi/ <i>Fingerprinting</i> /MNN-árvore de navegação
[40]	4,8x4,8m	3	0,15m	BLE/Trilateração/Média de distância
[22]	45x12m	19	2,6m em 95% do tempo	BLE/ <i>Fingerprinting</i> /-
[41]	15x14m	4	0,55m	BLE/ <i>Fingerprinting</i> /Diversificação

[2]	6,18x6,18 m	4	1,82m em 90% do tempo	BLE/ <i>Fingerprinting</i> /Trilateração-diversificação
	17,6x16,5 m	4	4,6m em 90% do tempo	
[42]	Automóvel	6-8	-	BLE/ <i>Fingerprinting</i> /Máquina de aprendizagem
[43]	44x22m	9	1,58m	BLE/ <i>Fingerprintin</i> /-
[44]	17,5x9,6m	10	2m em 90% do tempo	BLE/ <i>Fingerprinting</i> / <i>Deep learning</i>
[45]	32,5x19,2 m	10	3,02m em 80% do tempo	BLE/-/Multi-lateração adaptativa
Ferreira (2018)	Automóvel	3-4	> 75%	BLE/ <i>Fingerprinting</i> /Máquina de aprendizagem

4.2 PATENTES

As patentes descrevendo sistemas de acesso veicular baseados em comunicação BLE são profícuas e algumas delas são analisadas a seguir.

A invenção com número de pedido de patente US 9045102 B2 de 2 de junho de 2015 [46], trata de sistema para abertura remota e acionamento remoto da ignição do veículo. A comunicação é realizada através de protocolo BLE. Usado conjunto de 3 antenas concentradas na parte interna frontal do veículo para minimizar fiação e avalia ser mais simples que demais sistemas conhecidos. No mesmo local, faz uso de antena LF para otimizar determinar de intenção do condutor para abertura do veículo.

A patente US 2016/0347282 A1 de 1 de dezembro de 2016 [47] descreve uma invenção para controle remoto de veículo. Utiliza do sistema de *handoff*, uma caixa preta dentro do veículo, para receber sinais transmitidos de um dispositivo remoto. O sistema de *handoff* pode prevenir o usuário do veículo de violar restrições de direção tais como viajar a uma velocidade acima da configurada ou viajar por uma distância determinada.

A invenção com número de pedido de patente US 2015/0048927 A1 de 19 de fevereiro de 2015 [48] descreve um sistema para uso em *after market*. Este sistema

pode travar ou destravar acesso a veículos, prédios e outros locais de forma passiva por proximidade ou por telecomando em aplicativo no *smartphone*.

A invenção com número de pedido de patente US 2014/0240091 A1 de 28 de agosto de 2014 [49] descreve um sistema onde a localização é determinada somente para dentro e fora do veículo. Usa para realizar a localização a técnica de propagação eletromagnética (PLE – *Effective Path Loss*). Usa uma segunda antena, opcional, como antena de referência de forma a ter um melhor sinal para garantir resolução e determinar localização no interior ou exterior do veículo. A chave pode ser também um *smartphone*.

A invenção com número de pedido de patente US 2015/0362997 A1 de 17 de dezembro de 2015 [50] demonstra um dispositivo *wearable* que reconhece gestos para realizar ações no veículo.

A invenção de número de pedido de patente KR1020160059793 de 27 de maio de 2016 [51] descreve sistema para autenticar usuário e permitir controle do ambiente do veículo (cores, músicas, temperatura, iluminação) a partir da leitura de dados biométricos (reconhecimento de íris, batimentos cardíacos, estado emocional). Tais leituras seriam realizadas através de dispositivos *wearable* (*smart watch*, *smart glass*) e transmitidos via *wireless* para unidade de controle e autenticação do veículo.

A invenção de número de pedido de patente US 2016/0358389 A1 de 08 de dezembro de 2016 [52] foca em sistema de acesso e ignição remota de veículo usando *smartphone*. Utiliza de três códigos distintos: de pareamento, de identificação e de autenticação armazenados no *smartphone*. A inserção dos códigos no *smartphone* pode ser realizada na montadora ou concessionária ou podem ser baixados via rede GSM, outro *smartphone* ou veículo habilitado. O código é armazenado em um chamado “*secure memory element*”, recomendado pela descrição da invenção para que seja usado cartão SIM para tal. O processo de armazenagem dos códigos pode ser simultâneo ou em separado. Não utiliza criptografia adicional à do protocolo BLE. Memória segura é acessada via código de segurança.

A invenção de número de pedido de patente KR101680090000 de 5 de março de 2016 descreve um dispositivo *wearable* do tipo “*wristwatch*” com controle de comandos do veículo por gestos e com propriedades de localização, usando técnica de *fingerprinting*. Para detecção de gestos utiliza de sensores de movimento, seja

magnetômetro, acelerômetro, giroscópio etc. Para localização utiliza conjunto de quatro antenas. Realiza abertura e partida da ignição passivas. Não trata de questões de segurança da comunicação ou armazenamento de chaves.

A invenção de número de pedido de patente WO 2016/180724 A1 de 17 de novembro de 2016 [53] é um sistema para acesso remoto a veículos que utiliza ondas sonoras para propagar mensagens de autenticação. Foca em técnicas para redução de consumo de energia como desligar módulos do sistema quando não utilizados ou logo após a realização de autenticação de usuário. Em relação a segurança comenta sobre “*relay attacks*”, definindo uma região máxima de ativação para suprimir esse tipo de ataque. Não cita criptografia ou métodos de armazenamento de chaves nem como autenticação é realizada. Precisa de elemento para iniciar comunicação e, assim, diminuir consumo. Há necessidade de sincronizar relógios do “*key fob*” e do veículo para melhorar detecção de pisos e vales do sinal sonoro.

A invenção de número de pedido de patente US 2016/0183042 A1 de 23 de junho de 2016 [54] descreve um sistema para localização de usuários em ambiente “*indoor*”, não foca em acesso passivo. Para tal, propõe o uso de dispositivos de comunicação ao longo do ambiente que enviam suas posições absolutas, a partir do uso de coordenadas ou outras variáveis como o piso do andar, para um receptor, o qual pode ser um celular.

A invenção de número de pedido de patente US 2016/0288770 A1 de 06 de outubro de 2016 [55] descreve um sistema para entrada e partida veicular passiva. Foco maior em consumo de energia, principalmente com o automóvel desligado, em especial no *tradeoff* segurança versus “*key-off current load*” (KOL). Para tal, propõe inclusão de um novo módulo, denominado “*remote sentinel*” (RS), que monitora os movimentos próximos ao veículo e aciona os dispositivos, câmeras e radares, usados pelo sistema de prevenção a roubos reduzindo assim o tempo de funcionamento dos mesmos. Ou seja, ativa-os somente em caso de movimentos suspeitos próximos ao veículo. Cita mecanismos de entrada e partida passiva (PEPS) porque os mesmos compõem o mesmo subsistema do “*theft-deterrent system*” (TDS).

A invenção de número de pedido de patente WO 2016/184723 A1 de 24 de novembro de 2016 [56] descreve sistema que tem como foco barrar “*relay station attacks*” (RSA) no acesso de emergência (chave dotada de transponder RFID para

ignição). Para barrar tal ataque, usa elemento de autenticação e um sistema de autorização simples de ser integrado a sistemas existentes.

A invenção de número de pedido de patente US 2016/0292941 A1 de 6 de outubro de 2016 [57] descreve sistema de entrada e partida veicular passiva com foco em protocolo de comunicação entre veículo e *key fob* através de chaveamento de formas de modulação, tempos de cada modulação e sequência durante troca de informações de autorização. Utiliza inclusive chaveamento do sinal de requisição com pelo menos uma transição de modulação.

5 SISTEMA PROPOSTO

Nesta seção será detalhado o sistema de localização de dispositivo de acesso a veículo automotor. Serão descritos a forma como o sistema foi testado, ou seja, a configuração experimental como também a arquitetura do sistema. Dentro da arquitetura do sistema serão indicados e discutidos cada um dos blocos que compõem o sistema (módulo controlador, sensor de porta, módulo de localização etc). Será descrito o fluxo do sistema de localização, apresentando cada um dos módulos que compõem o sistema decisório.

5.1 INTRODUÇÃO

A escolha do sistema de acesso veicular aqui proposto foi realizada conforme uma avaliação prévia das tecnologias disponíveis, mostradas nos capítulos anteriores. A tecnologia de comunicação escolhida foi a *Bluetooth Low Energy* (BLE), por apresentar-se como a tecnologia com maior presença nos dispositivos de uso pessoal atualmente, como *smartphones*, *smartwatches*, *smartbands* etc. Desse modo, para realizar acesso veicular de modo passivo e simultaneamente provendo conectividade com dispositivos em largo uso no mercado, a tecnologia BLE foi utilizada como meio físico para troca de mensagens do sistema de acesso veicular passivo proposto. Além das características citadas, outros fatores relevantes foram considerados, tais como, experiência do usuário, segurança e robustez do sistema, custo agregado para desenvolvimento, implantação e manutenção do sistema.

No tocante à experiência do usuário, as condições a serem atendidas são usabilidade do sistema, nível de passividade para acesso ao veículo, conveniência de uso do dispositivo, tempo de resposta para acesso, dentre outros.

Em relação à segurança e robustez do sistema, deve-se garantir a confiabilidade do sistema como também garantir que o mesmo não se torne uma porta de acesso à rede intraveicular. Além disso, garantir que seja seguro a ataques externos de agentes maliciosos e que mantenha funcionamento adequado naqueles estabelecidos pelas especificações do veículo. Ou seja, o sistema deve ser o quanto

possível imune a ataques externos para acesso à comunicação intra veicular e, portanto, aos dados sensíveis de funcionamento do veículo (relacionado ao conceito de *security*).

O custo do sistema deve ser fator essencial para permitir a viabilidade do sistema para as montadoras, seja mantendo custo semelhante aos sistemas de acesso passivo anteriores ou agregando valor ao veículo que justifique um aumento no custo de implantação.

Dentre essas condições, a tecnologia que melhor atendeu às demandas e apresentou maior característica de inovação foi a *Bluetooth Low Energy* (BLE), como já citado acima.

Nas seções a seguir, serão apresentados a configuração do experimento utilizado para desenvolvimento e testes do sistema, as arquiteturas tanto do sistema, de hardware como do sistema de localização e a descrição de cada um dos módulos componentes de cada arquitetura.

5.2 CONFIGURAÇÃO DO EXPERIMENTO

A configuração experimental construída para o desenvolvimento do sistema teve como objetivo medir os valores de RSSI (*Received Signal Strength Indicator*) dos sinais recebidos do dispositivo utilizado como chave veicular.

Os módulos de hardware utilizados para implementar o sistema de abertura veicular passiva foram da empresa norueguesa Nordic, fabricante de SoC (*System on Chip*), contendo central de processamento ARM e módulo de comunicação na tecnologia BLE, entre outros módulos contidos no kit de desenvolvimento utilizado. O kit de desenvolvimento utilizado foi o nRF52 DK com o SoC nRF52832. Os desenhos da placa são mostrados na Figura 22 e na Figura 23.

Além dos módulos da Nordic, para o experimento foram utilizados módulos Arduino, um dispositivo de interface de hardware da empresa de dispositivos eletrônicos para o mercado automotivo Vector, o VN8900, e um computador para controlar os softwares dos dispositivos citados e receber e processar as mensagens trocadas em cada uma das interfaces. O diagrama de blocos da arquitetura de hardware do experimento é mostrado na Figura 24.

Figura 22 – Vista superior da placa nRF52.

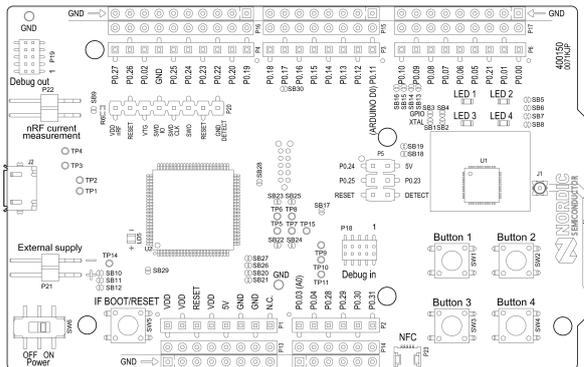
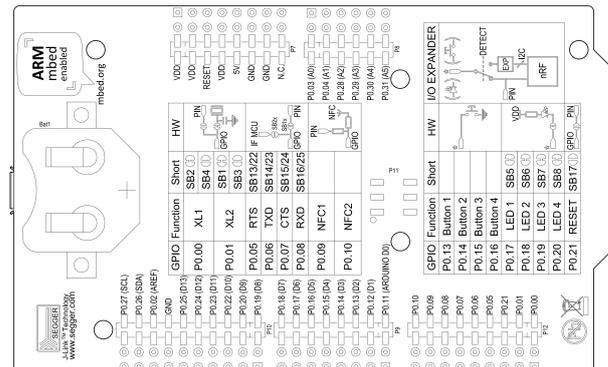


Figura 23 – Vista inferior da placa nRF52.



Para o sistema proposto e realização dos experimentos e provas de conceito, foi escolhido o Arduino Mega. Arduino é uma plataforma de prototipação com largo uso no mercado e com uma comunidade de desenvolvedores e de suporte ativa. O Arduino Mega, em particular, oferece três portas seriais adicionais (além daquela utilizada para comunicação com um computador), portanto, com uma conectividade com dispositivos externos considerável. Para o sistema proposto, essa característica é importante para centralizar os dados obtidos dos módulos da Nordic e a partir deles realizar o processamento da localização no próprio Arduino.

O veículo utilizado no experimento foi o Jeep Renegade Trailhawk com sistema Keyless Enter n'go da marca. Esse sistema permite a entrada e partida passiva com a chave, o chamado sistema PASE (*Passive Start and Entry*). Nesse sistema, uma nova ECM é necessária no conjunto eletrônico do carro, a RFHM.

No diagrama são indicadas as formas de comunicação entre cada um dos dispositivos. Os módulos nRF52 utilizam BLE como forma de comunicação. Um destes módulos faz o papel da chave do veículo, sendo carregada pelo usuário ou em algum suporte durante o experimento, e os demais módulos nRF52 estão posicionados como antenas no veículo para troca de mensagens com a chave. Estas mensagens servem tanto para determinar localização da chave em relação ao veículo, através do uso do RSSI de cada mensagem, como autenticar a sua abertura ou ignição. No fluxo de funcionamento do sistema, as próprias antenas são responsáveis pela autenticação da chave, possuindo nos seus firmwares os protocolos de segurança e autenticação. A determinação da localização da chave nas regiões definidas em relação ao veículo é realizada pela transmissão dos sinais de RSSI recebidos nas antenas para o Arduino, onde está implementado o algoritmo de localização. A comunicação entre Arduino e antenas é realizada através de sinais

de GPIO e comunicação serial. Os sinais de GPIO são usados para “acordar” determinada antena, a depender de qual sensor foi ativado. Por exemplo, se ativado sensor da maçaneta da porta do passageiro, a antena daquela região será “acordada” para realizar a autenticação da chave. O sinal do sensor é enviado através do módulo de interface de hardware VN8900. Portanto, o VN8900 envia para o módulo da Nordic um sinal que indica que o módulo deve iniciar o processo de recepção de sinais de RSSI e de envio desses sinais para o módulo Arduino. Além disso, o VN8900 é responsável pela troca de mensagens com as ECM (*Electronic Control Module*) do veículo, no caso BCM (*Body Control Module*) e RFHM (*Radio Frequency Hub Module*), pois é necessário que haja comunicação via CAN usando tabela de mensagens proprietária da montadora para ativar destravamento das portas ou ignição do motor.

No sistema proposto, o algoritmo de localização usa exclusivamente o sinal de RSSI como dado de entrada e precisa de uma base de dados de sinal gerada off-line, i.e., antes do funcionamento em tempo real do sistema, que corresponde a um mapeamento de sinal da área em torno do veículo como também no seu interior. Esta técnica, discutida na Seção 3.2, denominada *fingerprinting*, precisa de uma geração de base de dados anteriormente ao funcionamento do sistema em tempo real. Para tanto, nos experimentos, foi utilizado veículo do modelo Jeep Renegade onde algumas composições de antenas foram feitas de forma a buscar a melhor configuração para recepção dos sinais de RSSI e, portanto, otimizar o resultado do algoritmo de localização da chave. Um exemplo de pontos de captura e possível posicionamento de antena no veículo é mostrada na Figura 25. Esta imagem pode ser comparada com a Figura 26 que apresenta as divisões de regiões para localização da chave, discutida em mais detalhes no Seção 5.4.

A etapa de geração da base de dados pode ser representada como mostra a Figura 27. Os dados são adquiridos numa fase anterior de captura de dados, sendo realizada através dos pontos mapeados, por exemplo, como mostrado na Figura 25. Após a captura dos dados, os quais são valores de RSSI gerados pela chave e recebidos por cada um dos *dongles*, eles são usados na máquina de aprendizagem para treinamento e definição das regiões de localização através desse mapeamento prévio. Essa técnica é denominada *fingerprinting*, como já discutido na Seção 3.2.

Figura 24 – Diagrama de blocos dos módulos utilizados no experimento.

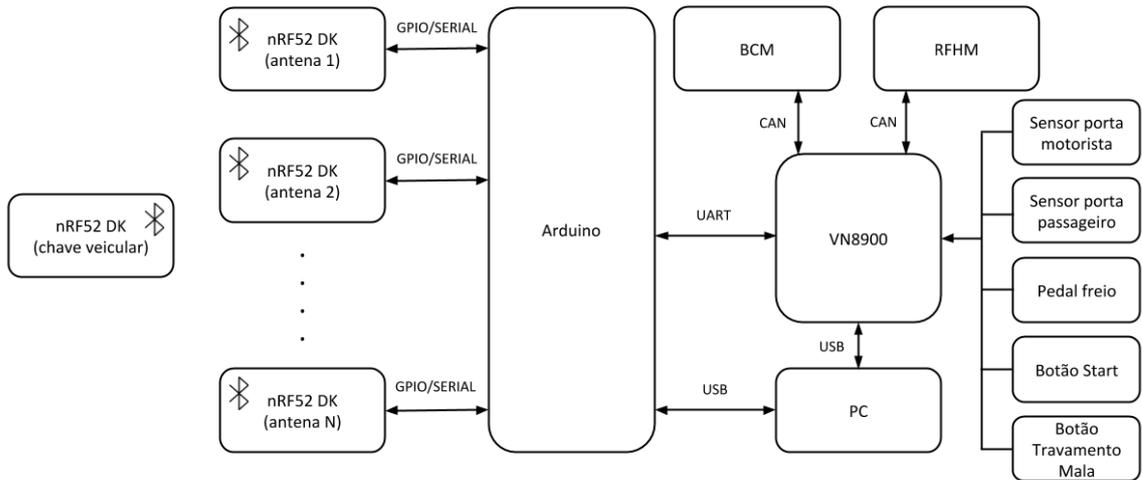


Figura 25 – Exemplo de configuração de antenas (losangos) e posição de captura de sinais de RSSI (círculos).

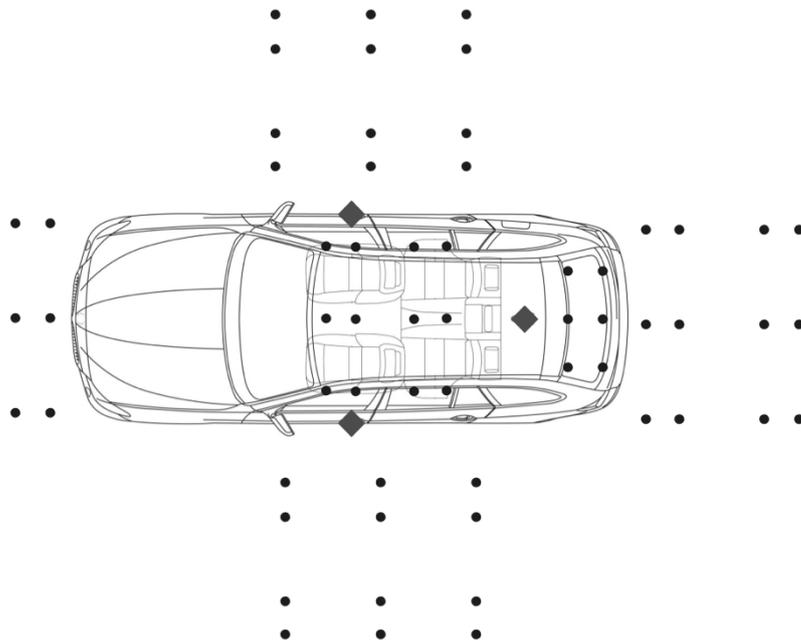


Figura 26 – Regiões de localização da chave.

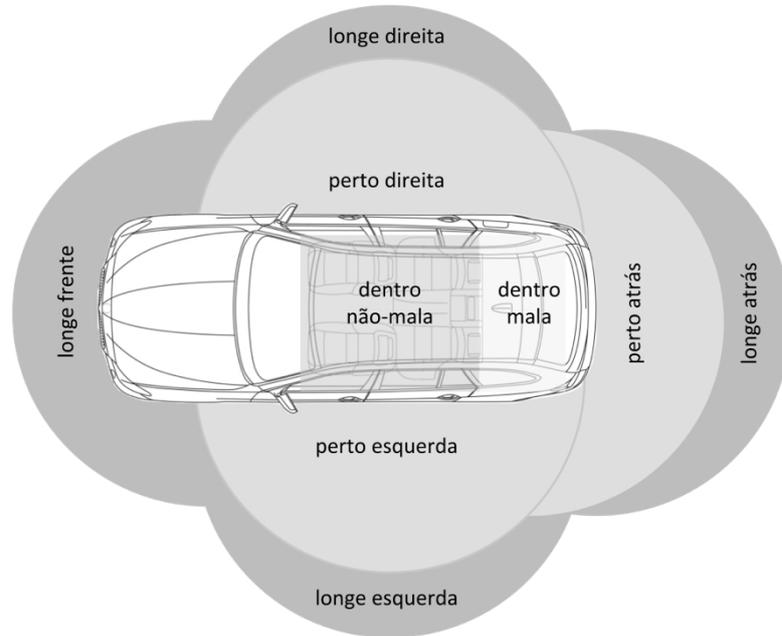
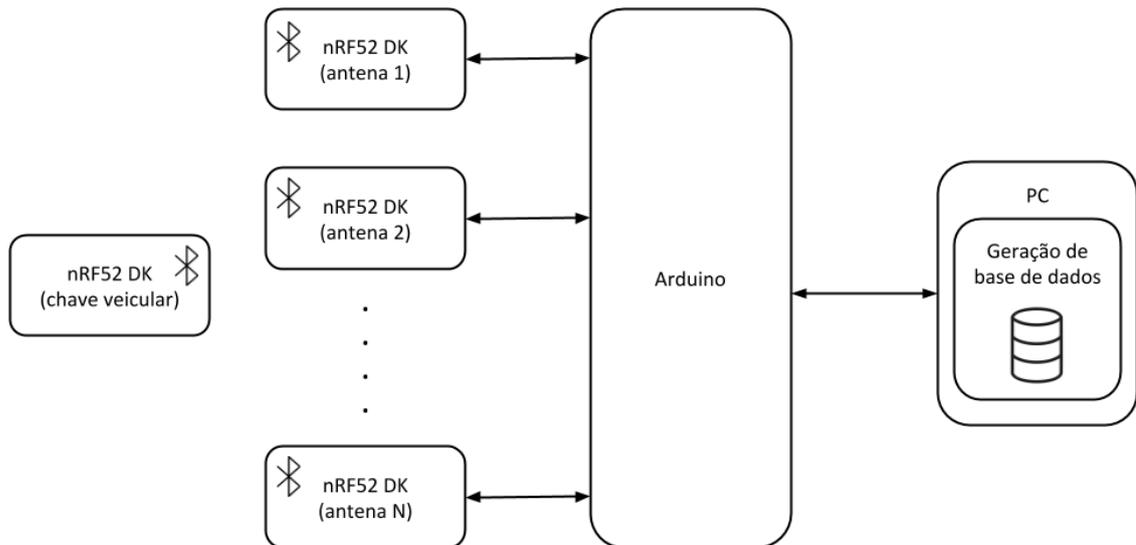


Figura 27 – Geração da base de dados.



5.3 ARQUITETURA DO SISTEMA DE ACESSO VEICULAR

Nesta seção será apresentada de forma detalhada o modo de funcionamento geral do sistema de acesso veicular e descrição de cada um dos módulos da arquitetura. Os diagramas presentes na Figura 28 e na Figura 29.

5.3.1 Funcionamento geral

No diagrama de blocos generalizado da Figura 28, a chave de acesso do veículo não está representada, mas o procedimento de acesso se inicia através da geração de mensagens de *advertising* pelo módulo BLE presente no dispositivo. Como discutido na Seção 2, o módulo BLE na função de *advertiser*, tem como função gerar pacotes de comunicação periodicamente. Os módulos *dongle* BLE, i.e., as antenas posicionadas no veículo, tem como função capturar esses pacotes de *advertising* e enviar para o módulo controlador de acesso o valor do RSSI que está contido como informação nesses pacotes.

O módulo de localização que faz parte do módulo controlador de acesso da Figura 28, como mostrado na Figura 29 onde a linha tracejada delimita os seus componentes, utiliza os dados de RSSI vindo de cada um dos Dongles BLE para executar o algoritmo de localização para determinar a posição do módulo BLE na função de *advertiser*, no final das contas, a chave de acesso do veículo. Após a execução do algoritmo de localização, o módulo de localização envia para o módulo controlador a localização encontrada. A função do módulo controlador foi realizada no experimento pelo módulo Vector VN8900. Este módulo recebe requisições dos sensores de porta e botão de partida e envia comando de abertura ou partida para as ECMs veiculares. O módulo de controlador executado pelo VN8900 opera como um *gateway* entre a BCM e a RFHM do veículo, interceptando mensagens CAN e alterando de acordo com o funcionamento do sistema proposto.

Figura 28 – Diagrama generalizado do sistema de acesso veicular.

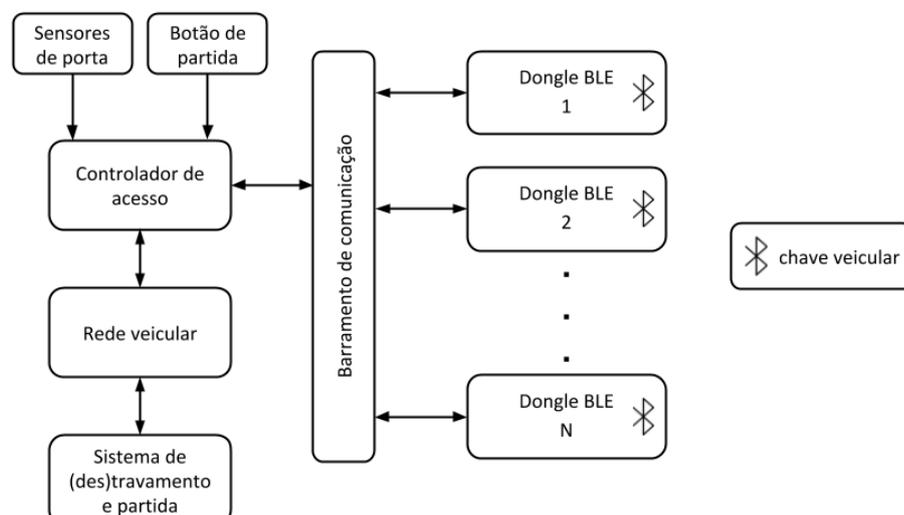
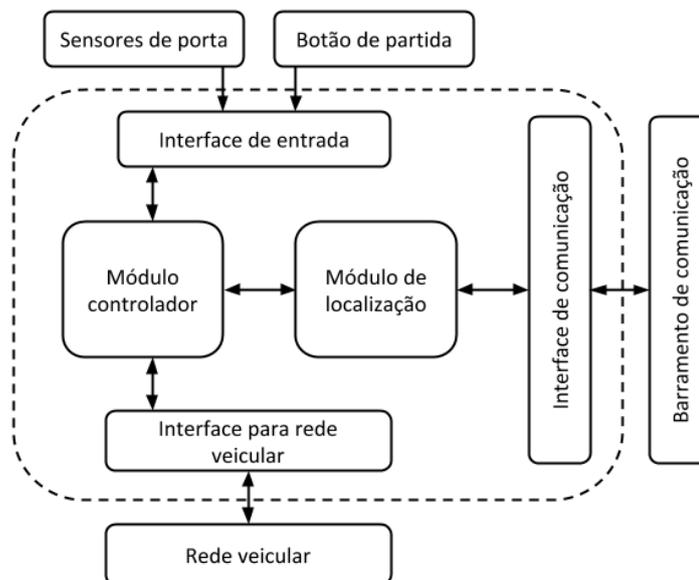


Figura 29 – Diagrama de blocos expandido do controlador de acesso.

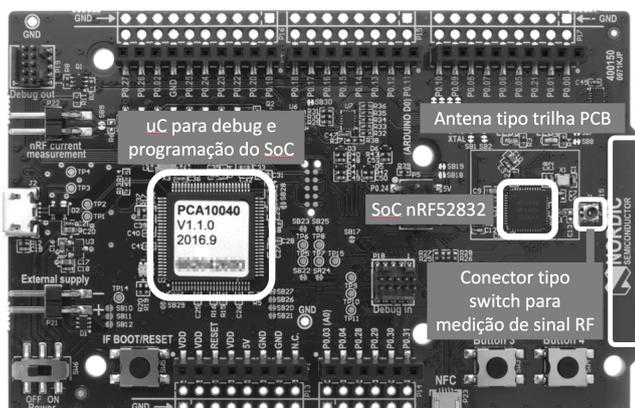


Nas seções a seguir serão detalhados os módulos do sistema proposto.

5.3.2 Chave veicular – módulo BLE na função advertiser

Como chave de acesso do veículo, foi utilizado para o sistema o módulo nRF52 DK da Nordic Semiconductor, como citado anteriormente. O módulo é mostrado na Figura 30.

Figura 30 – Módulo de desenvolvimento nRF52 DK.



O módulo da Nordic foi escolhido por apresentar documentação vasta e comunidade de desenvolvedores ativa. Algumas soluções de hardware BLE foram

testadas previamente e elas possuíam como módulo de comunicação BLE o SoC nRF52832 da Nordic. Algumas características do módulo são elencadas a seguir:

- SoC multiprotocolo: BLE, ANT e uso do rádio de 2,4GHz para aplicações sem fio de baixo consumo de potência usando protocolo proprietário;
- Processador de 32 bits ARM Cortex-M4;
- Memórias: 512kB de flash e 64kB de RAM;
- NFC;
- Níveis de potência de transmissão entre -40dBm e 4dBm;
- Recepção de sinal até -96dBm;
- Periféricos: SPI, I²C, UART, PWM, módulo de encriptação AES em hardware, ADC de 12 bits, contador de tempo real (RTC).

A placa de desenvolvimento foi utilizada somente para teste do sistema proposto. Em versões finais de um possível produto, o único módulo necessário da placa utilizada será o SoC nRF52832 ou uma versão dele, que contém todos os circuitos necessários para funcionamento de um protótipo, como microcontrolador e módulo de rádio, além de uma antena externa ao Soc.

O módulo em funcionamento no modo *advertiser* tem como função gerar pacotes ACL (Asynchronous Connectionless). O SDK (Software Development Kit) disponibilizado pela fabricante Nordic para desenvolvimento de firmwares para a plataforma permite variar alguns parâmetros de transmissão das mensagens, como potência do sinal transmitido e latência de transmissão dos pacotes. A transmissão de mensagens de *advertising* no protocolo BLE são realizadas pelos canais 37, 38 e 39, os quais não sofrem interferência de sinais Wifi. A transmissão desses pacotes é realizada de forma consecutiva pulando de um canal a outro, em períodos de tempo fixados no firmware, sendo o mínimo intervalo de tempo 20ms. A escolha do intervalo tem impacto no consumo de potência do módulo e, portanto, na duração da bateria da chave do veículo, como também no tempo de resposta do sistema de localização para obter uma decisão quanto a abertura ou partida. Este *tradeoff* é importante parâmetro para funcionamento do sistema de acordo com os requisitos de mercado para um sistema de entrada e partida passivas.

Como descrito na Seção 2, o módulo BLE operando como *advertiser* recebe a nomenclatura de periférico no protocolo BLE e a mensagem de *advertising* é

formada por 31 bytes de dados, reenviadas a cada intervalo de *advertising*. Para a descrição do sistema de localização proposto, o conteúdo da mensagem não interfere no seu funcionamento, já que o dado importante é o valor do RSSI. No entanto, para o sistema de entrada e partida passiva como um todo, o conteúdo da mensagem é utilizado para garantir segurança e autenticação na troca de mensagens entre o veículo e a chave.

O pseudocódigo do firmware executado no módulo em função *advertiser* é mostrado a seguir:

Quadro 1 – Pseudocódigo da transmissão de mensagens pelo módulo BLE na função *advertiser*.

- 1: Inicialize pilha do protocolo BLE
- 2: Defina número de aplicações de central como 0 e aplicações de periférico como 1
- 3: Defina parâmetros de transmissão do protocolo BLE, como intervalo de transmissão de *advertising*, 20ms, e potência de sinal, 4dBm
- 4: Monte dados do pacote de *advertising*
- 5: Inicie transmissão das mensagens de *advertising*

5.3.3 Dongle BLE – módulo BLE na função scan

O módulo *dongle* BLE, a antena para recepção do sinal de RSSI no veículo, opera como um módulo BLE na função de *scan*, no protocolo BLE um dispositivo nessa função também é denominado de central. Tem por objetivo capturar os pacotes de *advertising* enviados pela chave e obter os valores de RSSI. O hardware é o mesmo utilizado como chave. No entanto, para esse módulo foram usadas antenas omnidirecionais, conectadas usando um *pigtail* através do conector tipo *switch* da placa nRF52 DK, indicado na Figura 30. A antena externa foi usada como forma de melhorar a recepção do sinal e posicionar o *dongle* com maior facilidade nos locais escolhidos no veículo.

Através do software de desenvolvimento da Nordic alguns parâmetros de funcionamento da pilha BLE foram configurados para os *dongles*. O intervalo de *scan*, tempo no qual os dispositivos podem realizar a recepção de mensagem de *advertising* em um dos três canais, com valores de 2,5ms a 10s. Tamanho da janela de *scan* que é o tempo no qual o dispositivo realiza de fato a recepção da mensagem de *advertising*, com valores de 2,5ms a 10s. Configurando o intervalo e a janela de *scan* com os mesmos valores, o *dongle* permanece em modo de recepção todo o tempo, aumentando o consumo de potência, mas permitindo que ele não

perca nenhum pacote. Também é configurado o *timeout* que é o tempo no qual o *dongle* fará a recepção. Após passar o período configurado de *timeout*, o dispositivo entra em modo de economia de energia e passa a não realizar recepção de pacotes. Para fins da pesquisa, os valores foram escolhidos foram 2,5ms tanto para janela como para intervalo, de forma que não houvesse perdas de pacote de RSSI como entrada do algoritmo de localização. Dessa forma, a cada 20ms são recebidos pacotes de *advertising*, que é o menor valor para configuração do intervalo de *advertising* permitido pela pilha BLE da Nordic.

Quadro 2 – Pseudocódigo da transmissão de mensagens pelo módulo BLE na função *scan*.

```

1: Configure a taxa de transmissão (baud rate) para comunicação com o controle de
localização via UART
2: Inicie as configurações da pilha BLE
3: Configure o dispositivo como central (modo de operação scan)
4: Configure os parâmetros de escaneamento, como janela e intervalo de scan,
timeout
5: Configura lista branca de dispositivos (whitelist)
6: Inicie o escaneamento
    6.1: Verifique se pacote foi enviado por dispositivo cadastrado na whitelist
        6.1.1: Se sim, envia valor de RSSI recebido via UART para módulo de
localização
        6.1.2: Se não, descarta RSSI
        6.1.3: Retorna a passo 6.1

```

A biblioteca da Nordic permite configurar uma lista de dispositivos que terão pacotes recebidos pelo *dongle*. Essa lista é denominada *whitelist*, como citada no pseudocódigo. O cadastramento dos dispositivos é feito usando o endereço MAC.

5.3.4 Consumo de potência

O consumo de potência para o modo *scanning* depende, como mencionando, do tempo $T_{scanWindow}$. A Figura 31 mostra, como exemplo de um evento de scanning, uma medição realizada na placa de desenvolvimento nRF52 DK com SoC nRF52838 da fabricante de dispositivos semicondutores Nordic Semi. Nesse caso, a relação entre $T_{scanWindow}$ e $T_{scanInterval}$ foi configurada em 50% com valores de 50ms e 100ms, respectivamente. Como mostra o gráfico, a tensão de pico chegou a 126mV e como a resistência do circuito de medição é de 10 Ohms, então a corrente de pico

foi de 12,6mA. Realizando a média para o período em que o rádio está ativo encontramos uma corrente de 11,3mA e para o período em que o rádio está em *sleep* a corrente média é de 5,38uA. Encontra-se a corrente para o período completo de *scan* usando a Equação (17), sendo de 5,65mA nesse caso particular. A Figura 32 mostra a variação da corrente com aumento do intervalo de *scanning* mantendo a janela fixa. A partir desse gráfico, pode-se observar que o consumo de corrente e, portanto, de potência, apresenta uma queda considerável, abaixo de 1mA, em torno da configuração de um intervalo de *scan* maior que cerca de um segundo. Esse intervalo de *scan* que permite menor consumo, no entanto, pode ter impacto no tempo de resposta do sistema, retardando a tomada de decisão. Portanto, para obedecer à restrição de consumo de potência no caso de implementação em veículo, o desenvolvimento do sistema pode fazer uso de dispositivos de interação com o usuário, como maçaneta capacitiva ou botão, de forma que sistema de acesso somente seja acionado no momento necessário. Este aspecto não será tratado neste trabalho.

$$I_{scanning} = \frac{I_{active} \cdot t_{active} + I_{sleep} \cdot t_{sleep}}{t_{active} + t_{sleep}} \quad (17)$$

Figura 31 – Medição de corrente em dispositivo em scanning.

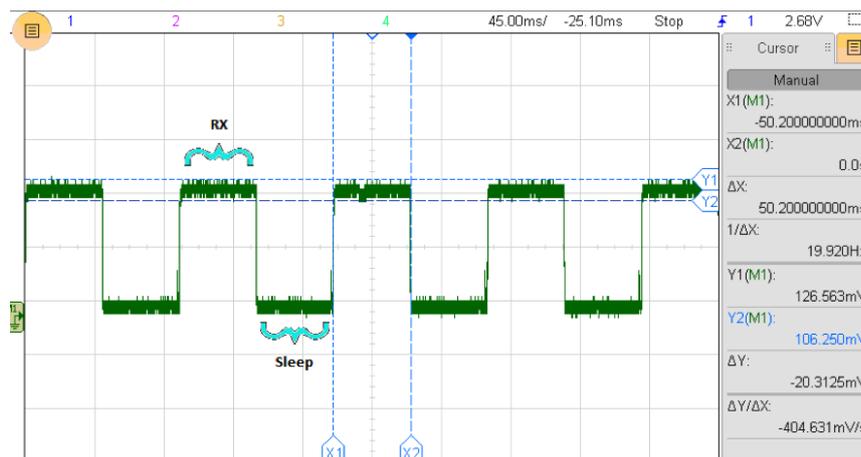
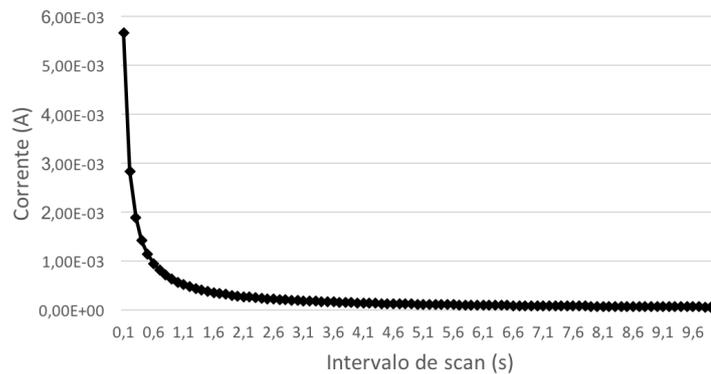


Figura 32 – Corrente para scan com janela fixa em 50ms.



Para o dispositivo em modo de *advertising*, diferentemente do modo de *scanning*, a cada intervalo de *advertising* o dispositivo transmite o pacote nos três canais, 37, 38 e 39, como mostra a Figura 14. O tempo total para a execução do evento de *advertising* é dado pela relação entre o tempo denominado de intervalo de *advertising* mais um tempo variável de atraso, $t_{advDelay}$. Então, o tempo total do evento é dado por $T_{advEvent} = T_{advInterval} + t_{advDelay}$. O protocolo BLE permite a configuração do parâmetro intervalo de *advertising* entre 20ms e 10,24s, com o tempo de atraso podendo variar entre 0 e 10ms. No caso do *advertising*, alguns *softwares* de desenvolvimento de aplicações BLE permitem configurar em quais canais serão realizadas as transmissões dos pacotes, permitindo que o dispositivo, dentro de um evento de *advertising*, só envie em um ou dois canais. A Figura 33 e a Figura 34 mostram, a título de exemplo, medição em laboratório da tensão consumida pelo dispositivo durante a realização dessas transmissões. Em particular na Figura 34 está mostrado, através dos três picos de tensão em destaque, que todos os canais de *advertising* estavam habilitados. Nesse caso, o intervalo de *advertising* foi configurado para 20ms, mas como mostrado na Figura 33, no canto direito para o valor de ΔX , o período total do evento ficou em 21,6ms. Pode-se perceber que esse período varia entre os eventos. Usando Equação (18), com I_{active} calculado em 5,4mA e a corrente I_{sleep} de 1,89uA, chegamos a um consumo de corrente médio do evento de *advertising* em 1,22mA. Da , no canto direito para o valor de ΔX , retiramos que t_{active} é 4,89ms, sendo um parâmetro que pode ser reduzido se menos canais forem utilizados.

$$I_{adv} = \frac{I_{active} \cdot t_{active} + I_{sleep} \cdot t_{sleep}}{t_{active} + t_{sleep}} \quad (18)$$

Figura 33 – Medição de tensão consumida durante advertising.

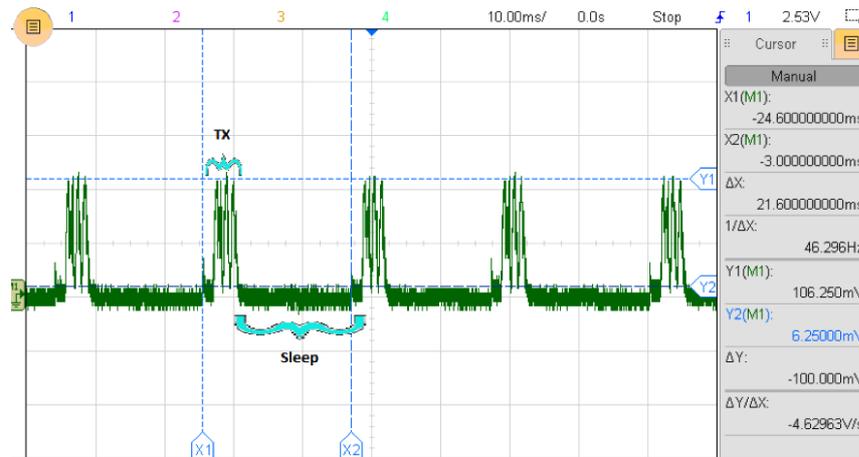


Figura 34 – Medição de tensão em advertising com foco na área de transmissão efetiva.

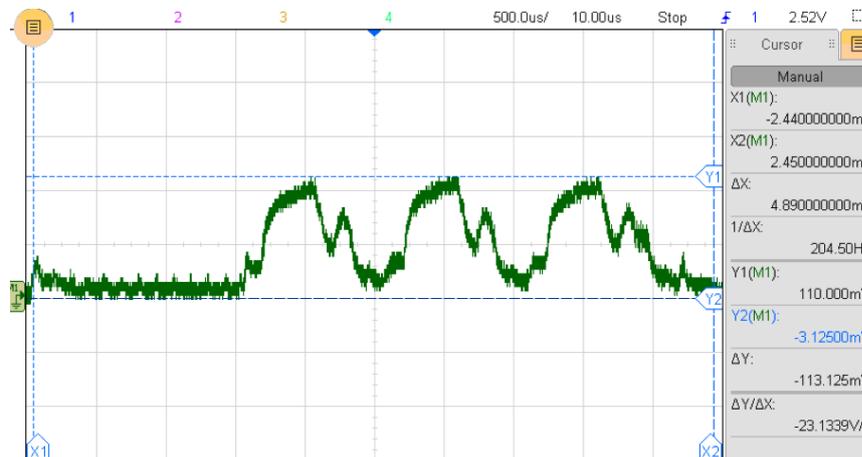
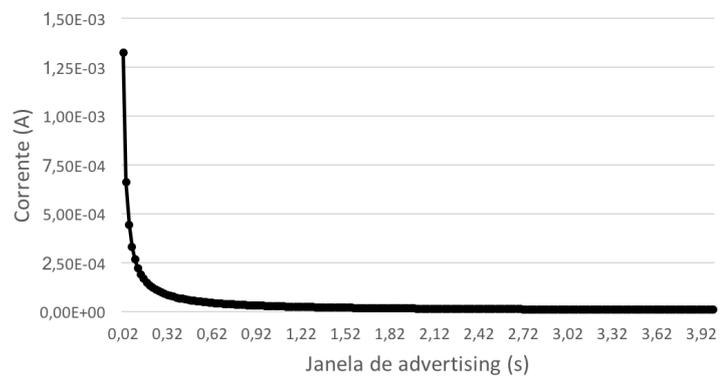


Figura 35 – Corrente consumida em advertising.



5.3.5 Módulo de localização – Arduino

Na configuração do experimento para implementar o módulo de localização foi utilizado o Arduino Mega. As características do hardware são:

- Tensão de alimentação: 5V;
- Tensão de entrada (recomendada): 7 a 12V
- Pinos de I/O digitais: 54, sendo 15 podendo ser usados para saída PWM
- Pinos de entrada analógicos: 16
- Memória: 256kB flash, 8kB SRAM, 4kB EEPROM
- Velocidade de clock: 16MHz

Como mostrado nos diagramas da Figura 28 e da Figura 29, o controlador de acesso tem entre seus componentes um módulo dedicado a implementar a localização da chave veicular. No sistema desenvolvido, o Arduino faz esse papel. Ele recebe os pacotes de dados de *advertising* enviados por cada dongle BLE via UART e extrai os valores de RSSI, identificando qual corresponde à determinada antena. O algoritmo de localização utiliza como entrada esses valores e determina em qual região em relação ao veículo a chave está localizada. As características desse algoritmo serão mostradas na Seção 5.4.

De acordo com a quantidade de *dongles* utilizados para recepção do sinal BLE, foi necessário o uso de mais de um módulo, pela limitação do Arduino em número de portas seriais disponíveis, três adicionais além da porta serial que permite comunicação com computador.

5.3.6 Módulo controlador – Vector VN8900

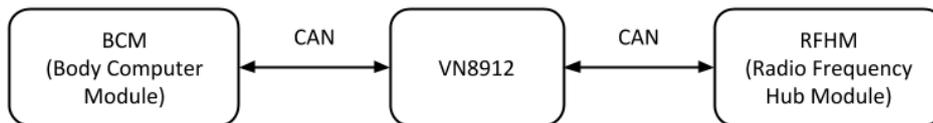
O módulo controlador serve como interface entre veículo e sistema desenvolvido de acesso e partida passivos. Para que o sistema proposto possa alterar comportamento do veículo é necessário que as mensagens CAN que trafegam na rede veicular sejam conhecidas e modificadas.

Como explicado na configuração do experimento, o carro utilizado foi o Jeep Renegade com o sistema PASE, portanto, o carro possui a central RFHM que implementa as funções de acesso e partida passivos. Para nosso sistema funcionar

alterando o funcionamento do veículo, era necessário que o carro utilizado possuísse essa central, permitindo que o sistema próprio da montadora fosse substituído pelo sistema PASE desenvolvido.

A rede interna veicular opera, entre outros tipos de protocolo, através de mensagens do protocolo CAN (*Controller Area Network*). Estas mensagens são montadas com cabeçalhos e comandos que, para algumas funções, exigem que as mensagens das centrais eletrônicas sejam de alguma forma autenticadas com uma identificação única da central (ECM). No sistema desenvolvido, o VN8912 foi utilizado como *bypass* entre as centrais BCM e RFHM do Renegade, como mostrado na Figura 36. Não sendo assim, o módulo controlador do sistema proposto deveria ser desenvolvido para substituir integralmente a RFHM, todas as funcionalidades implementadas, inclusive aquelas que não seriam modificadas, como também o protocolo de segurança utilizado na comunicação entre as centrais.

Figura 36 – VN8912 como gateway entre BCM e RFHM do Renegade.



Uma representação do módulo VN8912 utilizado no desenvolvimento é mostrado na Figura 37. O módulo VN8912 é um módulo base com processador Intel Core-i7 integrado que aceita placas de conexão como a mostrada na imagem. A placa de conexão utilizada foi a VN8970. Este módulo possui 8 canais que permitem comunicação CAN, LIN ou FlexRay e um nono canal dedicado para entradas ou saídas analógicas ou digitais.

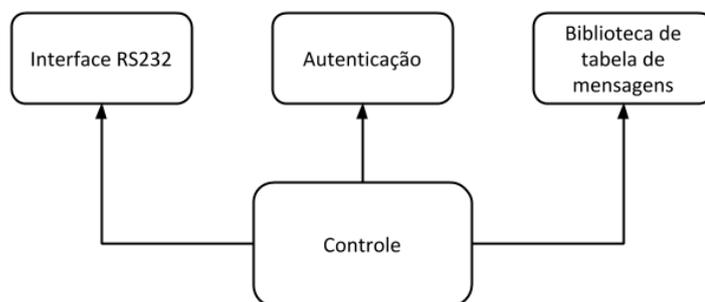
Figura 37 – Módulo VN8912 com placa VN8970 da Vector.



A arquitetura de software da aplicação desenvolvida no VN para controle de comandos no Renegade segue o diagrama mostrado na Figura 38. O módulo controle é responsável pelo gerenciamento dos demais módulos e tomada de decisões do funcionamento do sistema e o bloco da biblioteca de tabela de mensagens é responsável pela tradução de comandos do controle em mensagens CAN que estejam no padrão do mapa ou tabela de mensagens do Renegade, que serão enviadas pela rede CAN para a BCM ou para RFHM.

É através do módulo de controle que todas os tratamentos de eventos são executados, como por exemplo, inicialização de parâmetros quando se inicia a execução do *firmware*, implementação da função de *gateway* quando há recepção de mensagens CAN vindas da BCM ou RFHM e possível modificação dessas mensagens implementando as novas funcionalidades do sistema proposto, controle dos eventos gerados pelos periféricos conectados à RFHM, como sensor da maçaneta, eventos gerados por troca de mensagens pela porta serial, tais como informação de localização da chave pelo Arduino, geração ou não de mensagem de partida do veículo quando há acionamento do botão de partida.

Figura 38 – Arquitetura de software do VN.



O software do VN é implementado em linguagem CAPL (*Communication Application Programming Language*). É uma linguagem orientada a eventos, i.e., a execução de qualquer tarefa pelo *firmware*, seja ela função, atribuição para variável, é realizada se essa tarefa estiver dentro do corpo de um evento acionado por um *trigger* específico, algum sensor ou botão do veículo ativado, inicialização do sistema, recepção de alguma mensagem da rede veicular, alteração de sinal de GPIO ou da interface RS232, estouro de *timer* setado dentro do *firmware*.

O VN8912 pode operar em modo *standalone* ou através do uso do *software* de desenvolvimento da Vector, o CANoe. Através dele pode-se desenvolver o

firmware em linguagem CAPL, como também observar a troca de mensagens CAN entre o VN e o veículo, troca de mensagens pela porta serial etc.

O módulo de interface RS232 permite a comunicação do VN8912 com o Arduino. O VN8912 suporta comunicação com periféricos através um conversor USB/serial ou via Ethernet, como mostra a Figura 39, com as conexões disponíveis com periféricos.

Figura 39 – Parte do painel traseiro do VN8912.



No *firmware* é escolhida a porta que será utilizada na comunicação e a porta serial é configurada para receber o buffer de leitura e assim receber os dados. Quando um evento é detectado na porta serial, o procedimento de recepção é acionado e os dados são tratados. No caso do sistema proposto, a porta recebe continuamente (se chave veicular estiver próximo ao raio de recepção dos *dongles*) dados de localização vindos do Arduino, informando em qual região a chave se encontra, dentro, na mala, próximo do veículo no lado esquerdo ou direito, atrás, longe, por exemplo.

O módulo de autenticação é responsável pela validação do ID recebido via UART, como forma de autenticação da chave veicular pelo veículo através de uma tabela de IDs válidos.

5.3.7 Montagem do sistema e periféricos

Os demais componentes periféricos do sistema mostrado na Figura 28 tem documentação protegida por propriedade intelectual da montadora FCA (Fiat Chrysler Automobiles). Os sinais enviados por eles têm protocolo de comunicação específico e para que fossem incorporados ao funcionamento do sistema proposto eles tiveram que ser decodificados e/ou ter seus sinais interceptados. A Figura 40 e a Figura 41, mostram o sistema PASE da Continental com as antenas que compõem o sistema de comunicação em sinal em HF, como explicado na Seção 2.2, e o botão

KIN de partida conectados à RFHM e à pinagem dos dois conectores da RFHM que ligam fisicamente a ECU às antenas e à rede veicular.

Como mencionado, para que o sistema proposto esteja conectado aos periféricos e mesmo à rede CAN, os sinais correspondentes tiveram que ser interceptados fisicamente. Isto foi feito como mostrado na Figura 42.

Do conector mostrado foram interceptados os sinais da CAN (*high* e *low*), sinal de pedal de freio e posição do *switch* de ignição. Os sinais da CAN foram desconectados da RFHM, ligados em novos conectores para o VN, que reenvia os sinais para a rede CAN após modifica-los, se necessário. Os demais sinais não são retirados, mas usados para monitorar atividades e colher informações para a tomada de decisão do módulo de controle.

Figura 40 – Diagrama de blocos do sistema PASE da Continental retirado de manual do produto submetido à ANATEL.

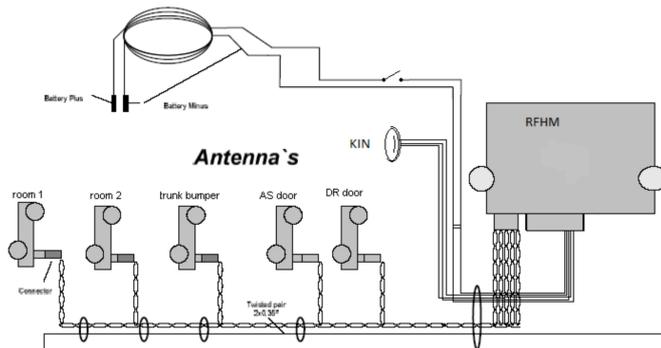
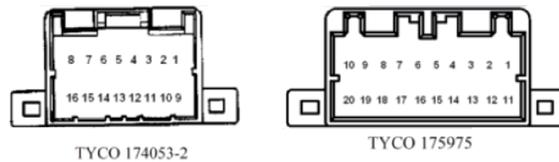
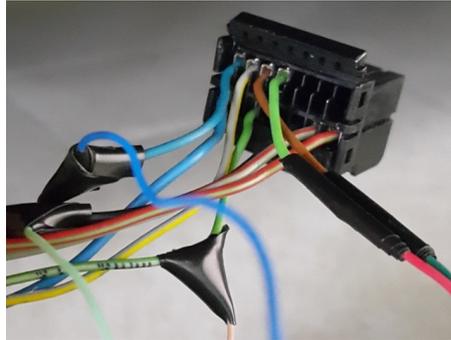


Figura 41 – Pinagem dos conectores da RFHM extraído de manual do produto submetido à ANATEL.



Pin	X1 Signal Name	Pin	X2 Signal Name
X1_1	PWR – IGN_RS	X2_1	PE antenna 5 Signal
X1_2	GATED PARK SW	X2_2	PE antenna 4 Signal
X1_3	BRK – PEDAL FORCE SW	X2_3	PE antenna 3 Signal
		X2_4	PE antenna 2 Signal
		X2_5	PE antenna 1 Signal
X1_6	GND – GENERIC		
X1_7	TRX CLUTCH SW SIG		
X1_8	PWR – GENERIC BATT FEED		
X1_9	CRK – IGNITION RUN/START SW SENSE		
		X2_11	PE antenna 5 Return
		X2_12	PE antenna 4 Return
		X2_13	PE antenna 3 Return
		X2_14	PE antenna 2 Return
		X2_15	PE antenna 1 Return

Figura 42 – Fios do conector da RFHM ligado à rede veicular alterados.



5.4 DESCRIÇÃO DO SISTEMA DE LOCALIZAÇÃO

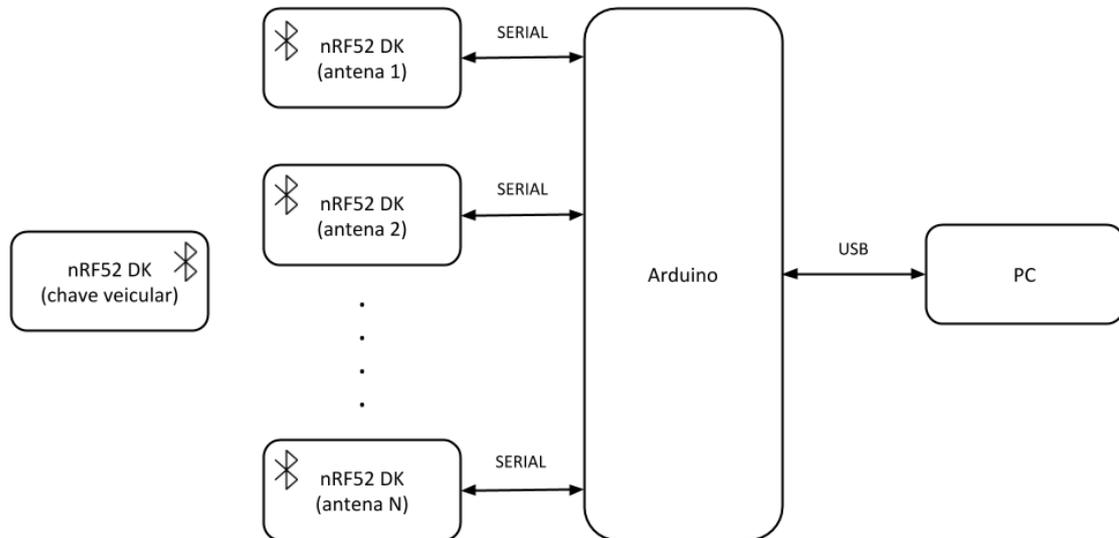
Nesta seção, será apresentado o sistema de localização utilizado no sistema proposto, sistema contido no módulo de localização mostrado na Figura 29, como parte do módulo de controle de acesso da Figura 28.

O desenvolvimento do sistema de localização usou os módulos de *hardware* mostrados na Figura 43. O sistema de localização desenvolvido tem como entrada unicamente os valores de RSSI da chave capturados pelos *dongles*. No entanto, como mostrado na Seção 5.4.1 os valores de RSSI apresentam uma variação que depende de fatores não-lineares não controláveis e facilmente calculáveis através de métodos analíticos, como interferência do corpo humano, multi-caminhos do sinal, *fading*, obstáculos externos ou internos ao automóvel. Dessa forma, considerando que a construção de modelo analítico para o cálculo da posição da chave seria onerosa tanto em tempo de desenvolvimento como para embarcar no módulo Arduino ou, futuramente, em sistema embarcado automotivo, foi definido que o sistema de localização seria baseado em máquina de aprendizagem. Para tanto, foi utilizada linguagem C, para desenvolvimento do *firmware* do Arduino, e linguagem Python para desenvolvimento das ferramentas de captura e visualização de sinais de RSSI dos *dongles* e desenvolvimento dos códigos para treinamento das máquinas de aprendizagem testadas e geração dos dados de saída.

Para melhor entendimento de como a comunicação entre os dispositivos de *hardware* foi implementada, mostra-se na Figura 44 o diagrama simplificado do fluxo de comunicação entre os dispositivos utilizados, juntamente com o passo a passo das configurações necessárias em cada dispositivo para início da comunicação. Vê-se que a comunicação acontece basicamente num único sentido, da chave em direção ao PC, no qual as informações de RSSI são salvas para treinamento

posterior dos classificadores na fase de treinamento e, na fase de localização ou de testes, o RSSI é utilizado pelo programa presente no Arduino para definição da região a que pertence o ponto testado.

Figura 43 – Componentes de hardware para desenvolvimento e teste do sistema de localização.

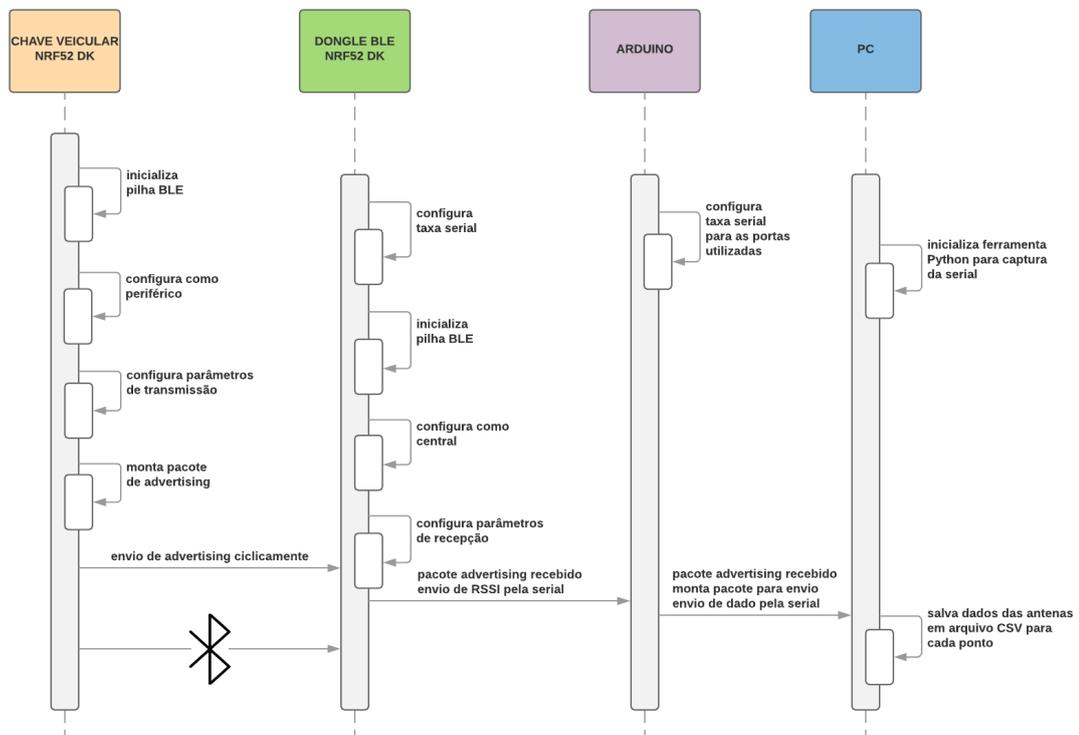


Ao utilizar aprendizagem de máquina como técnica de decisão no sistema, a forma como essa decisão é apresentada foi definida em termos de regiões em relação ao veículo. Por isso, a nomenclatura utilizada para descrição do sistema como regionalizado. Não é importante para a aplicação do sistema de acesso e partida passivos que a definição de posição da chave seja dada de forma precisa, em termos de distância absoluta. A saída do sistema, portanto, foi definida por regiões ao redor e no interior do veículo, diminuindo também a necessidade de captura de sinais de RSSI numa malha densa de pontos, o que afetaria o tempo tomado para geração de base, para treinamento do sistema e para tomada de decisão em tempo real. Dessa forma, foram escolhidas as regiões tomando em comparação as já utilizadas pelo sistema PASE próprio da montadora. As regiões estão mostradas de forma ilustrativa na Figura 26.

Por requisito do sistema PASE original, o limite das regiões marcadas como perto era de uma distância de 1,5m do veículo e foi usado como parâmetro no desenvolvimento. Para o funcionamento geral do veículo, em conjunto com o VN, a área além de 1,5m exige a ação de travamento do veículo suas tarefas auxiliares. A entrada na região de perto tem como ação o destravamento do veículo se acompanhado de toque no sensor da maçaneta. São exemplos da necessidade da

definição entre as regiões escolhidas. Definidas as regiões foram escolhidos os pontos de coleta das bases e com esses pontos, a máquina de aprendizagem foi treinada na fase de treinamento e testada em seguida.

Figura 44 – Diagrama simplificado do fluxo de comunicação entre os dispositivos.



O uso de máquina de aprendizagem implica em duas etapas no fluxo de desenvolvimento. O primeiro é a coleta dos valores de RSSI em pontos fixos previamente definidos, escolha dos grupos de pontos de cada região, tratamento desses pontos (seja por filtro e/ou normalização), escolha de métricas, treinamento da máquina e geração dos pesos, se for o caso. Numa segunda etapa, esses pesos ou mesmo as métricas são utilizados no *firmware* que vai ser embarcado no Arduino para realizar o algoritmo de localização em tempo real, também com as etapas de captura dos pontos, tratamento, entrada na máquina de aprendizagem e, dependendo do fluxo, um pós-processamento para diminuir variações de resposta no limite das regiões. Os fluxos das duas etapas são mostrados nos diagramas de bloco da Figura 45 e Figura 46.

Figura 45 – Fluxo de treinamento da máquina de aprendizagem.

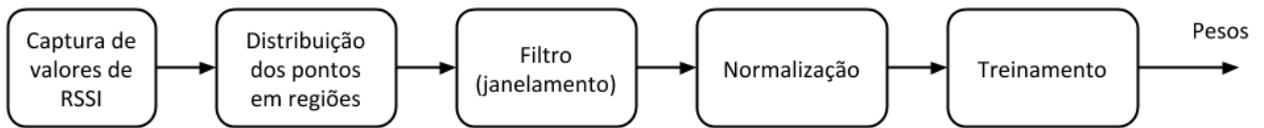
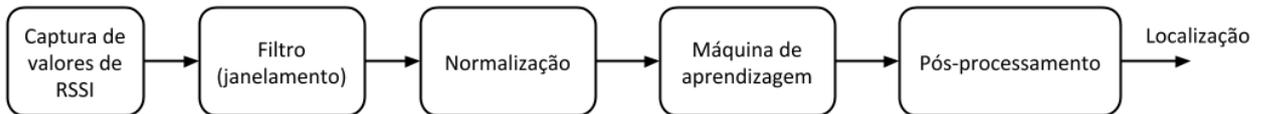


Figura 46 – Fluxo do sistema de localização.



Cada uma das etapas do processo será explicada nas seções seguintes.

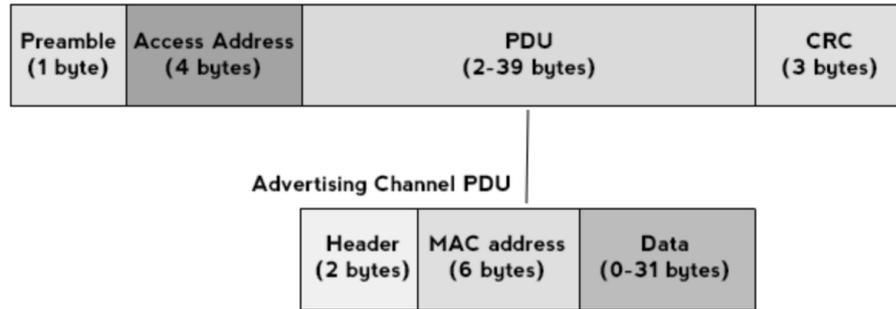
5.4.1 Captura de RSSI

A captura dos valores de RSSI da chave é feita indiretamente durante a recepção do pacote de dados de *advertising* da chave pelo *dongle*. Contido no rádio do SoC nRF52832 há um circuito para medição da potência de sinal com sensibilidade de sinal entre -90 e -20dBm, com uma precisão de 2dBm dentro desta faixa, resolução de 1dBm e tempo de amostragem de 8us.

O *dongle* recebe o pacote de *advertising* no formato mostrado na Figura 47. Pode-se ver que no pacote de *advertising*, o usuário tem 31 bytes de dados para uso (presente no PDU) e o dispositivo é identificado pelo endereço MAC de 6 bytes. O RSSI, como citado, não é um dado que é enviado, mas é gerado na própria unidade de recepção e seu valor lido através de registradores do SoC da Nordic.

Para auxiliar e agilizar o processo de captura dos pacotes e geração de base de dados, um programa foi desenvolvido em linguagem Python, com interface gráfica, para definir número de pontos a capturar e mostrar em tempo real os valores de RSSI de cada *dongle* para aquele *advertising*. Esse programa usa os endereços MAC dos *dongles* para identificar qual deles está fornecendo determinado valor de RSSI, de acordo com a posição na arquitetura do veículo.

Figura 47 – Formato do pacote de advertising BLE.



Um exemplo de configuração de placas segundo endereço MAC é mostrado a seguir.

Quadro 3 – Configuração física das placas informada para execução da captura de RSSI.

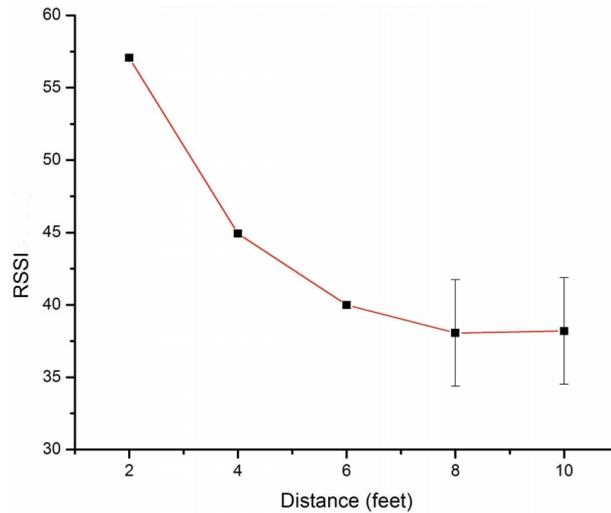
```
{
  "Placas":
  [
    {
      "Nome": "PLACA 1",
      "Posicao": "Lado Esquerdo",
      "Mac": "EF:FA:97:C5:0C:F4"
    },
    {
      "Nome": "PLACA 2",
      "Posicao": "Lado Direito",
      "Mac": "E9:11:B9:1A:8F:A2"
    },
    {
      "Nome": "PLACA 3",
      "Posicao": "Mala",
      "Mac": "D2:CF:AE:BA:C6:79"
    },
    {
      "Nome": "PLACA 4",
      "Posicao": "Centro",
      "Mac": "D2:CF:AE:BA:C6:79"
    }
  ]
}
```

O RSSI nada mais é que uma medida de potência de sinal recebido e pode ser descrito analiticamente como mostrado na Equação (19).

$$RSSI = -10 * n * \log(d) + A [dBm] \quad (19)$$

Onde n é o coeficiente de propagação de sinal, d é a distância entre transmissor e receptor e A é o valor de RSSI de referência medido a um metro. Um exemplo de variação do RSSI com a distância é apresentado na Figura 48, onde vê-se o comportamento não linear.

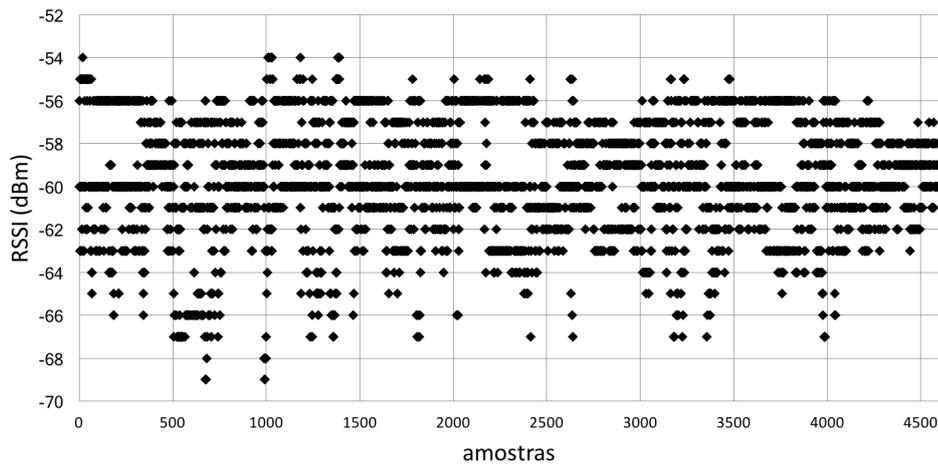
Figura 48 – Comportamento do RSSI com a distância.



O coeficiente de propagação é dependente do meio pelo qual o sinal se propaga, portanto, para o cenário aqui proposto esse coeficiente não tem comportamento linear, pois o meio de propagação não será homogêneo. Isso porque há interferência do chassi do veículo, corpo humano, objetos no interior do veículo a depender da posição do *dongle*, se chave está dentro de algum outro objeto etc.

O comportamento do RSSI pode ser visto na Figura 49. O gráfico mostra o comportamento para um conjunto de amostras de RSSI para um experimento onde receptor e transmissor eram SoCs da Nordic, fixos e sem obstáculos adicionais entre eles, a uma distância de 1 metro.

Figura 49 – Valores de RSSI obtidos com distância de 1m entre dispositivos.



Como forma de minimizar esse ruído apresentado na captura do RSSI, foi observado que a curva de retorno de sinal e $VSWR$ de antenas em geral, a utilizada no experimento foi do fabricante Inventek Systems, modelo W24-ASMA-M, apresentam curvas de S_{11} e $VSWR$ com comportamento parecido com o mostrado nas Figura 50 e Figura 51. De acordo com os gráficos, pode-se perceber que além da não-linearidade própria de sinais de rádio e sua propagação, há um componente adicional gerado pela própria característica da antena. Sabendo que a transmissão dos sinais de *advertising* ocorre através dos canais 37, 38 e 39, distribuídos respectivamente nas frequências, 2402MHz, 2426Mhz e 2480 MHz, uma estratégia foi definir que se usasse um canal fixo para transmissão. Através do *software* de desenvolvimento da Nordic, é possível configurar por quais canais que serão transmitidos os pacotes de *advertising*.

Figura 50 – Curva de return loss de antena utilizada no espectro de Wifi.

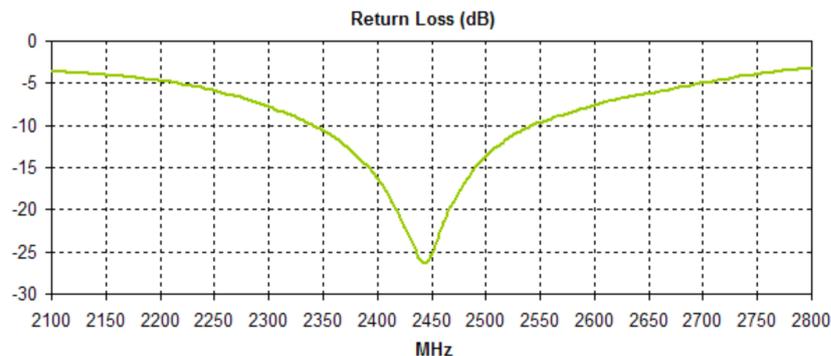
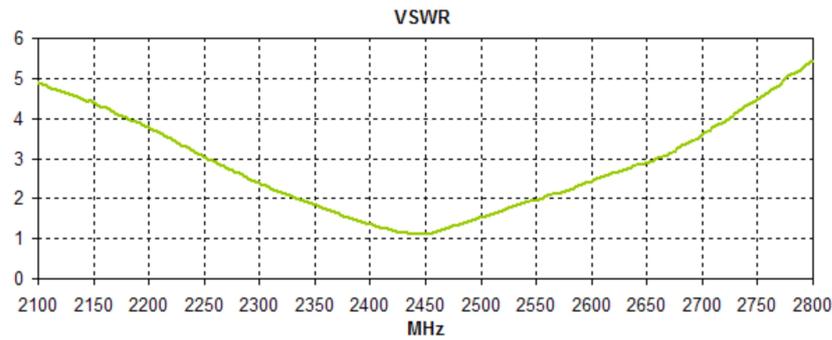


Figura 51 – Curva de VSWR da antena utilizada no espectro de Wifi.



No entanto, mesmo usando canal único para transmissão o comportamento do RSSI continua com bastante ruído. Para minimizar o impacto dessa não-linearidade nos dados de entrada da máquina de aprendizagem, o sistema proposto apresenta um estágio de filtragem.

5.4.2 Filtro – janela passante

A técnica de filtragem do RSSI utilizada usa a média da janela passante por uma quantidade escolhida de dados de RSSI. Essa técnica de filtragem é matematicamente definida pela Equação (20), onde $y[n]$ é o n -ésimo dado de saída, M_1 é o índice da primeira amostra da janela, M_2 é o índice da última amostra da janela e $x[n-k]$ é a amostra.

$$y[n] = \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n-k] \quad (20)$$

O filtro pode ser modificado para atribuir pesos às amostras, dando mais importância a amostras mais recentes. Um exemplo desse tipo de filtro é dado na Equação (21), onde a é um parâmetro de valor menor que 1.

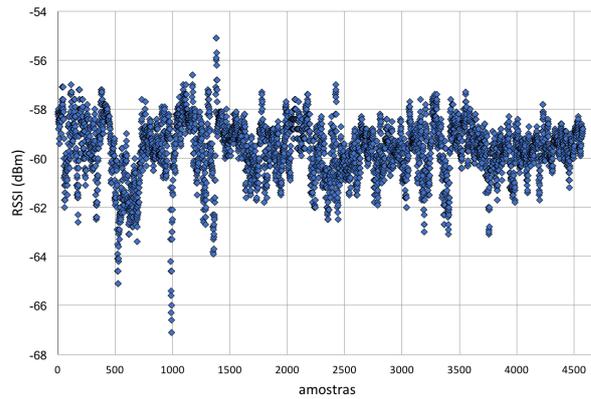
$$y[n] = a^3 x[n-3] + a^2 x[n-2] + a x[n-1] + (1-a) x[n] \quad (21)$$

O filtro de média de janela passante opera de forma a suavizar a saída em relação ao comportamento das amostras de entrada. A depender da variação

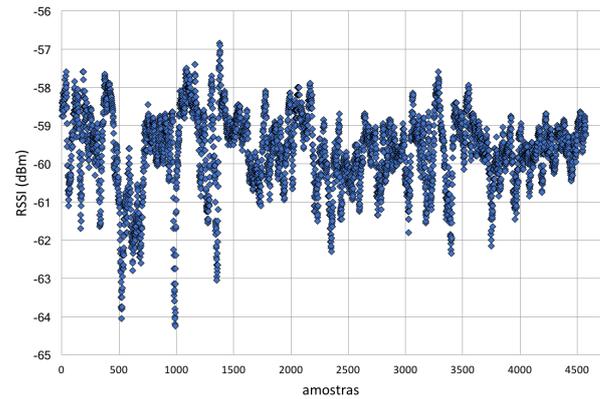
apresentada pelas amostras, pode-se usar uma janela de tamanho específico. Para o caso das amostras apresentadas na Figura 49, são apresentados resultados de aplicação do filtro para diferentes tamanhos de janela nas figuras a seguir. Observa-se que o número de amostras é reduzido em $N - 1$ amostras já que usando janela passante, a primeira amostra da média será calculada na n -ésima amostra de dados originais.

A diminuição do ruído no sinal de entrada para a máquina de aprendizagem deve ser levada em consideração na escolha do tamanho da janela. No entanto, deve-se contrapor que haverá um *delay* para obtenção da primeira amostra da média das amostras de RSSI e também uma demora entre a mudança da saída do algoritmo de localização em relação a mudanças reais na posição da chave. Isso porque a geração de sinais de *advertising* acontece num período mínimo de 20ms, ou seja, para um filtro com janela de 100 amostras, a primeira amostra de média deve estar presente na saída do bloco de janelamento com no mínimo 2s. Também numa possível mudança rápida da chave entre posições relativas do veículo, essa só seria sentida com esse *delay* de 2s.

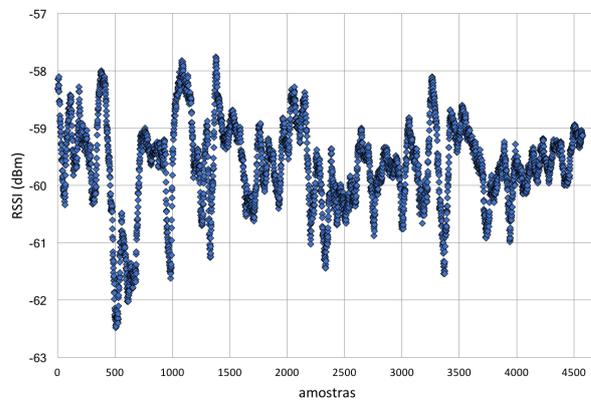
Figura 52 – Comportamento do sinal de RSSI variando tamanho da janela.



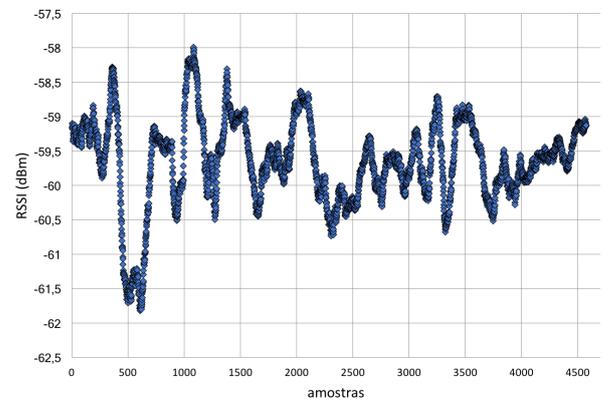
(a) Janela de tamanho 10.



(b) Janela de tamanho 20.



(c) Janela de tamanho 50.

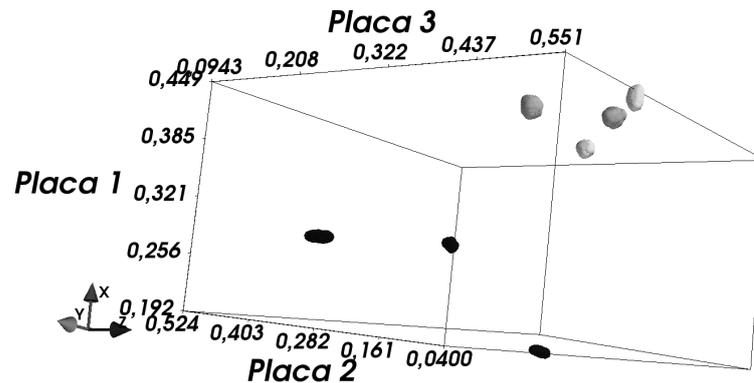


(d) Janela de tamanho 100.

5.4.3 Normalização

A normalização da saída do módulo de janelamento deveria ser realizada de acordo com os limites de RSSI possíveis do SoC nRF52832, segundo especificação de sensibilidade entre -96 e -20dBm. No entanto, ao longo dos experimentos, houve pontos de coleta que apresentaram sinal menor que -96dBm. Assim, para a execução do algoritmo foram escolhidos os limites de normalização entre -120 e -20dBm, com valores na saída no intervalo entre 0 e 1, seguindo a Equação (22).

Figura 53 – Exemplo de dados normalizados.



$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} = \frac{x - (-20)}{(-120) - (-20)} \quad (22)$$

5.4.4 Máquina de aprendizagem

A máquina de aprendizagem utilizada no experimento foi projetada com uma arquitetura de duas camadas. O diagrama de blocos é mostrado na Figura 54. Na entrada do primeiro bloco são inseridas as características escolhidas, no caso do sistema proposto as médias e desvios normalizados a partir das saídas do módulo de janelamento. A saída da primeira máquina divide os dados entre as classes DENTRO e FORA do veículo. Em seguida, os dados são classificados entre as outras duas máquinas, na primeira para definir entre MALA e NÃO MALA (compartimento dos passageiros) e a segunda máquina para definir entre PERTO e LONGE.

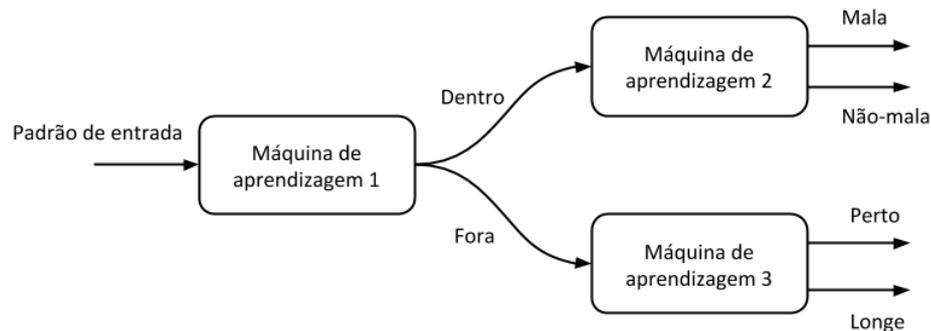
Como possíveis classificadores, foram utilizados o *kNN* (*k-Nearest Neighbour*), MLP (*Multilayer Perceptron*) e árvore de decisão, descritos com mais detalhamento na Seção 3.3. Foi desenvolvido código em Python, utilizando as bibliotecas de aprendizagem de máquina *scikit-learn* e *neurolab*. As máquinas usaram como medida de distância a distância euclidiana. Para o algoritmo de classificação das máquinas de aprendizagem, as máquinas 1 e 2 da Figura 54, foram usadas com MLP e houve variação entre os demais classificadores citados na máquina de aprendizagem 3. Essa escolha entre os algoritmos para cada máquina foi feita por observação do padrão de distribuição dos pontos da base, poder

computacional necessário para execução da máquina, uso de memória e performance na estimativa das regiões.

Para as MLPs, a configuração foi de uma camada escondida, variando a quantidade de neurônios em 5, 10, 15 ou 20 nós, treinamento que utiliza *backpropagation*, que se baseia na heurística de aprendizado por correção de erro propagado da camada de saída para as camadas intermediárias da rede neural. Eventualmente, essas camadas foram testadas com até 100 nós na camada escondida. A função de ativação usada foi a sigmoide logarítmica ou logística e o erro calculado usando erro médio quadrado. Os padrões de entrada são as médias de sinais de RSSI e desvios padrão normalizados obtidos de cada *dongle* BLE utilizado.

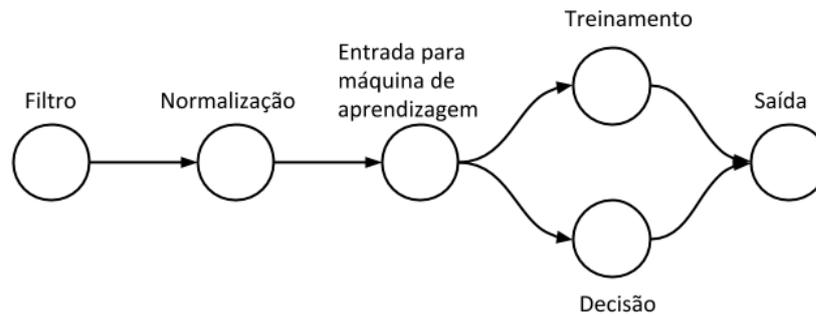
Para o caso da árvore de decisão, não foi definido valor máximo para a profundidade da árvore, método de divisão utilizado foi o Gini, usado por padrão na biblioteca *scikit-learn* de Python. No caso do *kNN* foi utilizado 1-NN, com distância euclidiana e algoritmo de busca de vizinhos variável entre *ball tree*, *KD-tree* e força bruta, a depender de qual o mais adequado segundo os dados de entrada. Todas essas configurações de classificadores encontram-se disponíveis nas bibliotecas Python usadas no desenvolvimento.

Figura 54 – Arquitetura de 2 níveis da máquina de aprendizagem.



Seguindo o fluxo já descrito do sistema proposto e reforçado na Figura 55, a máquina de aprendizagem recebe o sinal filtrado e normalizado. Em um primeiro momento, os dados de entrada são o conjunto dados para treinamento. Após a fase de treinamento, uma segunda base de dados, dita de teste, era utilizada com a máquina de aprendizagem obtida na primeira fase, com fluxo seguindo na parte de decisão.

Figura 55 – Fluxo do sistema proposto.



Para o treinamento, os classificadores foram avaliados através da técnica de *k-fold*, com $k=10$. A base, então foi dividida em 10 partes de tamanho igual de dados, mantendo a proporção entre as classes existente na base total. São realizados 10 testes, sendo que 9 partes são usadas para treino e 1 para teste. Ao final da bateria de testes, cada parte foi utilizada 1 vez como teste. A métrica utilizada para a avaliação dos classificadores foi a acurácia média dos *folds*. O número de épocas para o treinamento foi setada para 250, com meta de erro de 0,01.

O *software* desenvolvido gera automaticamente os códigos em linguagem C que serão exportados para uso no Arduino para testes, com os pesos da MLP, a árvore de decisão ou os pontos de cada classe para uso no kNN.

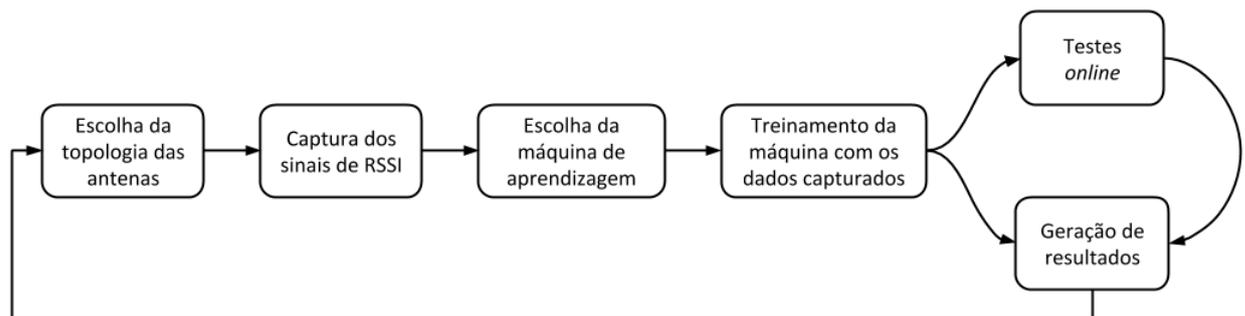
6 RESULTADOS

Esta seção tem como propósito apresentar os resultados obtidos usando o sistema de localização proposto para acesso passivo veicular.

6.1 METODOLOGIA

Os resultados serão apresentados conforme arquitetura apresentadas nas Figura 45 e Figura 46. Para melhor entendimento, a Figura 56 mostra de forma simplificada como os experimentos e resultados foram realizados e obtidos, mostrando um passo a passo metodológico.

Figura 56 – Metodologia dos experimentos.

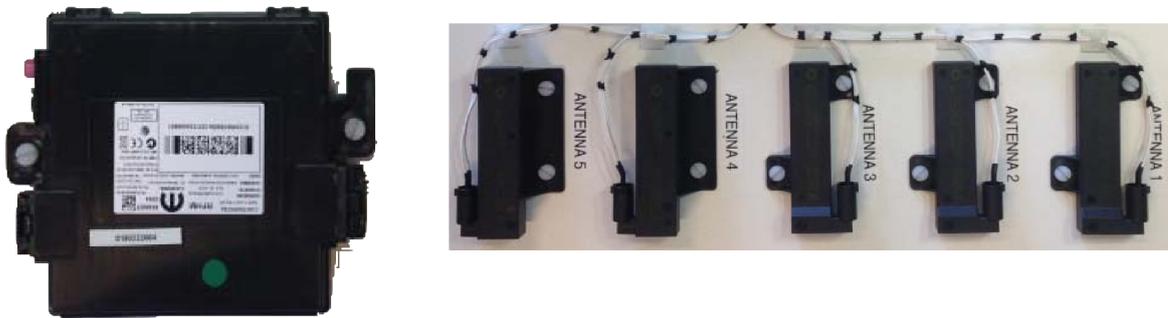


6.1.1 Topologia das antenas

Antes da etapa de escolha da topologia dos *dongles* BLE, ou seja, do posicionamento do conjunto nRF52 e eventual antena externa conectada à placa de desenvolvimento, foi feito levantamento de onde ficam posicionadas as antenas do sistema PASE atual do Jeep Renegade e de estudos que indiquem melhor posicionamento em sistemas que usem BLE nesse tipo de aplicação. No caso das antenas do sistema atual, Figura 57, baseado em sinal na faixa de LF e HF, o posicionamento das seis unidades (cinco independentes espalhadas no veículo e uma embarcada na RFHM) é preferencialmente no forro do veículo, parte superior. Em relação à pesquisa na literatura, em [42] não é apresentado um método ou

justificativa para escolha das posições a não ser que seriam as melhores posições para permitir com maior facilidade a instalação do sistema pela montadora. Outros trabalhos [58][59][60][61] apresentam considerações sobre posicionamento de antenas para outras aplicações, mas na mesma faixa de frequência do BLE, com alguns experimentos apontando melhores posicionamentos. No entanto, vê-se que essa escolha é basicamente empírica, através de tentativa e erro e entendendo o comportamento do sinal de rádio frequência em face da presença de obstáculos, principalmente pela presença humana.

Figura 57 – Antenas LF e módulo RFHM do sistema PASE do Jeep Renegade.



Neste trabalho, foram testadas diferentes topologias, sendo o foco encontrar aquela que apresentasse melhor acurácia com as máquinas de aprendizagem testadas.

6.1.2 Captura de sinais de RSSI

Em relação à escolha do *grid* (tarefa a ser executada dentro da etapa de captura de valores de RSSI), seguindo o exposto na Seção 3.2 e Figura 25 sobre posições de captura e ainda considerando que a região PERTO para o sistema de decisão deve se limitar a uma distância em relação ao veículo de 1,5m, os pontos foram definidos seguindo a lógica apresentada na Figura 58. Principalmente nos pontos limite entre as regiões PERTO e DENTRO do veículo uma distância simétrica foi adotada, ou seja, tomando a porta do veículo como divisor entre as regiões, os pontos foram posicionados a uma distância $x=30\text{ cm}$ da porta. Então, os pontos D1 a D3, E1 a E3 e os próximos às portas na região de DENTRO foram posicionados simetricamente em relação à porta. Dessa forma, empiricamente a porta ficou como limite físico entre essas regiões. Ao longo dos experimentos, o *grid* teve sua granularidade aumentada ou diminuída, ou seja, a densidade das posições de

captura foi variada conforme se sentiu a necessidade de garantir maior representatividade para as regiões ou cobertura maior do espaço de medição ou diminuir tempo de captura de base etc. Outra característica do *grid* em relação à simetria de captura foi manter os pontos mais próximos ao limite de 1,5m do veículo, com mesma distância a essa linha, portanto, se D6 está a 80cm dessa linha, também D7 deve estar.

Na Figura 25, as regiões a princípio foram divididas e o sistema assim identificar as regiões com suas variações da direita, da esquerda, da frente e da traseira do veículo (diferenciando entre PERTO e LONGE). Nos primeiros experimentos, após observar o comportamento do sinal BLE, as regiões foram modificadas e usadas somente PERTO, LONGE, MALA e NÃO MALA. A Figura 58 mostra a matriz de confusão obtida de um experimento com três antenas, com treinamento e teste do experimento usando máquina de aprendizagem baseada em algoritmo *kNN*. A matriz de confusão apresenta o resultado do teste com valores absolutos. A diferenciação entre as regiões de dentro (MALA e NÃO_MALA) em relação às regiões de fora não apresentou tantos erros, a não ser por pontos na traseira próxima ao veículo. No entanto, nas regiões externas a máquina apresentou acurácia muito reduzida, particularmente entre regiões da esquerda e da direita. Uma das razões encontradas foi a similaridade dos valores de RSSI para algumas regiões pela saturação do sinal, ou seja, valor de RSSI capturado pelos *dongles* muito baixo nesses pontos e com pouca variação. Como mostra a Equação (19), o RSSI apresenta comportamento não linear com a distância, então à medida que o dispositivo transmissor se distancia do *dongle*, ocorre uma saturação do valor, deixando de ser possível diferenciar pontos mais ou menos distantes. Então, para pontos a certa distância, estejam à direita ou à esquerda, a saturação começa a ser fator determinante afetando o processo decisório.

Na construção das regiões para entrada na máquina de aprendizagem, o procedimento foi informar o algoritmo de quais pontos formavam cada região. Então, seguindo os pontos mostrados na Figura 58, os pontos por região ficaram da forma mostrada na Tabela 5.

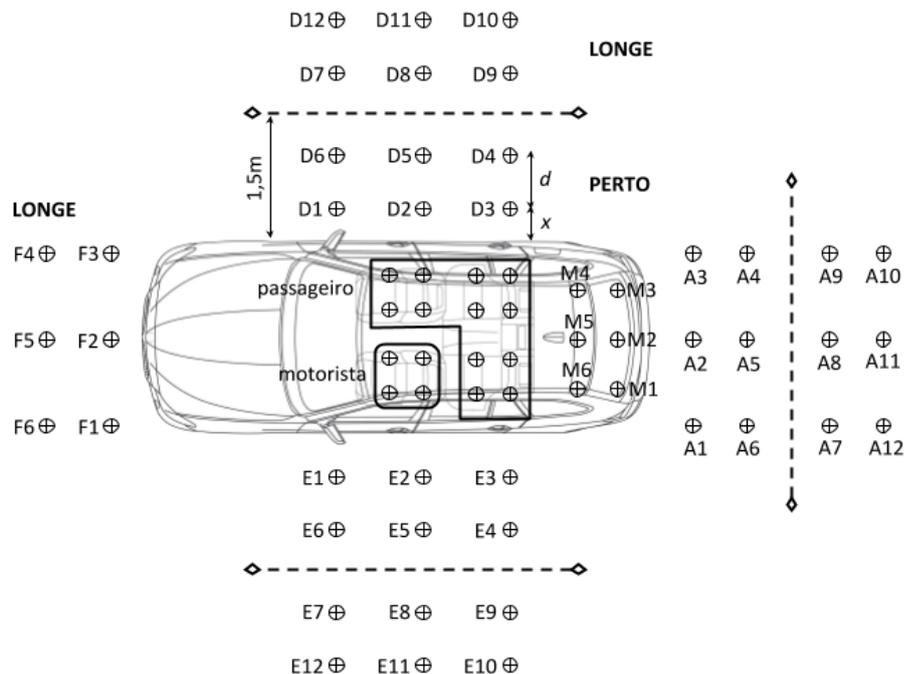
Essa desproporção entre a densidade de pontos para cada região foi corrigida na fase de treinamento dos algoritmos de aprendizagem, para não gerar distorções.

Tabela 5 – Divisão dos pontos do *grid* entre as regiões de decisão.

REGIÃO	PONTOS
MALA	M1, M2, M3, M4, M5, M6
NÃO MALA	Motorista, passageiro
PERTO	D1, D2, D3, D4, D5, D6, E1, E2, E3, E4, E5, E6, A1, A2, A3, A4, A5, A6
LONGE	D7, D8, D9, D10, D11, D12, E7, E8, E9, E10, E11, E12, A7, A8, A9, A10, A11, A12, F1, F2, F3, F4, F5, F6

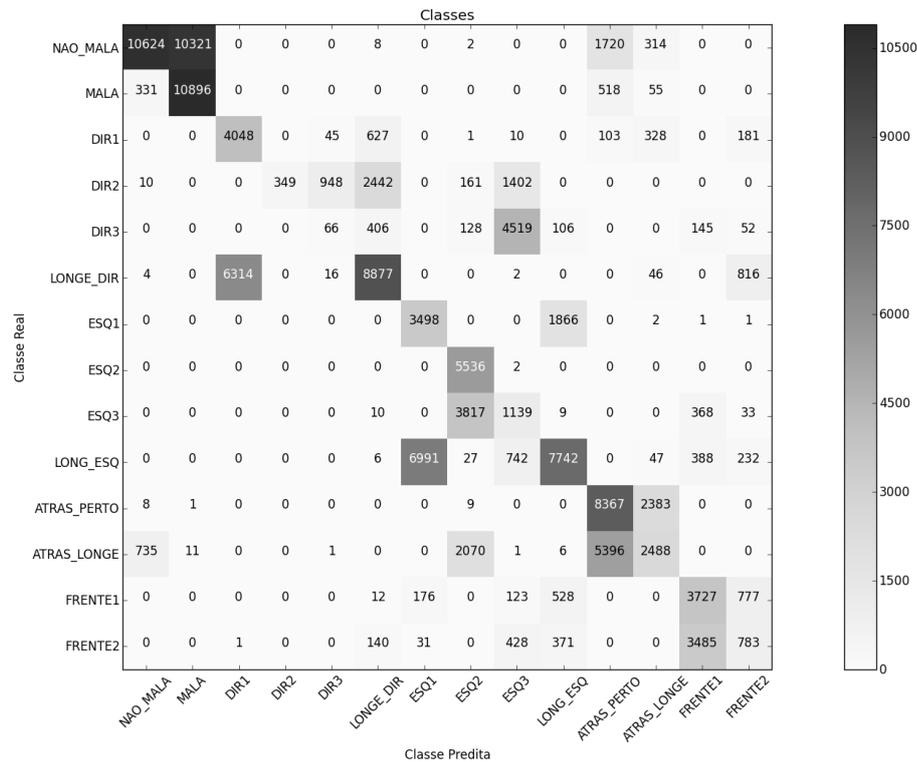
Para cada ponto físico apresentado na Tabela 5, foram capturados 3000 dados de média de RSSI usando a janela passante. Considerando que a janela de *advertising* utilizada foi de 20ms e que não há perda de transmissão, para cada ponto de captura toma-se um tempo de cerca de um segundo para obter os dados necessários.

Figura 58 – Exemplo de grid de pontos de captura para o método de fingerprinting.



Os valores de RSSI foram capturados pelos *dongles* com janela de *scanning* igual ao intervalo e com menor valor (2,5ms). Nos nossos experimentos, restrições de consumo não foram consideradas, já que o foco do trabalho foi especificamente o desenvolvimento de sistema de localização para acesso veicular e não o sistema de acesso completo para implementação no veículo, que, neste caso, deveria levar em consideração, além de consumo de potência, também questões de proteção de dados do veículo contra ataques externos, custo do sistema e integração com a rede intraveicular.

Figura 59 – Matriz de confusão considerando maior diferenciação entre as regiões.

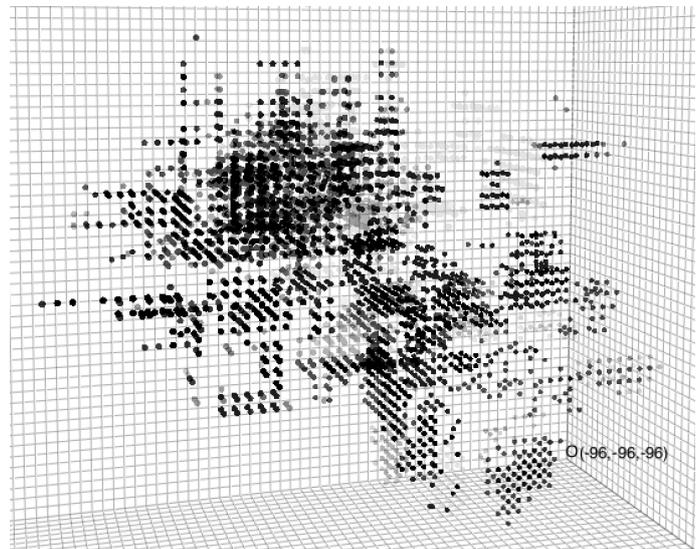


O intervalo de *advertising* para o dispositivo móvel foi configurado para 20ms. Como descrito na Seção 5, o Arduino recebe os valores de RSSI dos *dongles* via serial e um arquivo CSV para cada ponto de captura é criado para em seguida ser usado no treinamento ou teste da máquina de aprendizagem. Alguns valores de captura armazenados em arquivo CSV são mostrados na Tabela 6, para um ponto na posição A2, vide Figura 58. De cada posição do *grid* foram capturados 3000 pontos. Apesar do tempo reduzido para captura de cada ponto (sem considerar atrasos de comunicação ou perdas de pacote, cada ponto levaria 60s na aquisição de RSSI), essa etapa demandou tempo considerável devido à mudança do dispositivo entre as posições, a quantidade de pontos elevada e gerar os arquivos CSV de forma adequada. A Figura 60 mostra todos os pontos de uma base de dados para topologia de 3 antenas. Vê-se que, a partir dos dados crus, sem tratamento, a identificação visual das regiões não é possível, os dados se confundem nas áreas de limite entre regiões.

Tabela 6 – Exemplo de valores de RSSI capturados.

<i>DONGLE 1</i>	<i>DONGLE 2</i>	<i>DONGLE 3</i>
-83	-81	-73
-83	-81	-73
-83	-81	-73
-83	-81	-73
-83	-77	-81
-83	-77	-81
-83	-77	-82
-83	-77	-81
-83	-77	-81
-83	-77	-81
-83	-77	-82
-83	-77	-82
-82	-77	-81

Figura 60 – Representação de base de dados para topologia de 3 antenas.



6.1.3 Máquina de aprendizagem

Em seguida à captura dos sinais de RSSI, o sistema segue o fluxo de tratamento dos sinais como mostrado nas Figura 45 e Figura 46. Nessa etapa, ocorre o janelamento, normalização e geração dos padrões de entrada utilizados como parâmetros na máquina de aprendizagem. No sistema proposto, esses padrões foram as médias de RSSI calculadas na fase de janelamento e os desvios dessas médias. Um exemplo desse padrão de entrada é mostrado na Tabela 7.

Tabela 7 – Exemplo de características de entrada para máquina de aprendizagem.

MÉDIA <i>DONGLE 1</i>	DESVIO <i>DONGLE 1</i>	MÉDIA <i>DONGLE 2</i>	DESVIO <i>DONGLE 2</i>	MÉDIA <i>DONGLE 3</i>	DESVIO <i>DONGLE 3</i>
0.589473684211	0.0171557958032	0.694736842105	0.044034738238	0.52368421052	0.048701988219
0.568421052632	0.0521358230425	0.697368421053	0.075011541124	0.58684210526	0.039032623616
0.542105263158	0.0758150541626	0.678947368421	0.068116732134	0.52894736842	0.090397705459
0.576315789474	0.0468908675821	0.694736842105	0.073496000230	0.55	0.053770570372
0.460526315789	0.0980241080642	0.702631578947	0.047985743496	0.55526315789	0.025309715870
0.552631578947	0.0714656611263	0.689473684211	0.071828126652	0.54736842105	0.053931319820
0.478947368421	0.0969585969839	0.697368421053	0.048345192291	0.57631578947	0.056899337659
0.594736842105	0.0110086845597	0.673684210526	0.075815054162	0.57894736842	0.061010647996
0.555263157895	0.0612937615729	0.694736842105	0.044034738238	0.58157894736	0.061293761572
0.594736842105	0.0399098707582	0.681578947368	0.079166362928	0.56052631578	0.068116732134

Para escolha dos algoritmos de aprendizagem mais adequados para cada bloco, algumas características foram avaliadas. Possibilidade para embarcar no Arduino onde era implementado o sistema de localização após o treinamento da máquina foi um dos aspectos. Nesse caso, usar *kNN* a depender do tamanho da base capturada não era possível pelo limite de memória, no caso do Arduino Mega utilizado de 256KB de memória flash, sendo 8KB para *bootloader*. Outras características eram tempo de execução do treinamento e acurácia obtida. Observando a distribuição dos pontos capturados, exemplo na Figura 61 (pontos da Figura 60 após janelamento e normalização), pode-se entender o comportamento da base. Na figura citada, os pontos mais escuros fazem parte das regiões de PERTO (canto inferior do gráfico), os mais claros da região NÃO MALA e os pontos escuros entrelaçados são da região MALA. Essa separação entre os pontos de dentro do carro e de fora se mantém em todas as topologias de antena (ou *dongles*). O algoritmo para separar os dados na primeira máquina de aprendizagem da arquitetura proposta foi uma MLP. Para a diferenciação entre MALA e NÃO MALA, também foi utilizada MLP. Para o funcionamento do sistema, essa diferenciação é importante pois as montadoras não permitem que o veículo seja ligado (acionamento do botão de partida) com a chave na mala ou que a mala seja trancada quando a chave estiver no seu interior. Para a decisão entre os pontos PERTO e LONGE foram testados os algoritmos *kNN*, MLP e árvore de decisão. Para o funcionamento do sistema PASE proposto, a região LONGE não apresenta nenhuma ação tomada pelo veículo. A região LONGE serve como limitar a região PERTO à distância de 1,5m do veículo, a partir da qual a presença da chave provoca sua abertura. O algoritmo baseado na árvore de decisão alcançou consistentemente as acurácias mais altas nos treinamentos das bases.

Os treinamentos foram realizados usando a técnica de validação cruzada usando folds estratificados, ou seja, com mesma proporção entre as classes, com número de folds igual a 10. Três métodos foram utilizados e testados para escolha da máquina de aprendizagem que define entre as regiões PERTO e LONGE, conforme mostrado na Figura 62. De forma consistente, o algoritmo de árvore de decisão apresentou resultados de acurácia superiores aos de MLP e similares aos do *kNN*, como mostrado na Figura 63. Na média, a acurácia para *kNN* ficou em 98,21%, para a árvore em 97,87% e para MLP em 93,85%. Portanto, nos experimentos foi utilizado o algoritmo de árvore de decisão nas diferentes

topologias, exceto quando se explicita que foram testados outros algoritmos, já que kNN exige maior quantidade de cálculos para determinar a posição e maior uso de memória.

Figura 61 – Exemplo de base capturada em topologia de 3 antenas, dados normalizados e filtrados com janela de tamanho 15.

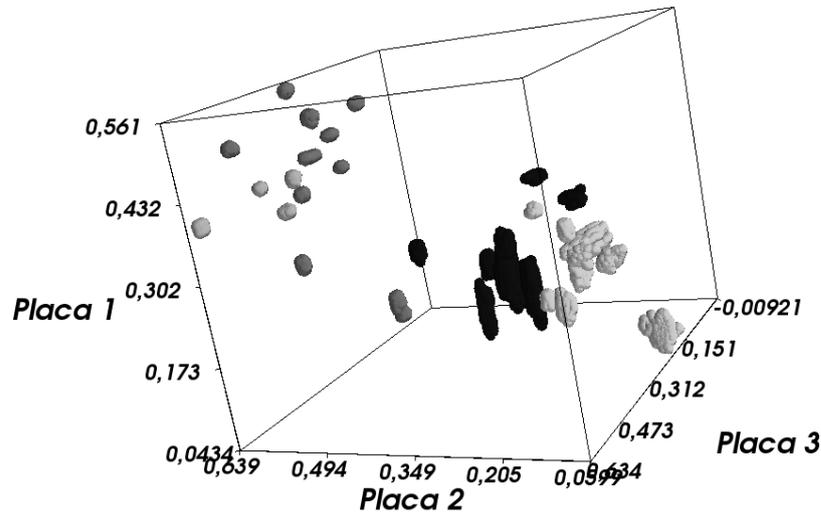
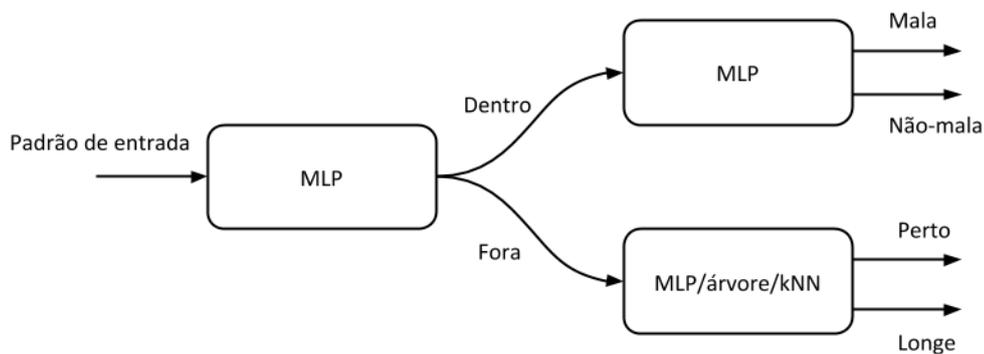


Figura 62 – Máquina de aprendizagem utilizada nos experimentos.

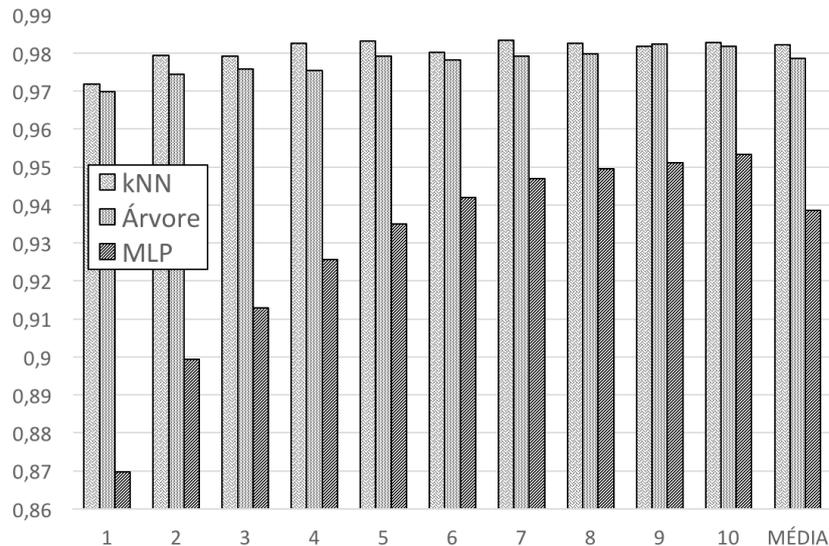


A MLP da primeira máquina de aprendizagem foi configurada com 100 nós na camada interna e a segunda, classificação de MALA e NÃO MALA, com 50 nós. O limite máximo de épocas do treinamento para essas máquinas foi configurado para 250 no caso de o treinamento não atingir o objetivo de erro mínimo configurado para 0,01.

Os resultados mostrados na Seção 6.2 foram obtidos usando MLP para definir DENTRO ou FORA e MALA ou NÃO MALA. Para PERTO e LONGE, foi usada árvore de decisão. A escolha de árvore de decisão no módulo de decisão para PERTO e LONGE, mostrado na Figura 62, como já discutido, foi tomada levando em

consideração que a acurácia obtida foi muito próxima à do classificador kNN, apesar de menor na maioria dos folds. No entanto, a árvore de decisão apresenta menor quantidade de cálculos por tomada de decisão e utiliza menor espaço de memória que o kNN, que precisa manter um conjunto de pontos representativos de cada região para comparação com ponto a ser localizado.

Figura 63 – Acurácia dos algoritmos de aprendizagem.



6.2 RESULTADOS

Os resultados apresentados foram obtidos usando três ou quatro *dongles* BLE, com potência de transmissão do dispositivo periférico atuando como chave configurada para 4dBm. Estão mostrados em conjuntos as topologias dos *dongles* no veículo juntamente com os resultados de treinamento (apresentando o melhor resultado da validação cruzada) e resultados de testes usando a máquina obtida. Esses testes foram feitos com pontos capturados em outro momento, da mesma forma (orientação do dispositivo, altura etc) que realizado na captura da base para treinamento. O dispositivo foi posicionado a uma altura de cerca de 1m do chão, de forma que estivesse à mesma altura quando posicionado no assento do veículo. Importante observar que as bases de treinamento e de testes foram capturadas nos mesmos pontos do *grid* e mantendo a mesma orientação do dispositivo, ou seja, sob as mesmas condições, mas em instantes de tempo diferentes.

Em todos os cenários apresentados, os resultados da validação cruzada apresentaram, como esperado, acurácia de 100% para todas as classes. No

entanto, ao aplicar o algoritmo ao conjunto de dados capturados em outro período, os erros foram significativos em algumas topologias. Como discutido na Seção 5.4.1, o comportamento do RSSI é não-linear e sofre influências do ambiente onde se propaga. No caso da frequência ISM que o BLE utiliza a variação é mais acentuada que no sistema PKE em LF.

As figuras a seguir mostram algumas das topologias testadas nos experimentos e apresentam as matrizes de confusão de treinamento e de testes variando o tamanho da janela do filtro. Nelas, as antenas instaladas no interior do veículo são representadas por círculos quando instaladas no interior do veículo e por losangos quando instaladas no exterior. Através dessas observações empíricas, algumas posições de antena foram descartadas para as demais topologias, como por exemplo, da Topologia 1 foi descartado o uso de duas antenas na mala e da Topologia 2 foi descartado uso da antena no compartimento do estepe, no interior da mala, abaixo do forro do piso. Nessa topologia, não foi possível diferenciar nos testes a posição de pontos na MALA.

Importante notar que todas as máquinas diferenciaram nos testes todos os pontos do interior (MALA e NÃO MALA) do veículo dos pontos do exterior (PERTO e LONGE), os erros ocorreram nas máquinas de aprendizagem 2 e 3.

Observa-se que, apesar da variação no tamanho de janela do filtro variando de 10 a até 25 amostras, os ganhos na acurácia não são consideráveis. Aumenta-se a espera inicial para obtenção do primeiro dado de entrada da máquina de aprendizagem, mas não há ganho significativo na precisão do sistema.

6.2.1 Topologia 1

A topologia 1 foi feita posicionando dois *dongles* no compartimento dos passageiros e outros dois no compartimento da mala, como mostrado na Figura 64. Os dois primeiros foram colocados na parte superior do forro próximo aos para-sois e janelas. E os dois da mala nas extremidades laterais.

Os resultados para essa topologia são mostrados nas Figura 66, Figura 67 e Figura 68. As matrizes de confusão mostram que esse posicionamento não ajuda na diferenciação da região de NÃO MALA, com 75% de acurácia em qualquer tamanho de janela, com o restante das decisões apontando MALA. A região MALA apresentou acurácia acima de 95% no pior caso, com janela de tamanho 15.

Figura 64 – Topologia 1.

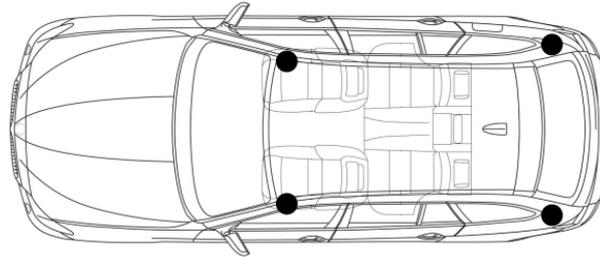


Figura 65 – Matriz de confusão do treinamento da Topologia 1 com janela de tamanho 15.

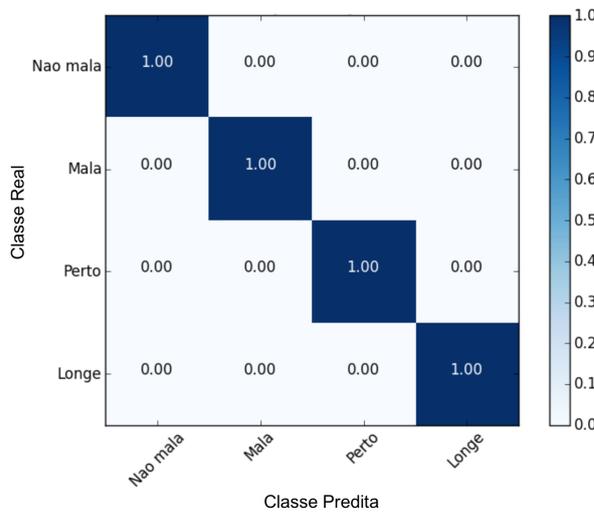


Figura 66 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 15.

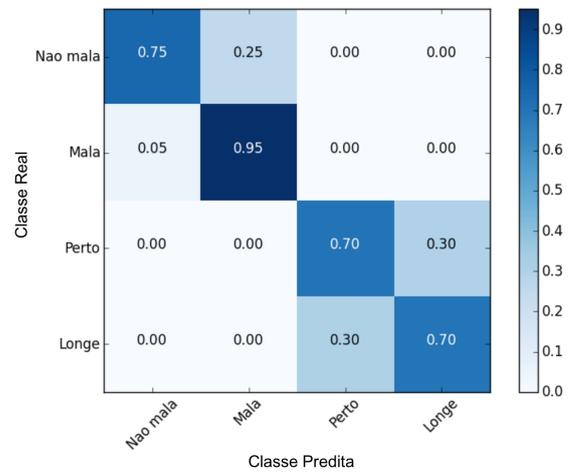


Figura 67 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 20.

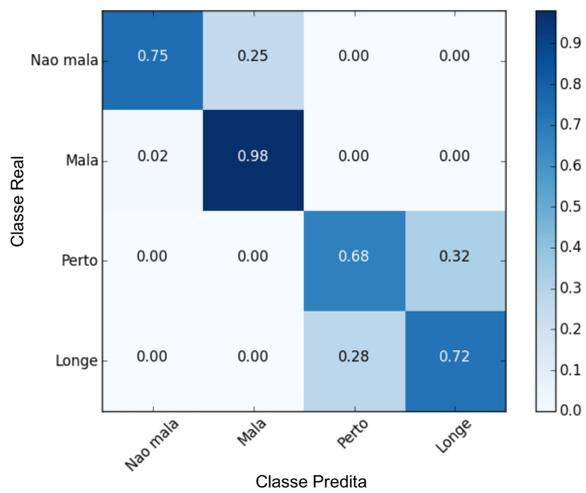
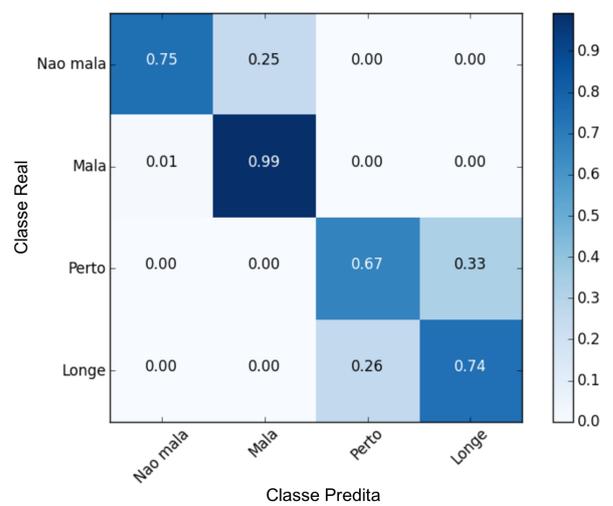


Figura 68 – Matriz de confusão de teste da Topologia 1 com janela de tamanho 25.



Para as regiões de PERTO e LONGE, o erro apresentado ficou em torno de 30% nos dois casos, tanto com máquina definindo a região como PERTO quando o

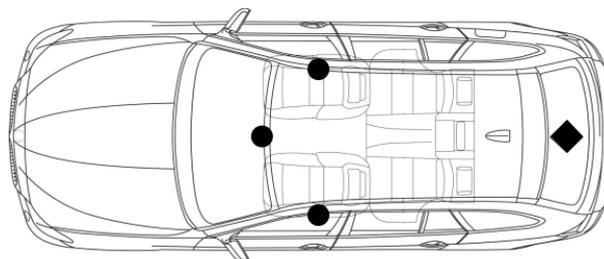
ponto se encontrava LONGE, como o contrário. Essa acurácia reduzida provoca situações que consideramos de *misuse* para aplicação de acesso veicular passivo. Por exemplo, se chave estiver em LONGE e a máquina estimar a região como PERTO, o veículo pode tomar alguma ação de destravamento de portas indesejado. Ou estando fisicamente PERTO e sistema de localização estimar LONGE, o veículo não tomar ação de destravamento esperada.

A partir da observação dos resultados para a topologia 1, a simetria no posicionamento dos *dongles* não foi usada nos experimentos seguintes. Essa simetria no interior do veículo, provoca uma região de indefinição onde os valores de RSSI do par de *dongles* do compartimento de passageiros se confundem com os valores do par da mala.

6.2.2 Topologia 2

A topologia 2 foi montada posicionando três *dongles* no compartimento dos passageiros e um no compartimento da mala, dentro do compartimento do estepe, como mostrado na Figura 69, sendo considerado como posição externa. Dois *dongles* do compartimento de passageiros foram instalados na parte superior do forro a cerca de metade do comprimento das portas. E o terceiro na parte superior central próximo ao retrovisor interno.

Figura 69 – Topologia 2.



Os resultados para essa topologia são mostrados nas Figura 70, Figura 71 e Figura 72. As matrizes de confusão mostram que esse posicionamento não ajuda na diferenciação da região de MALA, com 50% de acurácia em qualquer tamanho de janela, com o restante das decisões apontando NÃO MALA. Como o *dongle* estava dentro do compartimento de estepe, a atenuação do sinal recebido desse módulo em relação à pouca atenuação nos demais *dongles* causou pouca separação entre

os pontos de MALA e NÃO MALA. Então, quando realizada nova captura de dados, para teste, em que os valores de RSSI sofreram deslocamento em relação à base de treinamento, a máquina não conseguiu estimar corretamente as regiões dessa nova base. Portanto, nessa topologia o sistema proposto não consegue estimar a região da chave. No entanto, foi observada maior acurácia para estimativas das regiões de LONGE e de NÃO MALA, com acurácia de 100% para ambos.

A partir dessa topologia foi observado que manter todos os *dongles* na parte superior do compartimento interno faz com que ocorra muita similaridade para valores de RSSI recebidos da região em que chave está fora, provocando erros na região de PERTO, que apresentou acurácia em torno de 67%. E como discutido anteriormente, a instalação de *dongle* no compartimento do estepe também não permitiu boa performance do sistema.

Figura 70 – Matriz de confusão de treinamento da Topologia 2 com janela de tamanho 15.

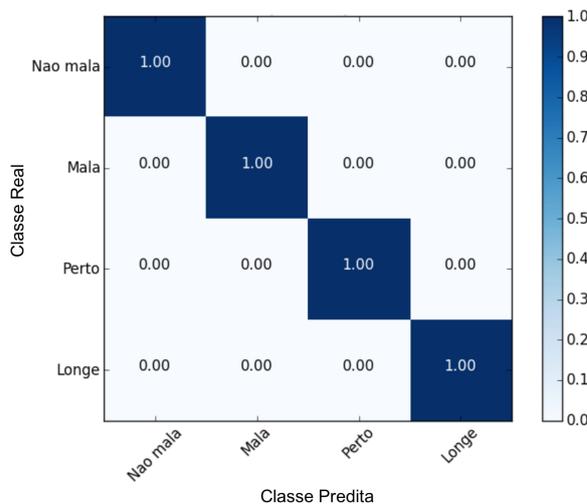


Figura 71 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 15.

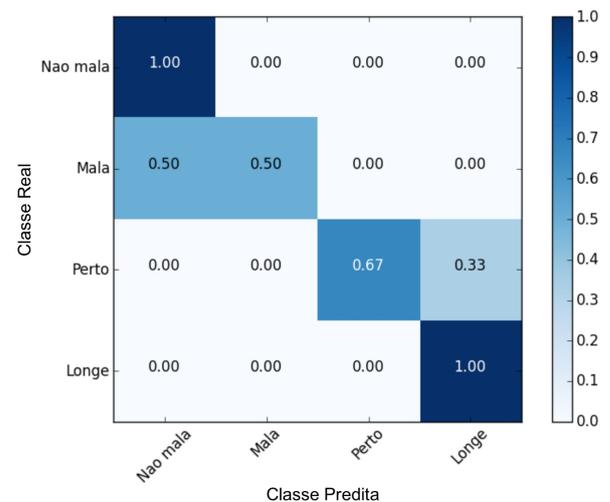


Figura 72 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 20.

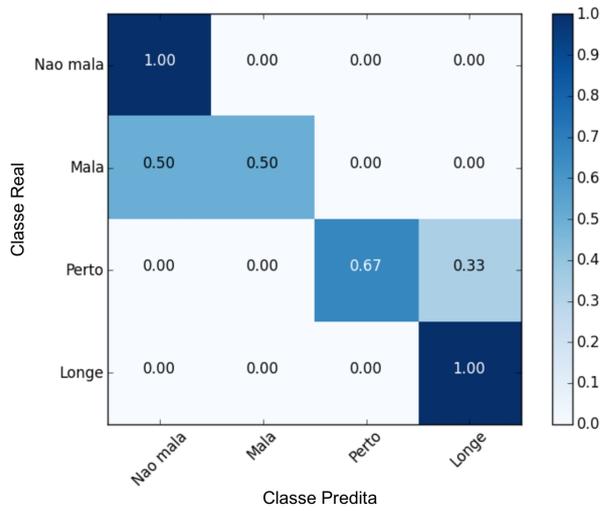
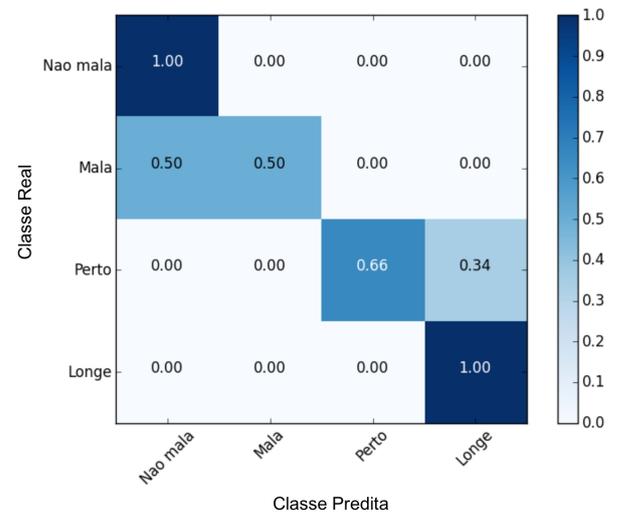


Figura 73 – Matriz de confusão de teste da Topologia 2 com janela de tamanho 25.



6.2.3 Topologia 3

A topologia 3 foi montada posicionando um *dongle* em cada retrovisor externo, um *dongle* no compartimento dos passageiros e um no compartimento da mala, como mostrado na Figura 74. O *dongle* interno foi instalado na parte traseira do apoio de braço central entre os bancos do motorista e passageiro. O instalado na mala ficou mais próximo ao banco de passageiros, centralizado.

Figura 75 – Matriz de confusão de treinamento da Topologia 3 com janela de tamanho 15.

Figura 74 – Topologia 3.

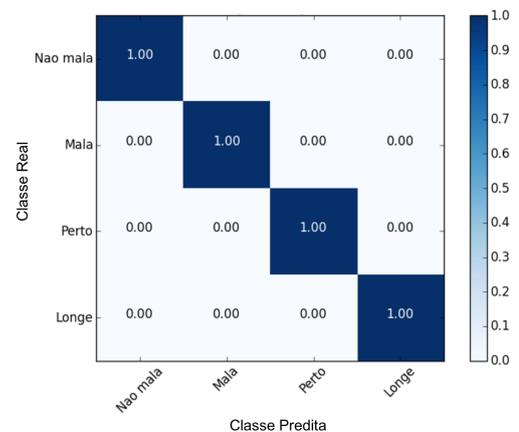
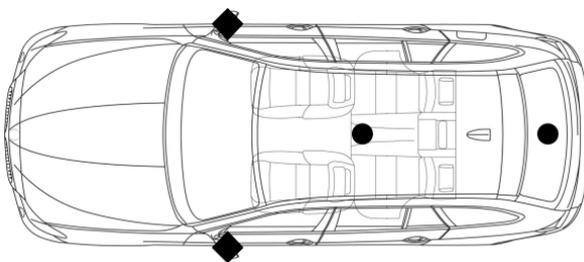


Figura 76 – Matriz de confusão de teste da Topologia 3 com janela de tamanho 15.

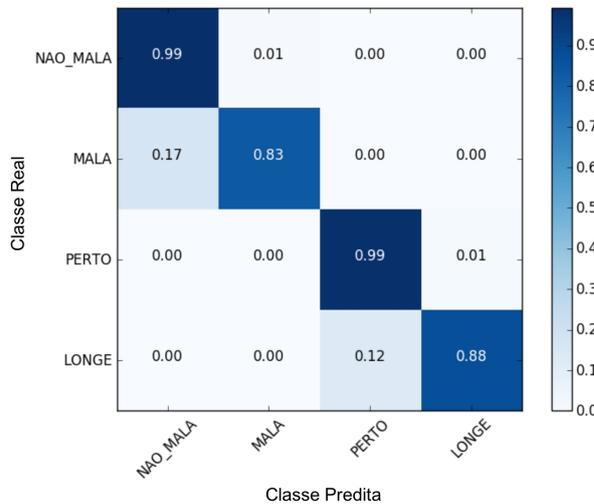
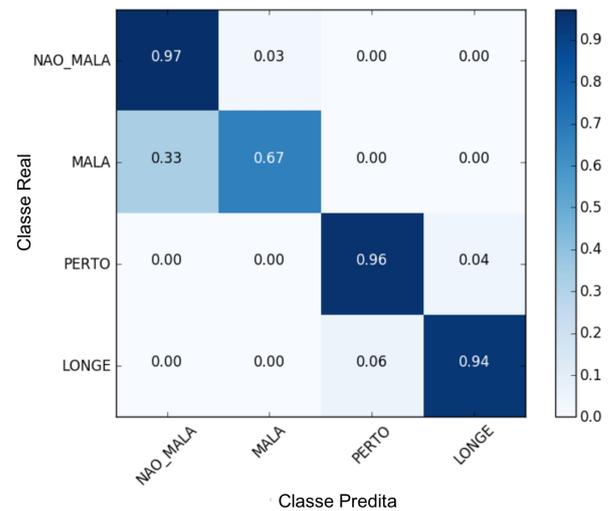


Figura 77 – Matriz de confusão de teste da Topologia 3 com janela de tamanho 20.



Com essa distribuição dos *dongles* foi obtido um ganho de acurácia na definição entre as regiões de MALA e NÃO MALA e entre PERTO e LONGE. Nas Figura 76 e Figura 77, as matrizes de confusão com tamanhos janela de 15 e 20, respectivamente, mostram que a acurácia para região de MALA ficou, no melhor caso, em 83%, para PERTO em 99% e LONGE em 88%.

Essa topologia apresentou melhoria na taxa de acerto em todas as regiões em relação às topologias anteriores. Na próxima topologia, será testado um conjunto com menor número de antenas, três, e usando as observações obtidas das topologias anteriores, como minimizar simetria do conjunto de *dongles* e variando as posições em todas as dimensões para aumentar diferenciação entre valores de RSSI recebidos.

6.2.4 Topologia 4

A topologia 4 foi montada reduzindo a quantidade de *dongles* para três como mostra a Figura 78. Dois foram posicionados no compartimento de passageiros de forma assimétrica e a terceira no compartimento da mala, próximo ao banco traseiro e centralizado. Os dois *dongles* internos na parte superior, um do lado da porta do motorista e outro do lado da porta do passageiro direita.

Figura 78 – Topologia 4.

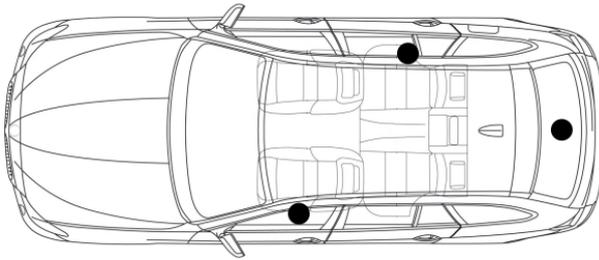
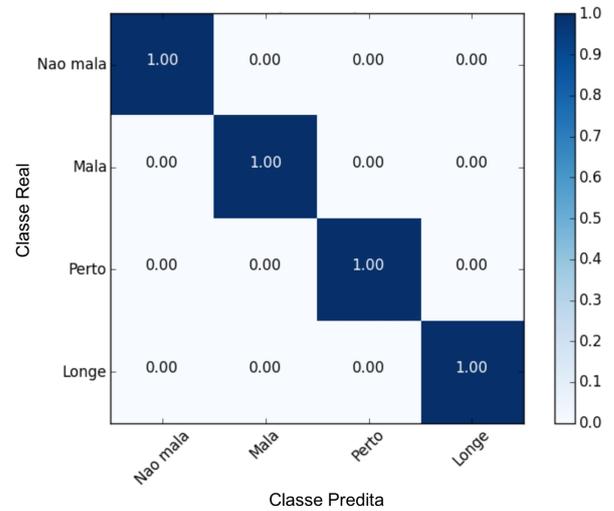


Figura 79 – Matriz de confusão de treinamento da Topologia 4 com janela de tamanho 15.



Os resultados para essa topologia são mostrados nas Figura 80, Figura 81, Figura 82 e Figura 83. Novamente observa-se melhora na acurácia do sistema na definição das regiões. No melhor caso, com tamanho de janela de filtragem igual a 15, foi obtido percentual de acerto de 100% para as regiões de MALA e NÃO MALA, 96% para PERTO e 87% para LONGE. Essa topologia, apesar de fazer uso de somente três antenas, apresentou melhor performance que as demais topologias, aplicando as características de posicionamento obtidas empiricamente. Observa-se, ainda, que o aumento da janela de filtragem não implica melhor performance do sistema.

Figura 80 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 10.

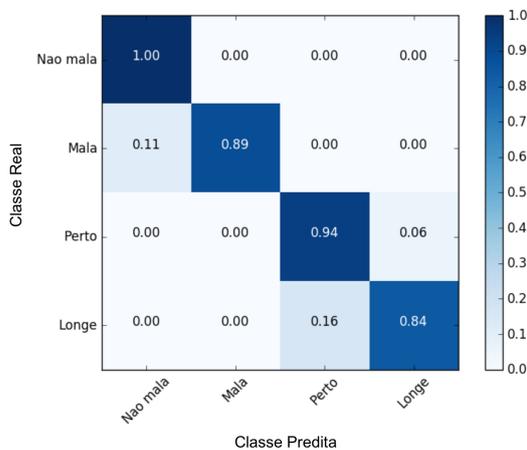


Figura 81 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 15.

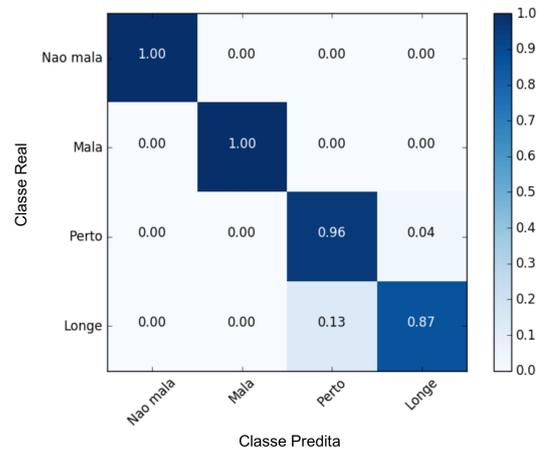


Figura 82 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 20.

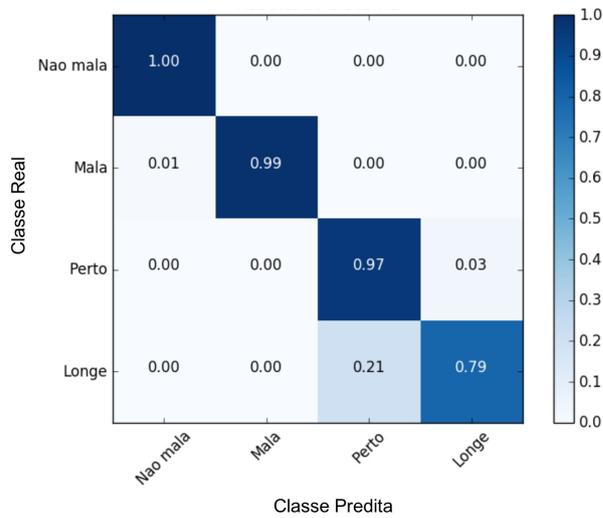
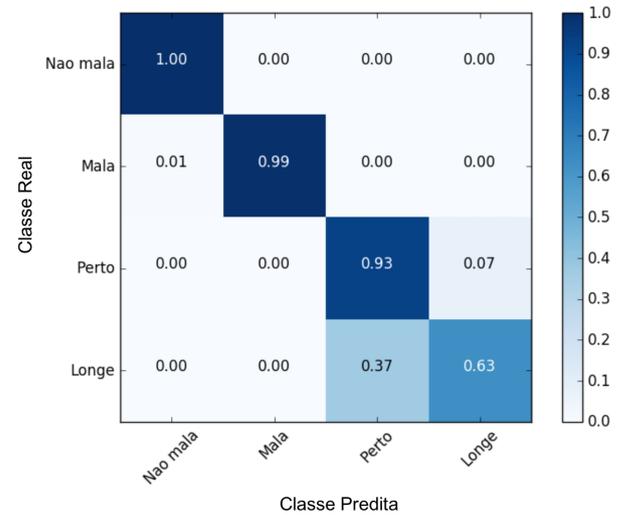


Figura 83 – Matriz de confusão de teste da Topologia 4 com janela de tamanho 25.



6.2.5 Topologia 5

A topologia 5 foi montada posicionando três *dongles* no compartimento dos passageiros e um no compartimento da mala, como mostrado na Figura 84. Dois *dongles* do compartimento de passageiros foram instalados na parte superior do forro a cerca de metade do comprimento das portas. E o terceiro na parte superior acima da porta do passageiro traseiro da esquerda.

Aplicando a característica de assimetria no posicionamento dos *dongles* na topologia, foi testada a topologia dessa forma. Os resultados apresentados para a base de testes construída sob os mesmos pontos de captura da base de treinamento, mas em tempo distinto, foi de 100% de acurácia em todas as regiões, seguindo o mostrado na Figura 85 para a matriz de confusão obtida na fase de treinamento.

Figura 84 – Topologia 5.

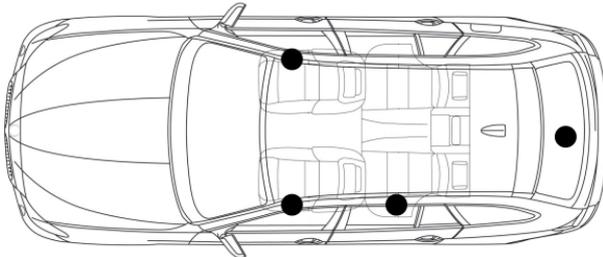
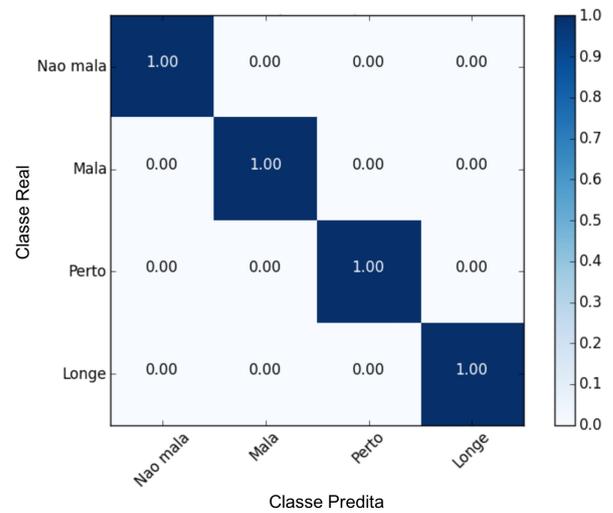


Figura 85 – Matriz de confusão de treinamento da Topologia 5 com janela de tamanho 15.



6.3 DISCUSSÃO

Como discutido nas seções anteriores, o sistema de localização de chave veicular proposto foi testado em cinco topologias com diferentes disposições de dongles no veículo. As bases testadas foram construídas a partir do grid mostrado na Figura 58, podendo haver variação na sua granularidade. Pela Tabela 4, pode-se comparar o sistema proposto com os demais, dos trabalhos relacionados na Seção 4. A forma de expressão da precisão do nosso sistema foi apresentada de forma diferente dos demais pela forma como esse dado foi capturado neste trabalho, sem levar em consideração distância, mas regiões definidas no grid de pontos. O sistema proposto, como falado, utilizou regionalização (divisão da área interna e externa do veículo em regiões) para estimar a localização dos pontos capturados, com as precisões obtidas resumidas na Tabela 8. Ainda em diferença para a maioria dos experimentos comparados que fizeram as coletas e testes em ambientes fechados, o sistema proposto de localização baseado em BLE foi aplicado em automóvel, como em [42].

No entanto, o trabalho citado não apresenta resultados em relação à precisão do sistema apresentado nem sobre quais algoritmos foram utilizados na máquina de aprendizagem, provavelmente para proteção intelectual de projeto comercial vinculado à pesquisa. No sistema aqui proposto, a acurácia para qualquer das

topologias mostradas ficou acima de 75% para testes em pontos fixos e em ambiente controlado, sem intervenção humana.

Em relação à quantidade de *dongles* utilizados, o sistema proposto usou entre 3 e 4, apresentando resultado de acurácia comparável aos demais sistemas, considerando que a região de PERTO, por exemplo, tem no máximo 1,5m e as regiões do interior do veículo tem tamanho ainda mais reduzido e enquanto [42] fez uso de seis a oito *dongles* na aplicação automotiva, gerando um custo maior para a implementação do sistema.

Como mostrado na Tabela 8, em termos de acurácia ou precisão a melhor topologia encontrada neste trabalho foi a topologia 5, com 4 antenas posicionadas de forma assimétrica no Renegade. Em termos de custo, a melhor topologia considerada foi a 4, que com somente 3 antenas obteve melhor performance que todas as demais topologias, exceto a 5. O fator custo é determinante e ponto sensível para a indústria automotiva, na qual a escala de produção é de grande monta. Portanto, qualquer redução de custo em material ou em processos ou sistemas inovadores gera uma economia significativa na cadeia de produção. Dessa forma, a topologia 4 com somente 3 antenas apresenta-se como a mais promissora para possíveis melhorias de acurácia e, assim, implementação no sistema de acesso veicular baseado em *Bluetooth Low Energy*.

Tabela 8 – Comparativo entre topologias testadas; janela de tamanho 15.

TOPOLOGIA	DONGLES	ACURÁCIA MÉDIA
1	4	77,5%
2	4	79,25%
3	4	92,25%
4	3	95,75%
5	4	100%

7 CONCLUSÕES E TRABALHOS FUTUROS

Esta seção tem como objetivo apresentar algumas considerações finais sobre os principais tópicos abordados nesta dissertação, incluindo as contribuições alcançadas e indicações para trabalhos futuros.

7.1 CONSIDERAÇÕES FINAIS E CONTRIBUIÇÕES

Neste trabalho, foi proposto um sistema para localização de chave veicular baseado no método de *fingerprinting*, usando sinais de RSSI da tecnologia de comunicação BLE e usando algoritmos de aprendizagem de máquina para estimar posição da chave, um dispositivo BLE, em relação a regiões definidas no entorno e no interior do veículo, no caso do experimento aqui descrito foi utilizado Jeep Renegado, disponibilizado pela montadora FCA. O sistema proposto apresenta as etapas de captura do sinal, formação da base de dados, aplicação de filtro (janelamento) ao conjunto de sinais, normalização, geração de padrões de entrada para o algoritmo de aprendizagem.

Como dispositivos para teste do sistema foram utilizados os kits de desenvolvimento da fabricante Nordic Semiconductor contendo o SoC nRF52832 que possui a pilha para BLE 4.2. O dispositivo utilizado como chave veicular foi configurado como periférico gerando mensagens de *advertising* a intervalos de 20ms, capturados por outros dispositivos similares atuando com central, aqui chamados de *dongles*. Os sinais capturados eram enviados para um Arduino que, por sua vez, enviava os dados para computador via USB. Esses dados foram utilizados para gerar as bases de treinamento e de teste. O conjunto de dados foi processado usando janela de filtragem, com tamanhos testados entre 10 e 25 amostras por janela, dita com deslocamento no tempo. As médias obtidas eram usadas como entrada do algoritmo de aprendizagem, juntamente com os desvios. Foi proposta uma máquina de aprendizagem com arquitetura em duas camadas, em um total de três máquinas de aprendizado utilizadas. Na primeira camada foi usado

o algoritmo MLP para definir se dado pertence a regiões do interior ou exterior do veículo. Na segunda camada, outra máquina MLP foi utilizada para definir entre as regiões do interior, se o dado pertencia à mala ou ao compartimento de passageiros. Nessa mesma camada, foram testados os algoritmos de árvore de decisão, MLP e kNN para definir se dado no exterior pertencia a região mais próxima ao veículo no limite de 1,5m ou longe a partir desse limite. Após testes realizados, foi escolhido como algoritmo preferencial a árvore de decisão, por apresentar melhor performance e menor uso de memória no Arduino em relação a kNN.

Foram testadas cinco diferentes topologias de posicionamento dos *dongles* no veículo e aplicadas as etapas do sistema para definir empiricamente qual dentre as testadas seriam as mais adequadas para definir posição da chave. Foram usados três ou quatro *dongles* nessas topologias. Observando as acurácias obtidas nos testes das topologias, viu-se que posições particulares dos *dongles* contribuem para melhor performance do sistema, como também a topologia apresentar alguma assimetria. A topologia 1 apresentou acurácia média de 77,25% usando 4 *dongles*, a topologia 2 obteve 79,25% com 4 *dongles*, a topologia 3 apresentou 92,25% com 4 *dongles*, a topologia 4 apresentou 95,75% com 3 *dongles* e a topologia 5 apresentou 100% com 4. Estes resultados foram obtidos com bases de teste capturadas em tempos diferentes, mas nos mesmos pontos do *grid* das bases usadas para treinamento.

7.2 TRABALHOS FUTUROS

Os principais trabalhos que podem ser realizados a partir deste trabalho são:

1. Avaliar aumento ou diminuição do número de *dongles* utilizados na topologia;
2. Realizar testes com chave em pontos distintos da base de treinamento e outros testes em tempo real;
3. Avaliar o impacto do corpo humano na qualidade dos sinais de RSSI e na performance do sistema;
4. Ampliar os tipos de parâmetros de entrada utilizados no algoritmo de aprendizagem: para aumentar robustez do sistema, introduzir uso de acelerômetro e magnetômetro para determinar posição relativa entre

chave e *dongles*, introduzir uso de sensores de temperatura, por exemplo, com tentativa de definir se chave está próxima a corpo humano;

5. Realizar experimentos em outros modelos de veículos.

REFERÊNCIAS

- [1] S. Jung, G. Lee; D. Han. Methods and Tools to Construct a Global Indoor Positioning System. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 6, pp. 906-919, Junho 2018.
- [2] Cantón Paterna, V.; Calveras Augé, A.; Paradells Aspas, J.; Pérez Bullones, M.A. A Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering. *Sensors* 2017, 17, 2927.
- [3] Van Wiemeersch, J. R., Kleve, R. B., Hille, K. T., inventores; Ford Global Technologies, LLC, cessionário. Phone Sleeve Vehicle Fob. United States patent US 9842444 B2. 2017.
- [4] J. Qiu, C. Lin and Y. Tseng, "BLE-based collaborative indoor localization with adaptive multi-lateration and mobile encountering," *2016 IEEE Wireless Communications and Networking Conference*, Doha, 2016, pp. 1-7.
- [5] Y. Wang, Q. Yang, G. Zhang and P. Zhang, "Indoor positioning system using Euclidean distance correction algorithm with bluetooth low energy beacon," *2016 International Conference on Internet of Things and Applications (IOTA)*, Pune, 2016, pp. 243-247.
- [6] J. Pelant *et al.*, "BLE device indoor localization based on RSS fingerprinting mapped by propagation modes," *2017 27th International Conference Radioelektronika (RADIOELEKTRONIKA)*, Brno, 2017, pp. 1-5.
- [7] C. Xiao, D. Yang, Z. Chen and G. Tan, "3-D BLE Indoor Localization Based on Denoising Autoencoder," in *IEEE Access*, vol. 5, pp. 12751-12760, 2017.
- [8] P. Huang and P. Zheng, "BluelD: Enabling robust in-car localization and on-demand personalization using Bluetooth," *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Montreal, QC, 2017, pp. 1-7.
- [9] Mills, J. Keyless wonder: how did we end up with "smart" keys for our cars? *The Sunday Times*, Londres, 8 de novembro de 2014. Disponível em: <<https://www.driving.co.uk/news/keyless-wonder-how-did-we-end-up-with-smart-keys-for-our-cars/>>. Acessado em 3 de agosto de 2018.
- [10] Sodré, E. Há 20 anos, o Brasil reabria os portos aos carros importados. Começava uma revolução. *O Globo*, Rio de Janeiro, 31 de março de 2010. Disponível em <<https://oglobo.globo.com/economia/ha-20-anos-brasil-reabria-os->

portos-aos-carros-importados-comecava-uma-revolucao-3031668#ixzz5PQOrsYtN>. Acessado em 3 de agosto de 2018.

- [11] Glehr, M. inventor; Siemens Aktiengesellschaft, cessionário. Keyless Access Control Device. United States patent US 005723911A. 1998.
- [12] Global Automotive Passive Keyless Entry System Market 2016-2020 [Internet]. Research and Markets. Maio de 2016. Disponível em <<https://www.researchandmarkets.com/reports/3721901/global-automotive-passive-keyless-entry-system#pos-2>>. Acessado em 5 de agosto de 2018.
- [13] Global Automotive Passive Keyless Entry System Market 2016-2020 - Rise in automotive sales to increase demand for PKE systems in emerging markets - Research and Markets [Internet]. Business Wire. Dublin, 24 de maio de 2016. Disponível em <<https://www.businesswire.com/news/home/20160524005910/en/Global-Automotive-Passive-Keyless-Entry-System-Market>>. Acessado em 5 de agosto de 2018.
- [14] T. Glocker, T. Mantere and M. Elmusrati, "A protocol for a secure remote keyless entry system applicable in vehicles using symmetric-key cryptography," *2017 8th International Conference on Information and Communication Systems (ICICS)*, Irbid, 2017, pp. 310-315.
- [15] The Basics of Remote Control and Remote Keyless Entry (Application Note AN-00320). 2012. Disponível no site de Linx Technologies: <<https://linxtechnologies.com/wp/wp-content/uploads/an-00320.pdf>>. Acessado em 5 de agosto de 2018.
- [16] Kamouaa, J. Turn-key passive entry passive start solution. 2013. Disponível no site de Microchip: <http://ww1.microchip.com/downloads/en/DeviceDoc/Article_AC10_Turn-Key-Passive-Entry.pdf>. Acessado em 6 de agosto de 2018.
- [17] Beard, D. The evolution of car keys is more interesting than you think. Car and Driver, Michigan, 25 de dezembro de 2017. Disponível em: <<https://www.caranddriver.com/news/the-evolution-of-car-keys-is-more-interesting-than-you-think>>. Acessado em 7 de agosto de 2018.
- [18] Barry, K. Will your smartphone replace your car key? Consumer Reports, Yonkers, 5 de abril de 2018. Disponível em: <<https://www.consumerreports.org/automotive-technology/will-your-smartphone-replace-your-car-key-virtual-key/>>. Acessado em 7 de agosto de 2018.

- [19] Bluetooth SIG, Inc. Bluetooth Market Update. Disponível em: <<https://www.bluetooth.com/markets/market-report>>. Acessado em 8 de agosto de 2018.
- [20] Heydon, R. Bluetooth low energy: the developer's handbook. 1st edition. Boston: Prentice Hall; 2012.
- [21] J. Lin, T. Talty and O. K. Tonguz, "On the potential of bluetooth low energy technology for vehicular applications," in *IEEE Communications Magazine*, vol. 53, no. 1, pp. 267-275, January 2015.
- [22] R. Faragher and R. Harle, "Location Fingerprinting With Bluetooth Low Energy Beacons," in *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418-2428, Nov. 2015.
- [23] Contreras, D.; Castro M.; de la Torre, D. S. Performance evaluation of bluetooth low energy in indoor positioning systems. *Transactions on Emerging Telecommunications Technologies*, vol. 28. Agosto 2017.
- [24] Argenox Technologies LLC. A BLE Advertising Primer [Internet]. Disponível em: <www.argenox.com/a-ble-advertising-primer>. Acessado em 12 de agosto de 2018.
- [25] Silver, O. An indoor localization system based on BLE mesh network [dissertação]. Linköping: Linköping University; 2016.
- [26] Perez-Diaz-de-Cerio, D.; Hernández-Solana, Á.; Valdovinos, A.; Valenzuela, J.L. A Low-Cost Tracking System for Running Race Applications Based on Bluetooth Low Energy Technology. *Sensors* 2018, 18, 922.
- [27] Nordic Semiconductor ASA. nRF52832 Product Specification v1.4. Outubro de 2017.
- [28] Zafari, F.; Gkelias, A.; Leung, K. A survey of indoor localization systems and Technologies. arXiv:1709.01015v2 [cs.NI]. Março 2018.
- [29] S. Sadowski and P. Spachos, "RSSI-Based Indoor Localization With the Internet of Things," in *IEEE Access*, vol. 6, pp. 30149-30161, 2018.
- [30] F. Seco, A. R. Jimenez, C. Prieto, J. Roa and K. Koutsou, "A survey of mathematical methods for indoor localization," *2009 IEEE International Symposium on Intelligent Signal Processing*, Budapeste, 2009, pp. 9-14.
- [31] R. Zemek, M. Takashima, D. Zhao, S. Hara, K. Yanagihara, K. Fukui, S. Fukunaga, and K. Kitayama, "Effect of walking people on target location estimation performance in an IEEE 802.15.4 wireless sensor network," *IEEE*

- Transactions on Communications, vol. E90-B, no. 10, pp. 2809–2816, Outubro 2007.
- [32] Behboodi, A.; Lemic, F.; Wolisz, A. A mathematical model for fingerprinting-based localization algorithms. arXiv:1610.07636v2 [cs.IT]. Fevereiro 2017.
- [33] Duda, R. O.; Hart, P. E.; Stork, D. G. Pattern Classification. 2nd edition. Nova Iorque: Wiley. 2001.
- [34] Kang, N. Introducing Deep Learning and Neural Networks. Junho 2017. Disponível em: <<https://towardsdatascience.com/introducing-deep-learning-and-neural-networks-deep-learning-for-rookies-1-bd68f9cf5883>>. Acessado em 18 de agosto de 2018.
- [35] M. A. Al-Ammar et al., "Comparative Survey of Indoor Positioning Technologies, Techniques, and Algorithms," 2014 International Conference on Cyberworlds, Santander, 2014, pp. 245-252.
- [36] Daniş, F.S.; Cemgil, A.T. Model-Based Localization and Tracking Using Bluetooth Low-Energy Beacons. *Sensors* 2017, 17, 2484.
- [37] Z. Ma, S. Poslad, J. Bigham, X. Zhang and L. Men, "A BLE RSSI ranking based indoor positioning system for generic smartphones," 2017 Wireless Telecommunications Symposium (WTS), Chicago, IL, 2017, pp. 1-8.
- [38] X. Wang, L. Gao, S. Mao and S. Pandey, "CSI-Based Fingerprinting for Indoor Localization: A Deep Learning Approach," in *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763-776, Jan. 2017.
- [39] M. Altini, D. Brunelli, E. Farella and L. Benini, "Bluetooth indoor localization with multiple neural networks," *IEEE 5th International Symposium on Wireless Pervasive Computing 2010*, Modena, 2010, pp. 295-300.
- [40] N. Kuxdorf-Alkirata and D. Brückmann, "Reliable and low-cost indoor localization based on bluetooth low energy," *2016 3rd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)*, Offenburg, 2016, pp. 92-96.
- [41] S. Ishida, Y. Takashima, S. Tagashira and A. Fukuda, "Proposal of Separate Channel Fingerprinting Using Bluetooth Low Energy," *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, Kumamoto, 2016, pp. 230-233.

- [42] Karlsson, E.; Lagerbielke, A. Demonstrator development for a phone as a key based on Bluetooth Low Energy [dissertação]. Norrköping: Linköping University. 2017.
- [43] Y. Wang, Q. Yang, G. Zhang and P. Zhang, "Indoor positioning system using Euclidean distance correction algorithm with bluetooth low energy beacon," *2016 International Conference on Internet of Things and Applications (IOTA)*, Pune, 2016, pp. 243-247.
- [44] C. Xiao, D. Yang, Z. Chen and G. Tan, "3-D BLE Indoor Localization Based on Denoising Autoencoder," in *IEEE Access*, vol. 5, pp. 12751-12760, 2017.
- [45] J. Qiu, C. Lin and Y. Tseng, "BLE-based collaborative indoor localization with adaptive multi-lateration and mobile encountering," *2016 IEEE Wireless Communications and Networking Conference*, Doha, 2016, pp. 1-7.
- [46] Caratto, D., Piazzano, C., Ginevro, S., inventores; TRW Automotive SRL, cessionário. System of passive entry and passive start for a motor vehicle. United States patent US 9045102 B2. 2015.
- [47] Krishnan, V., inventor; Ford Global Technologies LLC, cessionário. Keyless Handoff Control. United States patent US 2016/0347282 A1. 2016.
- [48] Simmons, M. S., inventor; Directed LLC, cessionário. Smartphone based passive keyless entry system. United States patent US 2015/0048927 A1. 2015.
- [49] Talty, T. J., Kalhous, A. J., Gautama, N. R., inventores; GM Global Technology Operations LLC, cessionário. Vehicle integration of BLE nodes to enable passive entry and passive start features. United States patent US 2014/0240091 A1. 2014.
- [50] Hatton, D. A., inventor; Ford Global Technologies LLC, cessionário. Vehicle computing system in communication with a wearable device. United States patent US 2015/0362997 A1. 2015.
- [51] Sangkim, H., Sang, H., Kijeon, S., Ki, S., Woonghur, N. Woong, N, inventores; Hyundai Motor Company, cessionário. Apparatus and method for controlling vehicle using user authentication. South Korea patent KR1020160059793. 2016.
- [52] Menard, E., Masson, F., Gehin, F., inventores; Valeo Comfort and Driving Assistance, cessionário. Mobile telephone capable of automatically pairing with a motor vehicle and automatic pairing method. United States patent US 2016/0358389 A1. 2016.

- [53]Weghaus, L., inventor; Hella Kga Hueck and Co, cessionário. Access and drive authorization system with positioning capability of an authorization means using a sound-based signal propagation time measurement. Deutschland patent WO 2016/180724 A1. 2016.
- [54]Weizman, R., Broussard, N. S., Gutman, I., Goldenthal, L., inventores; Intel Corp, cessionário. Location BLE beacon. United States patent US 2016/0183042 A1. 2016.
- [55]Hermann, T. J., Van Wiemeersch, J. R., Holub, P. K., inventores; Ford Global Technologies LLC, cessionário. Vehicle key off load reduction via off-board sensor. United States patent US 2016/0288770 A1. 2016.
- [56]Weghaus, L., inventor; Hella Kga Hueck and Co, cessionário. Access and driving authorization system with improved security against relay attacks directed to the transponder interface. Deutschland patent WO 2016/184723 A1. 2016.
- [57]Miyazawa, A., inventor; Alps Electric Co Ltd, cessionário. Keyless entry apparatus. United States patent US 2016/029294 A1. 2016.
- [58]M. Rütshlin and D. Tallini, "Simulation for antenna design and placement in vehicles," *Antennas, Propagation & RF Technology for Transport and Autonomous Platforms 2017*, Birmingham, 2017, pp. 1-5.
- [59]S. Kaul *et al.*, "Effect of Antenna Placement and Diversity on Vehicular Network Communications," *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, San Diego, CA, 2007, pp. 112-121.
- [60]B. Yuan, X. H. Zhang, Z. F. Hu and G. Q. Luo, "Slotted patch antennas with wide axial ratio beamwidth for passive keyless entry system," 2015 IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting, Vancouver, BC, 2015, pp. 1616-1617.
- [61]B. I. Ahmad, T. Ardeshiri, P. Langdon, S. J. Godsill and T. Popham, "Modelling received signal strength from on-vehicle BLE beacons using skewed distributions: A preliminary study," 2017 20th International Conference on Information Fusion (Fusion), Xi'an, 2017, pp. 1-7.
- [62]J. Lin, T. Talty and O. K. Tonguz, "On the potential of bluetooth low energy technology for vehicular applications," in *IEEE Communications Magazine*, vol. 53, no. 1, pp. 267-275, January 2015.