

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

SISTEMA PARA SUPORTE À IMPLANTAÇÃO E MONITORAMENTO
DE APLICAÇÕES DE CAMPUS INTELIGENTE BASEADO EM
PROTOCOLOS DE IoT

FRANKLIN ARTHUR MENDES VENCESLAU

JOÃO PESSOA - PB

2021

FRANKLIN ARTHUR MENDES VENCESLAU

SISTEMA PARA SUPORTE À IMPLANTAÇÃO E MONITORAMENTO
DE APLICAÇÕES DE CAMPUS INTELIGENTE BASEADO EM
PROTOCOLOS DE IoT

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Informática do Centro de Informática da Universidade Federal da Paraíba, como requisito parcial para obtenção do título de Mestre em Informática.

Orientador: Prof. Dr. Iguatemi Eduardo da Fonseca

Co-Orientador: Prof. Dr. Ruan Delgado Gomes

JOÃO PESSOA - PB

2021



5

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

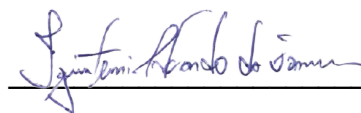


Ata da Sessão Pública de Defesa de Dissertação de Mestrado de Franklin Arthur Mendes Venceslau, candidato ao título de Mestre em Informática na Área de Sistemas de Computação, realizada em 30 de julho de 2021.

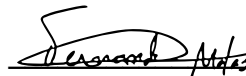
1 Aos trinta dias do mês de julho do ano de dois mil e vinte e um, às nove horas, por meio de
2 videoconferência, reuniram-se os membros da Banca Examinadora constituída para julgar o
3 trabalho do sr. Franklin Arthur Mendes Venceslau, vinculado a esta Universidade sob a
4 matrícula nº 20191000728, candidato ao grau de Mestre em Informática, na área de
5 “Sistemas de Computação”, na linha de pesquisa “Sinais, Sistemas Digitais e Gráficos”, do
6 Programa de Pós-Graduação em Informática, da Universidade Federal da Paraíba. A
7 comissão examinadora foi composta pelos professores: Iguatemi Eduardo da Fonseca
8 (PPGI), Orientador e Presidente da Banca; Fernando Menezes Matos (PPGI), Examinador
9 Interno; Anderson Fabiano B. F. da Costa (IFPB), Examinador Externo à Instituição; e Ruan
10 Delgado Gomes (IFPB), coorientador e Examinador Externo à Instituição. Dando início aos
11 trabalhos, o Presidente da Banca cumprimentou os presentes, comunicou a finalidade da
12 reunião e passou a palavra ao candidato para que ele fizesse a exposição oral do trabalho
13 de dissertação intitulado: “Sistema para Suporte à Implantação e Monitoramento de
14 Aplicações de Campus Inteligente Baseado em Protocolos de IoT”. Concluída a exposição, o
15 candidato foi arguido pela Banca Examinadora que emitiu o seguinte parecer: “**aprovado**”.
16 Do ocorrido, eu, Ruy Alberto Pisani Altafim, Coordenador do Programa de Pós-Graduação
17 em Informática, lavrei a presente ata que vai assinada por mim e pelos membros da banca
18 examinadora. João Pessoa, 30 de julho de 2021


Prof. Dr. Ruy Alberto Pisani Altafim

Prof. Iguatemi Eduardo da Fonseca
Orientador (PPGI-UFPA)



Prof. Fernando Menezes Matos
Examinador Interno (PPGI-UFPA)



Prof. Anderson Fabiano B. F. da Costa
Examinador Externo à Instituição (IFPB)



Prof. Ruan Delgado Gomes
Coorientador (IFPB)



Catálogo na publicação
Seção de Catalogação e Classificação

V451s Venceslau, Franklin Arthur Mendes.

Sistema para suporte à implantação e monitoramento de aplicações de campus inteligente baseado em protocolos de IoT / Franklin Arthur Mendes Venceslau. - João Pessoa, 2021.
199 f. : il.

Orientação: Iguatemi Eduardo da Fonseca.
Coorientação: Ruan Delgado Gomes.
Dissertação (Mestrado) - UFPB/CI.

1. Informática. 2. Campus Inteligente. 3. IoT - Internet das coisas. 4. Protocolo MQTT. 5. RSSF - Redes de sensores sem fio. I. Fonseca, Iguatemi Eduardo da. II. Gomes, Ruan Delgado. III. Título.

UFPB/BC

CDU 004(043)

Agradecimentos

Insegurança, dúvida e incerteza, foram alguns sentimentos que permearam durante maior parte do tempo de desenvolvimento deste trabalho. No entanto, a vontade, a perseverança e o desejo pelo conhecimento superaram todos os obstáculos presentes. Dizem que uma pós-graduação *Stricto Sensu* é um estudo muito solitário. Sabe que não é verdade? Porque ao longo de todo este período eu tive o privilégio de contar com uma lista imensurável de pessoas que me apoiaram em praticamente todos os momentos inerentes à condução deste estudo.

Gratidão primeiramente a Deus, pelo dom da sabedoria, capacidade e por tornar tudo possível.

Gratidão a minha esposa, Nohara, que foi a válvula propulsora de tudo isso. Obrigado, amor. Pelo companheirismo de sempre, por estar ao meu lado em todos os momentos e por aguentar minhas infinitas crises de ansiedade. Gratidão a minha família, a minha mãe, ao meu pai (que está vendo tudo lá de cima) e meus irmãos por sempre acreditarem no meu potencial e capacidade. Gratidão a minha cadelinha Kika, por me fazer companhia em diversos momentos de estudo solitário, tornando assim estes momentos menos entediantes e mais divertidos.

Agradecimento especial aos meus orientadores Iguatemi e Ruan: Prof. Iguatemi, por sua paciência ímpar, por tornar tensas reuniões de mestrado (para mim) em praticamente uma sessão de coaching pessoal, com tanta sensibilidade, dicas, conselhos, sugestões não somente para a vida acadêmica, mas para o meu próprio desenvolvimento pessoal. Iguatemi, meu mais sincero agradecimento por tudo, obrigado por sempre confiar e acreditar no meu potencial, em diversos momentos em que nem eu mesmo acreditava. E pode ter certeza de que nunca vou esquecer da questão da curva de aprendizado, como você sempre diz, no começo são só troços, mas quando a gente supera essa curva, tudo passa a fluir mais fácil e foi exatamente assim. Prof. Ruan, muitíssimo obrigado por todo o suporte dado na elaboração deste trabalho, por todas as dicas, conselhos e correções. Pelas intermináveis mensagens que trocamos, em especial na fase do experimento, que

muitas vezes, sempre com sua disponibilidade ímpar, tirasse minhas dúvidas em dias de semana, sábados, domingos, feriados, muitas vezes passando até das 23h. Deixo aqui minha gratidão por tudo! Um privilégio luxuoso para mim ser coorientado por um professor do seu calibre.

Agradeço a Vitor Marques, por todo apoio durante a condução deste estudo. Agradeço a Felipe Ferreira, por me mostrar as coisas incríveis que podemos fazer com um OpenMote B. Ao corpo de docentes do CI, em especial aos professores Alisson Brito e Natasha Lino.

Aos membros da banca avaliadora, Prof. Fernando e Prof. Anderson, por todas as correções, críticas e conselhos dados para que este trabalho pudesse ficar mais consistente.

A Vitor Marinho, Camila Jácome, Lucas Pereira, Fabrício Leite e todos os colegas do PPGI, pelo apoio mútuo. A Mônica Saramago da SENSORNETS por todas as orientações dadas no meu primeiro congresso internacional como palestrante.

Agradecimento igualmente especial a CAPES por financiar boa parte das pesquisas aqui conduzidas.

Por fim, a vida é um desafio constante e os desafios existem para serem superados.

"Um barco, no ancoradouro, está seguro. Mas não é para isso que os barcos são feitos."

William Shedd

Resumo

Nos últimos anos vem-se observando um crescimento acentuado do conceito de Internet das Coisas (IoT) como uma infraestrutura para monitorar e gerenciar com eficiência o ambiente estudado. A modernização das cadeias de produção, em especial a indústria 4.0, foi determinante para a popularização dos chamados “sistemas de monitoramento inteligentes”. No entanto, até recentemente, os dispositivos IoT eram limitados pela capacidade computacional, especialmente por fatores como largura de banda limitada e comunicação sem alto grau de confiabilidade. Tomando-se por base essa premissa, este trabalho propõe a concepção de uma arquitetura e um sistema para monitoramento de aplicações de Campus Inteligente baseado em protocolos de IoT. Esse sistema tem como principais objetivos dar suporte à implantação de novos sensores e atuadores, além de realizar o monitoramento do desempenho das aplicações. A arquitetura de rede considerada neste trabalho provê confiabilidade por meio do uso de técnicas de diversidade em nível de camada física e enlace e realiza a persistência de informações baseado em um banco de dados séries temporais. Por meio desse sistema, informações podem ser coletadas sobre as aplicações atualmente em execução e sobre a qualidade dos enlaces e nível de sobrecarga da rede sem fio, de modo a ajudar no planejamento para evolução da infraestrutura de rede e incorporação de novos dispositivos e aplicações. Como forma de validação da arquitetura proposta, foi realizado um experimento e uma rede baseada no padrão IEEE 802.15.4g foi implantada em um Campus Universitário.

Palavras-chave: Campus Inteligente, IoT, MQTT, RSSF

Abstract

Over recent years a sharp growth in the concept of Internet of Things (IoT) has been observed as an infrastructure to effectively monitor and manage the studied environment. The modernisation of production chains, in particular Industry 4.0, was crucial to the popularization of the so-called “Intelligent Monitoring Systems”. However, until recently, IoT devices were limited by computational capability, mainly by factors such as limited bandwidth and communication without a high degree of reliability. Based on this premise, this paper proposes a system for monitoring Smart Campus applications based on IoT protocols. This system will have as main objectives to support the deployment of new sensors and actuators and to monitor the performance of applications. The network architecture considered in this paper provides reliability through the use of diversity techniques at physical and data link layers and it performs the persistence of information based on time series. Through this system, information on the currently running applications and on the quality of the data links and the wireless network’s overload level can be collected, in order to help planning for the network infrastructure evolution and the incorporation of new devices and applications. To validate the proposed architecture, an experiment was performed and a IEEE 802.15.4g network was deployed in an University Campus.

Keywords: Smart Campus, IoT, MQTT, WSNs.

Sumário

1	Introdução	18
1.1	Contextualização	18
1.2	Proposta de Pesquisa	20
1.3	Objetivo Geral	22
1.4	Objetivos Específicos	22
1.5	Estrutura da Dissertação	23
2	Fundamentação Teórica	25
2.1	Redes de Sensores Sem Fio	25
2.2	Aplicações em Campi Inteligentes	27
2.3	Padrão IEEE 802.15.4	29
2.3.1	IEEE 802.15.4g	32
2.4	Wireless Smart Utilities / Ubiquitous Networks	36
2.5	InfluxDB	39
2.6	Protocolo MQTT	41
2.6.1	Principais Parâmetros e Níveis de QoS	44
2.7	Protocolo CoAP	45
2.7.1	Modelo de Solicitação/Resposta	46

3	Trabalhos Relacionados	48
4	Sistema de Gerenciamento de Aplicações de Smart Campi	54
4.1	Arquitetura do Sistema	54
4.1.1	Descrição da Arquitetura Lógica	54
4.1.2	RAW Manager	57
4.2	Definição da Arquitetura Física e Ambiente de Validação	59
4.3	Tecnologias e Componentes Utilizados	60
4.3.1	OpenMote B	62
4.3.2	Raspberry Pi 4	62
4.4	Implementação do RAW-Manager	63
4.4.1	Transmissão de Dados	64
4.4.2	Escrita no Banco com Saída Bidirecional	65
4.4.3	Monitoramento em Página WEB	66
5	Validação do Sistema Proposto	68
5.1	Preparação do Ambiente	68
5.2	Métricas de Avaliação de Desempenho	69
5.3	Condução do Experimento	70
5.4	Análise dos Resultados	75

5.4.1	Análise em Camada Física	75
5.4.2	Análise em Camada de Aplicação	78
5.4.3	Análise em Camada de Aplicação com Gateways Combinados . . .	81
5.4.4	Análise do RSSI	83
5.4.5	Análise de PDR em Função do Tempo e Condições Meteorológicas .	85
5.4.6	Monitor de Rede e Comunicação via MQTT	87
6	Conclusão	89
6.1	Considerações Finais	89
6.2	Publicação Durante o Mestrado	89
6.3	Trabalhos Futuros e Dificuldades Encontradas	90

Lista de Abreviaturas

AES : Advanced Encryption Standard

AMQP : Advanced Message Queuing Protocol

API : Application Programming Interface

CoAP : Constrained Application Protocol

CSMA-CA: Carrier Sense Multiple Access With Collision Avoidance

Dp : Desvio Padrão

DSSS : Direct Sequence Spread Spectrum

FIFO : First-in, First-out

FSK : Frequency Shift Keying

GSM : Global System for Mobile Communications

HTML : HyperText Markup Language

HTTP : Hypertext Transfer Protocol

IBM : International Business Machines Corporation

IIoT : Industrial Internet of Things

IoT : Internet of Things

ISM : Industrial Scientific Medical

JSON : JavaScript Object Notation

KMP : Protocolo de Gerenciamento de Chaves

MAC : Camada de Acesso ao Meio

MQTT : Message Queuing Telemetry Transport

NES-COM : Comitê de Novos Padrões

O-QPSK : Modulação por Deslocamento de Fase

OFDM : Orthogonal Frequency-Division Multiplexing

P2P : Ponto a Ponto

PDR : Packet Delivery Ratio

PHY : Physical Level

PPDU : Unidade de Dados da Camada Física

RAW : Reliable and Available Wireless

RSSI : Received Signal Strenght Indication

SBC : Single Board Computer

SMS : Short Message Service

SQL: Structured Query Language

SUN : Smart Utility Network

TG4 : Task Group 4

TSCH: Time Slotted Channel Hopping

Tx_Co : Compilação de Transmissores

Wi-SUN : Wireless Smart Utilities / Ubiquitous Networks

WPAN : Wireless Personal Area Networks

XML : Extensible Markup Language

XMPP : Extensible Messaging and Presence Protocol

Lista de Figuras

2.1	Exemplo da Estrutura de uma RSSF	26
2.2	Topologias Mais Utilizadas em RSSF	27
2.3	Camadas do Padrão IEEE 802.15.4	30
2.4	Principais Arranjos Topológicos	33
2.5	Estrutura de Camadas do Padrão IEEE 802.15.4g	34
2.6	Modulação FSK	34
2.7	Modulação OFDM	35
2.8	Exemplo Wi-SUN com Multi-Hop	37
2.9	Infraestrutura da Sub-Rede HAM	38
2.10	Tempo Médio de Execução por Número de Consulta	40
2.11	Tempo Médio de Execução para Todas as Consultas	40
2.12	Representação do Broker MQTT	42
2.13	Modelo Representativo de Publicação-Assinatura do MQTT para IoT . . .	43
2.14	Representação dos Métodos GET e ACK	46
2.15	Representação do Método GET com Resposta Separada	47
4.1	Comunicação Entre os Nós Finais e o Gateway	55
4.2	Detalhamento dos métodos de Comunicação entre o Nó Final e o <i>Gateway</i>	57

4.3	Detalhamento do RAW Manager com troca de Mensagens via MQTT . . .	58
4.4	Gateway Internamente.	60
4.5	Delimitação Geográfica da Área Teste	61
4.6	Exemplo de um OpenMote B	63
4.7	Raspberry Pi 4 Modelo B	63
4.8	Representação do ciclo de transmissão	64
4.9	Comunicação com saída bidirecional	66
5.1	Arquitetura no Domínio do Experimento	69
5.2	Distribuição dos Nós no Campus V da UFPB	71
5.3	Arquitetura Interna dos Nós Finais	72
5.4	Nós Finais Alocados Internamente.	73
5.5	Nó Final Externo e Gateway Receptor	73
5.6	Análise em Camada Física no Gateway 1	76
5.7	Análise em Camada Física no Gateway 2	77
5.8	Compilação dos Nós Finais em Camada Física	78
5.9	Análise em Camada de Aplicação no Gateway 1	79
5.10	Análise em Camada de Aplicação no Gateway 2	80
5.11	Compilação dos Nós Finais em Camada de Aplicação	81

5.12	Análise com Gateways Combinados	82
5.13	Análise do RSSI no Gateway 1	84
5.14	Análise do RSSI no Gateway 2	85
5.15	Análise do RSSI Compilado	86
5.16	Dados Recebidos em Função do Horário	87
.1	Monitor de Rede Construído.	92
.2	Consumo de dados via MQTT.	93
.3	Visão Diagonal.	94
.4	Visão Lateral.	94
.5	Gateway 01.	95
.6	Gateway 02.	95

Lista de Tabelas

2.1	Aspectos da Camada Física do Padrão IEEE 802.15.4	31
3.1	Relação de Trabalhos Relacionados	52

1 Introdução

1.1 Contextualização

O desenvolvimento das redes sem fio proporcionou grandes avanços na forma de comunicação entre dispositivos geograficamente distribuídos. Mesmo com a evolução dos meios de transmissão guiados, tornando assim o cabeamento cada vez mais maleável, em locais de difícil acesso, a alocação do cabeamento enfrenta ainda grandes dificuldades. As redes sem fio surgiram então como uma alternativa que possibilita economia de recursos, diminuição da complexidade de instalação, flexibilidade para modificações de topologia, bem como uma maior facilidade para gestão e controle de incidentes [Tanenbaum, 2010].

Nos últimos anos, vem-se observando um crescimento bastante acentuado do conceito de Internet das Coisas (IoT) como uma infraestrutura para monitorar e gerenciar com eficiência um determinado ambiente a ser analisado. De acordo com o *Abi Research Institute*, ao final de 2021 estima-se que existirão por volta de 30 bilhões de dispositivos sem fio conectados à Internet [Intelligence, 2018].

Posteriormente, com o amadurecimento das tecnologias de transmissão sem fio, em especial o desenvolvimento da IoT (*Internet of Things*), as chamadas RSSF (Redes de Sensores Sem Fio) passaram a ter grande aplicação na indústria, agricultura, medicina e nas *Smart Grids* (Redes Elétricas Inteligentes), entre outras aplicações. Apoiados pela internacionalização dos padrões mais conhecidos, por intermédio da Comissão Eletrotécnica Internacional (IEC): WirelessHART (IEC 62591), ISA 100.11a (IEC 62734) e WIA-PA (IEC 62601) [Wang and Jiang, 2016] e do IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos): IEEE 802.15.4 [Chen et al., 2014].

Atualmente, diversas aplicações de cidades e campi inteligentes vêm sendo propostas, a exemplo: controle e segurança, iluminação inteligente, além de monitoramento ambiental, tráfego veicular e de outros recursos diversos [Ejaz et al., 2017]. A maior parte dessas aplicações requerem o uso de transmissão sem fio para comunicação entre os sistemas de sensoriamento distribuídos pela cidade/campus até um servidor centralizado,

onde, normalmente ocorre o gerenciamento das informações obtidas. Diante desta perspectiva, muitas vezes as soluções para transmissão de dados já existentes, como Wi-Fi, Bluetooth e ZigBee não são adequadas por limitações de alcance de comunicação. No entanto, uma alternativa pode ser o uso de redes de longo alcance e de baixa potência, como LoRa ou SigFox [Vejlgaard et al., 2017]. Ambos os padrões operam no espectro Sub-GHz, atuando com eficientes técnicas de modulação que possibilitam a criação de redes esparsas ou com topologia em estrela, esquema este mais comum e de mais fácil gerenciamento e manutenção.

Ainda assim, estas soluções não são ideais por tratar-se de tecnologias proprietárias e que possuem uma flexibilidade limitada no que envolve requisitos críticos de comunicação em diferentes dispositivos. Dentre estas limitações podemos citar: no caso do SigFox, a dependência de um provedor externo de serviços além da sua baixa taxa de transmissão, cerca de 600 bps em condições ideais [Vejlgaard et al., 2017]. Por sua vez, LoRa é uma opção parametrizável, de acordo com o fator de espalhamento, porém a taxa máxima de bits é de 21,9 kbps, na faixa licenciada no Brasil [Vejlgaard et al., 2017]. Outra alternativa para implementação de redes LPWAN é o padrão IEEE 802.15.4, que incluiu um adendo em nível de camada física (IEEE 802.15.4g) propondo três diferentes esquemas de modulação e visando aplicações de redes de utilidade inteligentes (*Smart Utility Networks - SUN*), sendo elas: SUN-FSK (*Frequency-Shift Keying*), SUN-OQPSK (*Offset Quadrature Phase-Shift Keying*) e SUN-OFDM (*Orthogonal Frequency-Division Multiplexing*) [Tuset-Peiró et al., 2020]. Diferentemente do LoRa, este padrão é aberto e oferece maior nível de flexibilidade, uma vez que 31 configurações diferentes de camada física são suportadas com taxas de bit variando entre 6,25 kbps a 800 kbps [Muñoz et al., 2018]. Além do fato de operar tanto no espectro Sub-GHz como na faixa de 2.4 GHz.

No entanto, diversos desafios são encontrados na comunicação em Redes de sensores sem fio (RSSF), independente da tecnologia escolhida. Dentre estes principais aspectos podemos citar:

i) em princípio a comunicação sem fio está sujeita a problemas de propagação de sinal, sendo os mais comuns: sombreamento, atenuação por multipercurso e interferências diversas que provocam altos índices de atenuação. Além deste aspecto, a comunicação

sem fio muitas vezes aparece como única alternativa por questões relacionadas a limitação orçamentária, ou ainda, inexistência de infraestrutura que comporte a alocação de cabeamento local;

ii) outro aspecto relevante a ser destacado, é que diferentes dispositivos e aplicações com diferentes requisitos de comunicação podem coexistir. Deste modo, torna-se de grande importância a capacidade de compartilhamento da infraestrutura de rede, quando possível, objetivando-se a redução de custos de implementação de novos sensores ou atuadores.

Objetivando a construção de um cenário para alocação de diversas aplicações de *Smart Campus*, uma arquitetura de rede foi concebida, com o foco em prover comunicação confiável por meio da possibilidade de aplicação de técnicas de diversidade a nível de modulação e de receptor, conceitos estes propiciados pelo padrão IEEE 802.15.4-2015, e que oferece ainda a capacidade de uso de diferentes esquemas de modulação e configurações em um único transceptor. Ao mesmo tempo, é de suma importância existir uma forma de gerenciar a implantação e evolução dessa infraestrutura de rede, bem como prover meios para monitorar as aplicações que estão executando em um determinado instante, para oferecer informações sobre o atendimento a requisitos de confiabilidade e sobre a viabilidade de inserção de novos dispositivos na rede. Nesse contexto, esta dissertação consiste na proposta de um modelo arquitetural de comunicação e no desenvolvimento de um sistema de gerenciamento que se conecta aos *gateways* para configuração e monitoramento da rede, bem como para oferecer informações importantes para dar suporte à implantação de novas aplicações para evolução da rede.

1.2 Proposta de Pesquisa

Diante do exposto e visando lidar com os desafios descritos anteriormente, esta dissertação propõe o desenvolvimento de um sistema para gerenciamento de aplicações de *Smart Campus*, visando fornecer suporte à implantação e evolução da rede sem fio que oferece conectividade a essas aplicações, bem como monitorar o desempenho das aplicações e a viabilidade de inserção de novos dispositivos na rede, com base em dados coletados a partir do funcionamento atual da rede. Nesse contexto, uma arquitetura de

rede que usa diferentes estratégias de redundância, com base no Padrão IEEE 802.15.4-2015, é considerada, para prover alta confiabilidade e disponibilidade. O sistema proposto nesta dissertação é denominado RAW-Manager (*Reliable and Available Wireless*).

Nesta arquitetura de rede temos os nós finais se comunicando com as aplicações executadas na nuvem por meio de *gateways* e eles podem transmitir os pacotes usando diferentes esquemas de modulação, por meio de um único transceptor. Os *gateways* são configurados com múltiplos transceptores e são capazes de receber diversos pacotes de modo simultâneo, utilizando configurações de modulação distintas. Essa arquitetura foi inicialmente proposta em [Tuset-Peiró et al., 2020], no entanto, nesse trabalho apenas três configurações de camada física foram consideradas e um único *gateway* foi utilizado. O sistema proposto nesta dissertação irá considerar um cenário mais genérico, em que configurações distintas de camada física podem ser utilizadas, além da possibilidade do sistema operar com múltiplos *gateways*. A ideia por trás desta perspectiva seria utilizar a melhor configuração possível obtida por cada variante de modulação em camada física. Por exemplo: em uma transmissão de dados utilizando um tipo de modulação mais indicado para quando o alcance de comunicação seja um fator crítico. Paralelamente teríamos a possibilidade de usar redundância através de outra configuração de modulação mais propícia para situações em que a taxa de transferência de bits seja, por sua vez, o fator crítico. Além destes aspectos, técnicas para tratamento de duplicação de pacotes e persistência de dados são consideradas.

Para prover a comunicação desde os nós finais até as aplicações de IoT que irão consumir a partir dos dados obtidos, serão explorados os protocolos CoAP e MQTT. Cada um responsável por um trecho da arquitetura de comunicação. Respectivamente o primeiro, para o trecho de comunicação que não conta com conexão à Internet. Neste trecho, a transmissão sem fio é implementada pelo uso do padrão IEEE 802.15.4g. Já o segundo trecho, entre o *gateway* centralizador até as aplicações finais de IoT, a comunicação é provida por meio do modelo *publish-subscriber* do protocolo MQTT.

A validação foi realizada por meio da implantação de uma rede composta por oito nós finais e dois *gateways*, sendo cada um destes dispositivos alocados em pontos estratégicos. Foram avaliadas métricas relacionadas à qualidade da transmissão entre tais pontos.

Sendo o sistema de suporte o cerne desta proposta de estudo, a estratégia de validação da rede sensores consiste em um estudo de caso, voltado a avaliação da qualidade da comunicação entre os nós finais e *gateways* da rede, por meio do uso de técnicas de diversidade na camada física. Esta avaliação da qualidade na comunicação utiliza parâmetros comumente utilizados na avaliação de desempenho em RSSF, tais como: PDR (*Packet Delivery Ratio*) e RSSI (*Received Signal Strength Indication*). Após a aferição dos dados obtidos, o sistema propõe também o adequado tratamento e gerenciamento das informações obtidas, além do registro em um sistema de banco de dados baseado em séries temporais com a apresentação em uma plataforma *web*. Ao final, espera-se que o sistema permita a construção de uma rede de sensores sem fio base apta a receber outros dispositivos de sensoriamento em larga escala e que possam monitorar distintos fenômenos relevantes a um ambiente de Campus Inteligente.

1.3 Objetivo Geral

O objetivo geral deste trabalho é propor e implementar um sistema para gerenciamento de aplicações de *Smart Campus* que ofereça suporte à implantação de novos sensores e atuadores, além da evolução da rede sem fio que atribua conectividade a essas aplicações, bem como monitorar o desempenho da rede e das aplicações.

1.4 Objetivos Específicos

- Desenvolver um módulo de software para permitir realizar o gerenciamento remoto dos *gateways* e nós finais, para permitir a gestão de vários parâmetros de configuração dos rádios, tais como: modulação, canal e potência de transmissão;
- Implementar uma funcionalidade para proceder com a centralização, tratamento adequado, preparação e limpeza de duplicações e persistência das informações geradas e coletadas pelos múltiplos *gateways* da rede;
- Realizar a integração da rede com protocolos padronizados de camada de aplicação (ex: CoAP e MQTT), para permitir o desenvolvimento de aplicações de Internet

das Coisas;

- Implementar um monitor de rede, para permitir acompanhar a quantidade de pacotes recebidos pelos *gateways* em cada módulo de rádio e a qualidade dos enlaces utilizando diferentes métricas para os diferentes nós finais;
- Realizar análise nos dados coletados a partir da rede para permitir a construção de modelos e *Datasets* que caracterizem o desempenho da rede em sua configuração atual;
- Realizar testes de desempenho do sistema, por meio da implantação de protótipos de nós finais e *gateways* em ambiente de Campus Universitário.

1.5 Estrutura da Dissertação

O escopo desta dissertação está estruturado da seguinte forma:

- Capítulo 1 - Introdução: contextualização do problema de pesquisa, apresentação geral da proposta de estudo e delimitação das etapas necessárias para a busca da solução do problema evidenciado.
- Capítulo 2 - Fundamentação Teórica: apresentação dos tópicos mais relevantes da revisão de literatura realizada objetivando um melhor entendimento acerca dos temas envolvidos neste estudo;
- Capítulo 3 - Trabalhos relacionados: apresentação de estudos semelhantes que foram úteis para a produção deste estudo;
- Capítulo 4 - Descrição do Sistema de Gerenciamento: apresentação da metodologia, do sistema proposto na dissertação e as etapas necessárias para sua execução;
- Capítulo 5 - Validação do Sistema Proposto: descrição das métricas de avaliação de desempenho, projeto, preparação e descrição detalhada das etapas inerentes ao experimento realizado, com a análise dos resultados coletados.

- Capítulo 6 - Conclusões e Perspectivas de Trabalhos Futuros: considerações finais, principais contribuições do trabalho, projeções futuras e dificuldades encontradas.

2 Fundamentação Teórica

2.1 Redes de Sensores Sem Fio

Uma Rede de Sensores sem Fio (RSSF) consiste em uma ramificação das redes *ad-hoc*, subcategoria das redes sem fio, que não necessitam, necessariamente, de uma estação base, o qual exerce atividades de comutação entre os nós que compõem a rede. Sua principal função consiste em medir e monitorar a ocorrência de determinados tipos de fenômenos físicos ou ambientais.

As redes de sensores formadas pelos nós, possuem atualmente vasta aplicação na monitoração de fenômenos, permitindo a alocação de sensores em locais remotos e de difícil acesso. Alguns componentes básicos são comuns nas arquiteturas de RSSF, tais como: i) o próprio sensor, dispositivo que “percebe” o ambiente por meio da resposta a estímulos físico/químicos; ii) o fenômeno, grandeza ou acontecimento observável e capaz de ser medido; iii) a memória, dispositivo capaz de realizar o registro dos fenômenos mensurados; iv) o microcontrolador, hardware programável base que adquire as informações dos sensores, processa os dados, armazenando em memória e que geralmente é alimentado por bateria [Ruiz et al., 2004].

A Figura 2.1 representa a abrangência geográfica do campo de sensoriamento composto pelos nós sensores. Estes comunicam-se com os nós sorvedouros que fazem ligação direta com o agente responsável pela comunicação com a rede externa, que possibilita o monitoramento remoto.

Atualmente as RSSF são aplicáveis nas mais distintas áreas por meio do monitoramento em tempo real de fenômenos e grandezas. Como exemplo destas aplicações podemos citar:

- ambiente industrial: substituindo as redes cabeadas convencionais para mensurar níveis de temperatura, pressão e grandezas elétricas, tais como: potência, corrente e tensão;

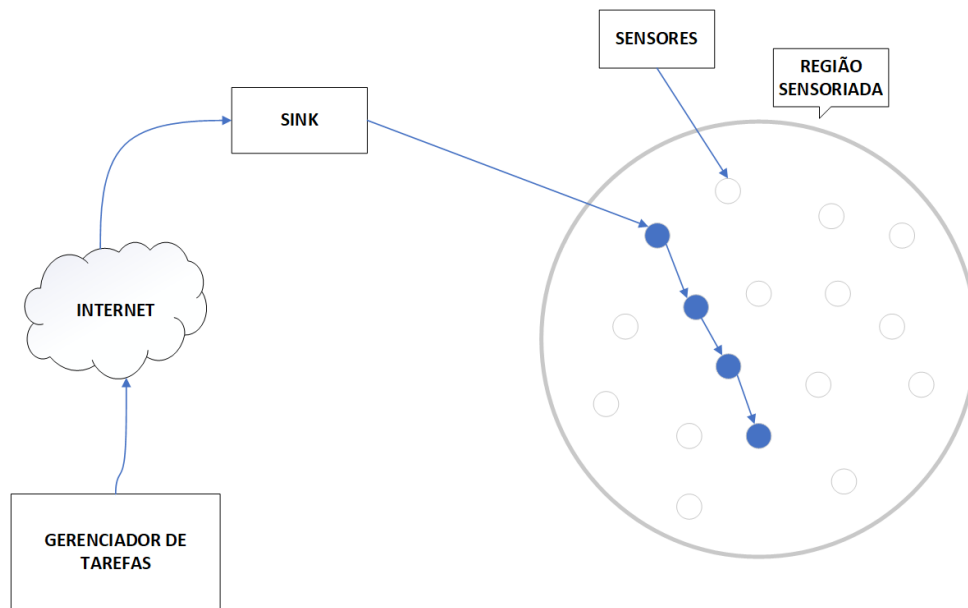


Figura 2.1: Exemplo da Estrutura de uma RSSF. Adaptado de: [GTA-UFRJ, 2014]

- medições climáticas: observação de índices pluviométricos, pressão atmosférica, luminosidade, umidade;
- trânsito inteligente: monitoramento das condições de trânsito, das vias, variação de fluxo, previsão de horários de pico;
- agricultura inteligente: observação das condições do solo, presença de predadores, luminosidade e variação de temperatura ao longo do dia;

Diversas configurações de topologia são passíveis de utilização nas RSSF. No entanto, três configurações são mais utilizadas: i) estrela: neste modo de operação, os nós alocados em diferentes pontos estabelecem ligações do tipo ponto a ponto com o *gateway* ou receptor central. ii) híbrido ou árvore/cluster: neste modo temos configurações híbridas que permitem ligações intermediárias com o objetivo de aumentar o alcance do sinal de transmissão. Assim, cada nó pode se comunicar diretamente a outros nós, repetidores ou roteadores intermediários. Esta configuração permite larga ampliação do alcance, porém, em caso de falha de algum roteador intermediário, toda a sub-rede pertencente a este ficará indisponível. iii) mesh: este modo configura-se como o mais seguro, confiável e tolerante a falhas, porém, o mais custoso. O custo de implementação de uma RSSF com este padrão de topologia cresce exponencialmente ao efetivo número de nós. [Mainwaring et al., 2002]

A Figura 2.2 faz uma representação das topologias mais utilizadas em RSSF, mostrando os esquemas de ligação entre eles e ilustrando os dispositivos que compõem a configuração em nível físico.

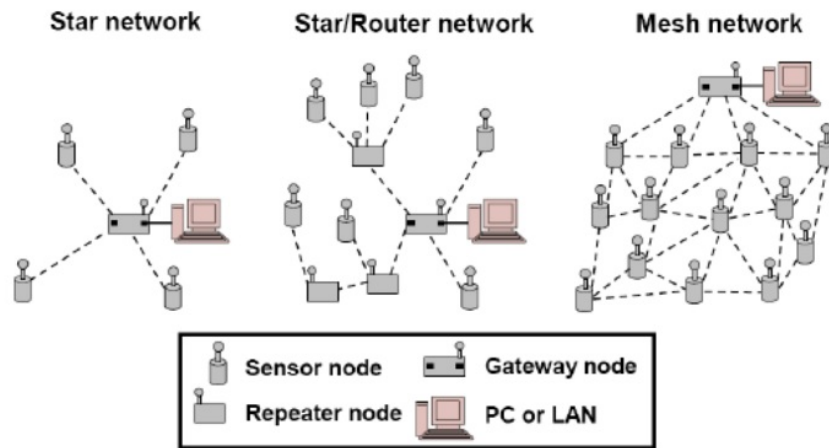


Figura 2.2: Topologias Mais Utilizadas em RSSF. Adaptado de: [GTA-UFRJ, 2014]

2.2 Aplicações em Campi Inteligentes

Os conceitos relacionados a cidades e campi inteligentes estão cada vez mais presentes com a rápida popularização da IoT e das RSSF. Nos últimos anos, diversas aplicações vêm sendo propostas para transporte inteligente, iluminação e monitoramento ambiental [Shkurti et al., 2017], [Visconti et al., 2017]. A maior parte dessas aplicações requerem o uso de comunicação sem fio objetivando a transmissão de informações a partir de sensores espalhados pelo ambiente. Diversos campi universitários possuem um alto tráfego de pessoas e veículos, o que demanda as mesmas características físicas de uma pequena cidade. O uso de redes sem fio permite obter um baixo custo de implantação e alta flexibilidade em comparação ao uso de redes cabeadas. Além disso, muitas dessas aplicações requerem redes de sensores que abrangem grandes áreas.

Como exemplo de aplicação em campi inteligentes, a implementação de um sistema inteligente para medição e monitoramento (*Smart Meters*) pode ser observada como uma possível solução para a alta demanda de energia em diversas repartições públicas e privadas de grande porte. Os medidores inteligentes contam com a capacidade de realizar leituras de dados mais abrangentes, obtendo diversas informações relevantes, tais como:

medição de parâmetros, cálculos, calibração automática dos dispositivos, entre outros. Estes sistemas de medição possuem alto nível de flexibilidade pelo fato de sua programação possuir características de fácil escalabilidade e modularidade [Barai et al., 2015].

A gerência do contrato de concessão energética torna-se amplamente simplificado com a utilização de *smart meters*, pelo fato dos consumidores conseguirem mensurar o seu próprio consumo em um determinado instante, enquanto que as concessionárias de energia podem promover um desligamento automático em tempo real para aquele consumidor inadimplente [Zheng et al., 2013].

Vale destacar alguns aspectos relevantes aos *smart meters* [Pfeifer et al., 2018]:

- Melhor gestão de tempo: as leituras em um medidor inteligente são enviadas eletronicamente, deste modo, evitam-se custos e atrasos relacionados à logística quando se necessita de uma leitura naquele instante;
- Aumento do nível de conscientização do consumidor final: através da utilização destes dispositivos, consegue-se realizar um acompanhamento mais detalhado sobre determinado horário, sendo possível avaliar o consumo em horários de pico e programar-se melhor para evitar os excessos em determinado momento;
- Auxílio por efeito de definição quantitativa: métricas podem ser utilizadas, as quais irão auxiliar em uma melhor projeção da despesa mensal de energia. Por exemplo, através da definição de metas diárias de consumo e picos máximos em horários de alta demanda;

Vale destacar ainda, que a implantação de redes de monitoramento inteligentes exige um certo grau de convergência entre diferentes variáveis, baseadas tanto em relação as tecnologias tradicionais do setor de energia, como em tecnologias de informação e comunicação, sensoriamento e monitoramento de equipamentos. Além do desenvolvimento de práticas operacionais responsáveis pela integração entre essas tecnologias, de modo a permitir que possam operar de modo conjunto. Esta, ainda trata-se de uma tarefa complexa, ainda em estágio experimental em diversos países do mundo [Bandeira, 2012].

2.3 Padrão IEEE 802.15.4

Com a evolução das tecnologias associadas às RSSF, cada vez mais buscavam-se alternativas que unissem simplicidade, baixo custo e um diminuto consumo de energia. O Comitê de Novos Padrões (*New Standards Committee*, NES-COM) do IEEE, em dezembro de 2000, criou um grupo de trabalho denominado TG4 (*Task Group 4*), associado a um grupo maior, o 802.15, que já desenvolvia pesquisa e padronizações para as redes pessoais sem fio (*WPAN - Wireless Personal Area Networks*). Ao TG4 foi atribuída a missão de desenvolver e especificar um novo padrão de WPAN, que trabalhasse com baixa taxa de transferência (LR-WPAN). Baseados nestes objetivos atribuídos ao TG4, deu-se início ao desenvolvimento do padrão IEEE 802.15.4, que permitiria associar: conectividade sem fio com baixa complexidade, diminuto consumo de energia, baixo custo e uma taxa reduzida para transferência de dados entre dispositivos distribuídos [Gutierrez et al., 2001].

Após o seu desenvolvimento e padronização, a norma IEEE 802.15.4 passa a definir as especificações para comunicação de dados entre dois níveis: a camada física (PHY) e a camada de acesso ao meio (MAC), para sistemas de comunicação sem fio que trabalham com reduzida taxa de transmissão de dados. A Figura 2.3 traz uma hierarquia baseada no modelo TCP/IP de 5 camadas ilustrando as camadas em que o padrão IEEE 802.15.4 atua. Vale destacar ainda que as camadas superiores não são definidas no IEEE 802.15.4, mas podem ser definidas por outros padrões complementares, como ZigBee e WirelessHART.

a) Camada Física (PHY)

Esta camada possui como principal objetivo a transmissão dos bits entre os níveis de origem e destino, transmitindo e recebendo dados conhecidos como (PPDU - *Physical Protocol Data Unit*). Uma vez transmitidos, este pacote de dados segue para a camada MAC. Este processo de transmissão e recepção de dados é possibilitado pelas técnicas de modulação e demodulação inerentes à camada PHY.

O Padrão IEEE 802.15.4 define 3 bandas de frequência não licenciadas pertencentes ao "range" destinado ao ISM (*Industrial Scientific Medical*). São elas: 868 MHz

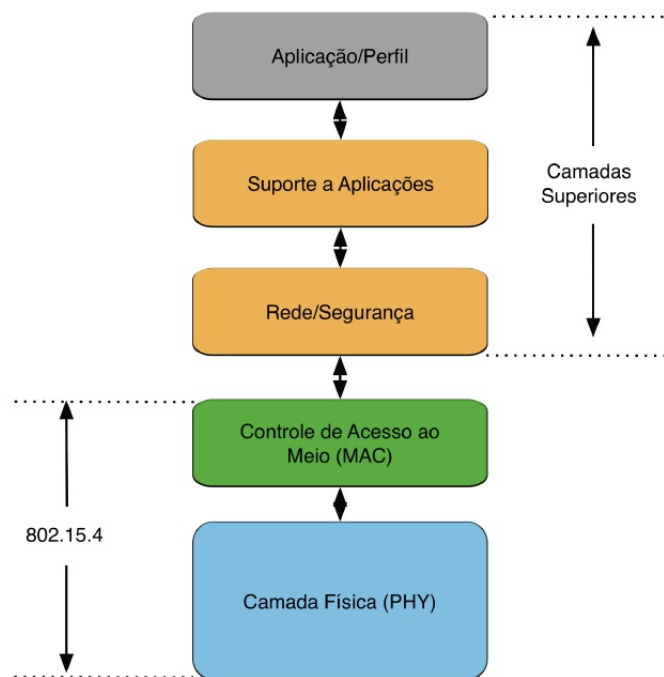


Figura 2.3: Camadas do Padrão IEEE 802.15.4. Adaptado de: [Hanes et al., 2017]

(uso específico na Europa), 915 MHz (EUA e Américas) e 2,4 GHz (no mundo todo) [Soujeri et al., 2013].

Ao todo, estas faixas reservadas do espectro de frequência ficam distribuídas em 27 canais, alocados da seguinte forma:

- 1 canal para a banda de frequência utilizada na Europa;
- 10 canais para a banda de frequência utilizada pelos EUA e Américas;
- 16 canais para a faixa de frequência utilizada no resto do mundo;

Conforme pode-se inferir acerca destas informações, a faixa de 2,4 GHz é a mais abrangente, pelo fato de propiciar sua disponibilidade para uso sem licença, baixa latência e uma gama maior de canais possíveis a serem utilizados. A banda de 2,4 GHz possibilita 16 canais em intervalos regulares de 5 MHz para diferenciar cada um deles, tendo assim um escopo entre: 2,4 - 2,4835 GHz. Já a banda utilizada pelos EUA, faz uma alocação de 902 a 928 MHz, sendo os canais divididos em intervalos de 2 MHz.

A Tabela 2.3 ilustra o resumo com os aspectos mais importantes relacionados à camada física.

Tabela 2.1: Aspectos da Camada Física. Adaptado de: [Ferret, 2013]

Banda de Frequência (MHz)	Canais	Modulação	Taxa de bits (Kb/s)	Largura
868 - 868.6	16	BPSK	20	5 MHz
902 - 928	1	BPSK	40	600 KHz
2.400 - 2.483,5	10	O-QPSK	250	10 MHz

Vale destacar que esta descrição foi definida na primeira implementação do padrão IEEE 802.15.4 e que vigorou até o ano de 2015, quando foram incluídas várias outras definições para a camada física advindas da emenda IEEE 802.15.4g. Mais detalhes sobre as especificações da emenda IEEE 802.15.4g são fornecidos na Seção 2.3.1.

b) Camada MAC

Esta camada consiste em uma subdivisão do nível de enlace segundo o modelo TCP/IP. Já no contexto do padrão IEEE 802.15.4, exerce a gestão de toda a operação que envolve o nível físico no eixo de transmissão e recepção de dados. A este nível também é atribuída a função de produzir e sincronizar os *beacons*, além de ordenar os dados e prepará-los para as camadas superiores do protocolo.

Dependendo da configuração da rede, são possíveis dois mecanismos diferentes para acesso ao canal de rádio. Em uma rede baseada em *beacon* habilitado, o mecanismo conhecido por *slotted CSMA-CA* (*Carrier Sense Multiple Access With Collision Avoidance*) é utilizado pela subcamada MAC para tratar as transmissões dentro do *superframe*; de modo oposto, em redes sem *beacon*, o mecanismo utilizado é o *unslotted CSMA-CA*. Nos dois casos, a implementação do algoritmo é feita com base em unidades de tempo denominadas "períodos de *backoff*" ($BP = \text{Backoff Period}$). Este BP refere-se ao tempo necessário para ser transmitidos 20 símbolos, ou $4 \times 20 = 80$ bits, quando a rede estiver operando em nível de camada física em 2.4 GHz [de Souza et al., 2013].

Vale destacar que a partir da versão 2015, com a incorporação da emenda IEEE 802.15.4e, novos protocolos de acesso ao meio foram adicionados, como o TSCH (*Time Slotted Channel Hopping*), que é baseado em acesso múltiplo por divisão de tempo e salto

de frequência.

c) Padrões de Topologia

Dois arranjos de interligação física são mais utilizados nos dispositivos que fazem uso do padrão IEEE 802.15.4. A topologia ponto-a-ponto e a estrela. Neste primeiro modo, existe uma ligação direta entre as entidades pertencentes à rede, de modo que não existe hierarquia, podendo qualquer parte exercer o papel de cliente ou servidor. Já no outro arranjo topológico, existe a presença de um nó sorvedouro, o qual exerce a função de gerenciamento de todo tráfego de informações da subrede. Além destes padrões mais utilizados, vale destacar também a possibilidade de configuração dos arranjos em árvore ou malha (*mesh*). Na topologia em árvore, temos uma estrutura hierárquica baseada na distribuição dos nós. Enquanto que na configuração em malha, existe uma quase completa integração entre as entidades pertencentes à rede.

De acordo com [Hanes et al., 2017] o padrão IEEE 802.15.4 descreve, além das possibilidades de arranjos topológicos, a presença de entidades classificadas de acordo com as funções exercidas. Sendo elas: FFD (*Full Function Device*) e RFD (*Reduced Function Device*). Como a própria nomenclatura descreve, o modo FFD dispõe de mais recursos para a gestão da estrutura hierárquica, possibilitando que o nó possa atuar como roteador ou coordenador. Em contrapartida, os dispositivos atuantes como RFD possuem um conjunto de serviços MAC reduzido, o que restringe a atuação em determinadas funções mais restritas.

A Figura 2.4 ilustra os principais padrões topológicos utilizados, além das entidades atuantes com suas respectivas representações funcionais.

2.3.1 IEEE 802.15.4g

O Padrão IEEE 802.15.4g consiste em uma emenda da norma IEEE 802.15.4, que padroniza alterações em sua camada física. Tais modificações são necessárias para implementação das SUNs (*Smart Utility Networks*). Este adendo da norma IEEE 802.15.4

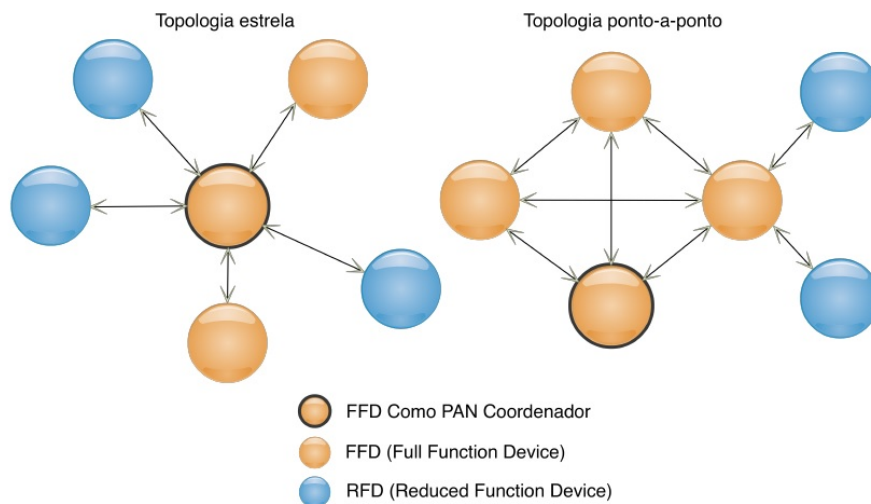


Figura 2.4: Principais Arranjos Topológicos. Adaptado de: [Saraiva., 2019]

possibilitou a adição de características significativas, tais como, taxas de transmissão mais elevadas, maior alcance, além de descrever um conjunto de possibilidades de modulação em diferentes bandas de frequência.

Com a evolução do padrão IEEE 802.15.4, foi possível ampliar a gama de recursos oferecidos para um *"range"* maior, ou seja, uma maior área de abrangência geográfica, mantendo a economia de energia em ambientes tanto internos, como ao ar livre. Optou-se ainda por um adendo da norma, para aproveitar a maior parte de sua estrutura, deste modo, realizando alterações apenas em nível físico. Com a evolução do padrão anterior, tivemos uma grande expansão no leque de possibilidades de monitoramento, que antes, apesar de já existentes, foram aprimoradas, tais como: agricultura inteligente, campi inteligente, cidades, controle de tráfego, entre outros conforme mencionados anteriormente [Munoz et al., 2018]. A Figura 2.5 ilustra a modificação na estrutura das camadas que compõem padrão IEEE 802.15.4g.

Das modificações realizadas em nível físico, podemos citar três novas variações de modulação definidas:

- MR-FSK (*Multi-rate and multi-regional frequency shift Keying*). Este modo de operação apresenta como vantagem a otimização energética por meio da utilização de algoritmos de menor complexidade. Já como desvantagem, este fator está total-

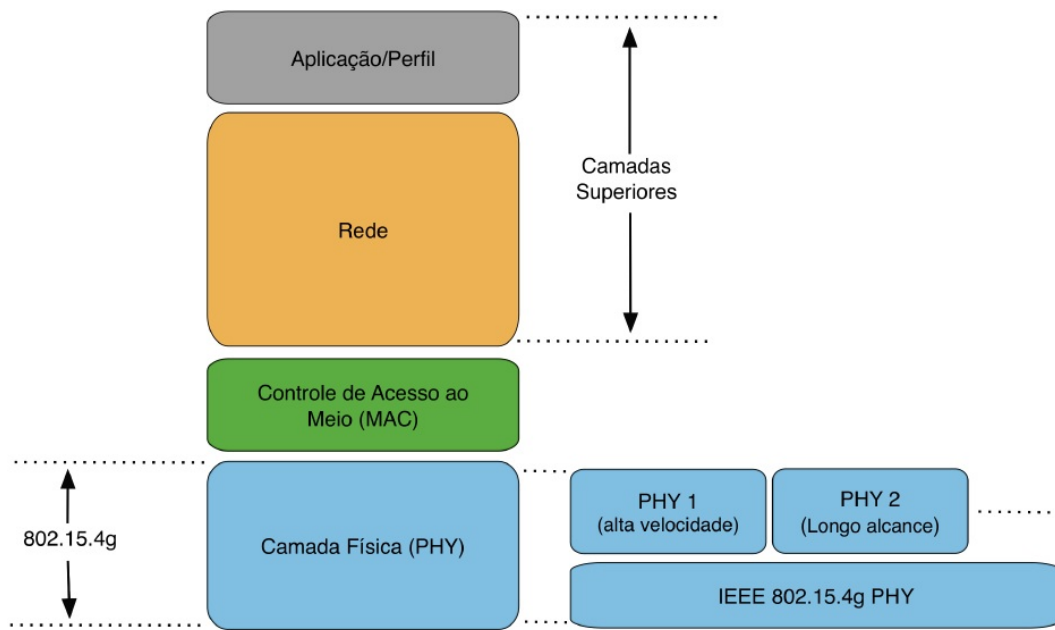


Figura 2.5: Estrutura de Camadas do Padrão IEEE 802.15.4g. Adaptado de: [Saraiva., 2019]

mente ligado à baixa largura de banda provocada pela alta fragilidade na incidência de interferências eletromagnéticas [Sum et al., 2017]. A Figura 2.6 ilustra uma representação da Modulação FSK;

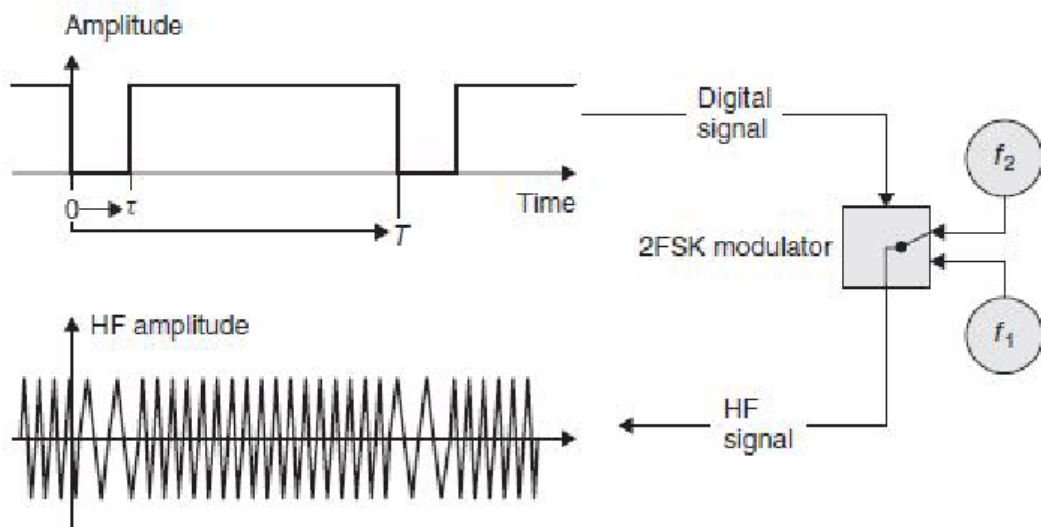


Figura 2.6: Modulação FSK. [Bilic and Ozdemir, 2017]

- MR-OFDM (*Multi-rate and multi-regional Orthogonal frequency-division multiplexing*). Neste padrão de modulação ocorre a codificação em múltiplas portadoras. Este sistema advém como uma evolução do padrão FDM, que define a segmentação

baseada em divisão de frequência. Ou seja, o OFDM baseia-se no mesmo princípio, porém com o espaçamento ortogonal entre as subportadoras [Sum et al., 2017], como ilustrado na Figura 2.7;

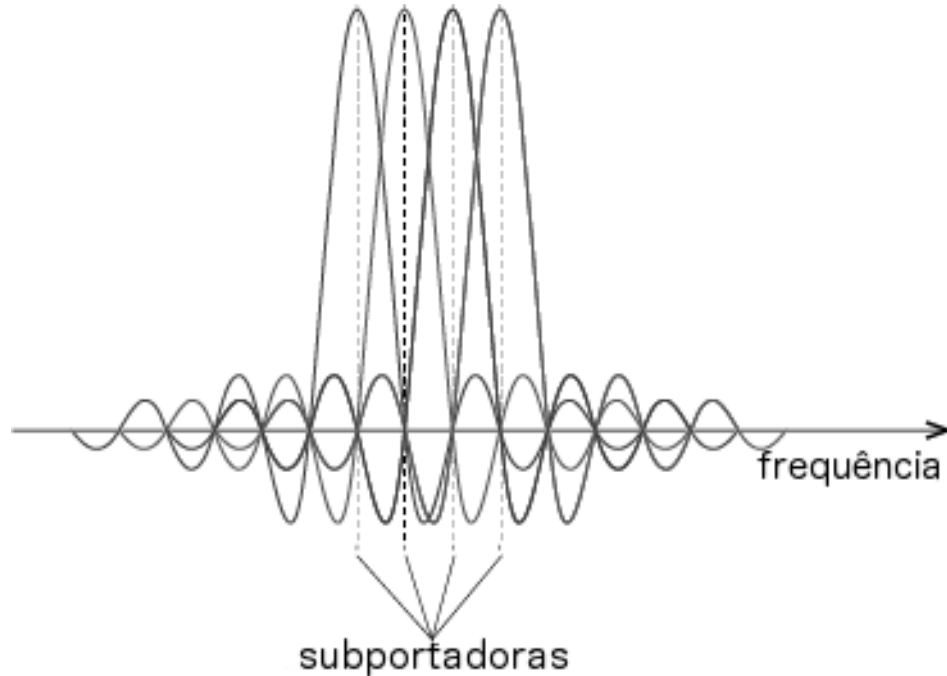


Figura 2.7: Modulação OFDM. [Abdul-Jabbar, 2015]

- MR-OQPSK (*Multi-rate and multi-regional Offset Quadrature Phase-shift Keying*). Nesta técnica, a taxa de bits varia de 6 kbps a 500 kbps. Este modo também faz uso de DSSS (*Direct Sequence Spread Spectrum*), técnica que fornece uma densidade espectral de potência baixa, exigindo uma amplitude de faixa grande para maiores taxas de transmissão [Munoz et al., 2018].

Dois aspectos são importantes de serem destacados: o primeiro deve-se ao fato de que para cada modulação existem diferentes configurações que podem ser utilizadas e isso permite obter uma relação de compromisso entre alcance, confiabilidade, taxa de bits e consumo de energia. O segundo aspecto a ser destacado é o da segurança. Segundo [Munoz et al., 2018], o recurso de criptografia existente no padrão IEEE 802.15.4g advém da especificação AES (*Advanced Encryption Standard*), onde, a referida inclusão das diretrizes de segurança são implementadas na camada MAC com uma chave de 128 bits. De acordo com [Hanes et al., 2017] a autenticação no quadro completo é estruturado

através do campo MIC, que consiste em um padrão de valor único calculado com base nas informações contidas no quadro.

Este quadro conta ainda com um cabeçalho de segurança, além de itens que disponibilizam elementos de informação a cerca da mensagem tratada. Vale destacar a ligação com o KMP (Protocolo de Gerenciamento de Chaves IEEE 802.15.9), que disponibiliza um canal confiável para o compartilhamento de chaves, necessárias para a comunicação segura em contraponto a abordagem em texto claro [Saraiva., 2019].

2.4 Wireless Smart Utilities / Ubiquitous Networks

Podemos compreender a Wi-SUN como um padrão internacional que permite a interoperabilidade com o protocolo IEEE 802.15.4g em nível físico (PHY) e em nível da camada de ligação de dados (LLC) com os padrões IEEE 802.15.4/4e. Uma das principais vantagens de utilização deste padrão advém da sua alta eficiência energética, possibilitando a utilização de baterias com grande autonomia e demandando substituições em intervalos bastante amplos.

Wi-SUN é uma tecnologia que vem sendo amplamente utilizada em sistemas de monitoramento de água, eletricidade e consumo de gás. Por exemplo, no subcampo do monitoramento de campi inteligentes, observa-se grande utilização em sistemas de *smart metering*, por meio da implementação conjunta com novos medidores que permitem a monitoração de grandezas em tempo real.

Os sistemas que utilizam Wi-SUN podem ser implementados em áreas de curta, média ou grande abrangência espacial. Em áreas mais amplas, normalmente configuram-se dois tipos de dispositivos: os de medição e os coordenadores. No *uplink* os dados são enviados para o coordenador pelos dispositivos de medição. Já no modo *downlink*, o coordenador, como a própria nomenclatura sugere, gerencia a transmissão enviando sinais de controle para os dispositivos de medição [Neves, 2019].

Além dos dispositivos de medição e coordenadores, vale destacar a existência dos

roteadores intermediários, que fazem a comunicação direta para fora da rede, permitindo que outras entidades possam sem comunicar com servidores externos à rede. Na utilização de Wi-SUN com topologia em malha, por meio de implementação da técnica de *multi-hop* existe grande flexibilidade na rede, principalmente pelo fato dos coordenadores e dispositivos de medição utilizarem a mesma tecnologia de rádio, o que possibilita taxas de transmissão e recepção bastante próximas tanto em dispositivos individuais como nas entidades transceptoras [Neves, 2019].

A Figura 2.8 ilustra um sistema Wi-Sun com configuração multi-hop, representando o fluxo bidirecional de baixa potência entre dispositivos e o agente coordenador.

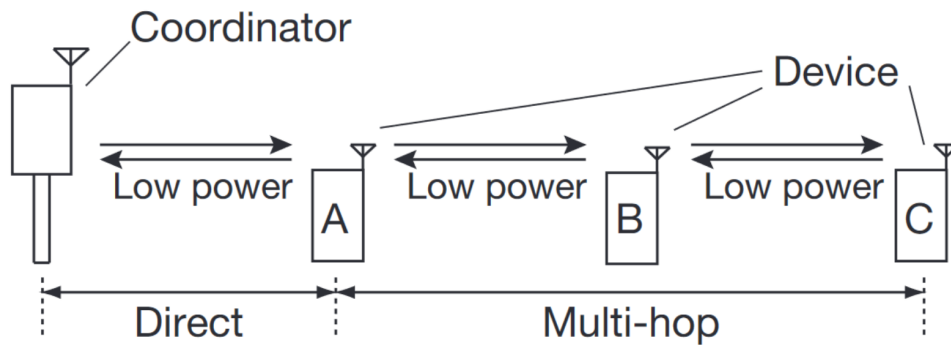


Figura 2.8: Exemplo Wi-SUN com *multi-hop*. Adaptado de: [Mochizuki et al., 2016]

Com relação às faixas de operação do padrão Wi-SUN, estas são atuantes no espectro sub-GHz. A faixa espectral é diferente em determinadas partes do mundo, sendo os intervalos: 863-870 MHz e 870-876 MHz na Europa, 920-928 MHz nos EUA, Canadá e Japão e 470-510 MHz, 779-787 MHz e 920,5-924,5 MHz na China em uma taxa de transmissão média de 300 kbps. O raio de abrangência das comunicações Wi-SUN é bastante amplo, e pode ser aumentado com a adição de repetidores de sinal, respeitando-se as topologias definidas pelo padrão. Este intervalo médio normalmente fica entre 500 m a 5 km [Al-Turjman and Abujubbeh, 2019].

A Wi-SUN Alliance, entidade sem fins lucrativos iniciada oficialmente em 2012, é a mantenedora do projeto Wi-SUN, que conta com a colaboração de mais de 100 empresas parceiras. Esta entidade é a responsável por promover a adoção de padrões industriais

abertos, estabelecer equipamentos interoperáveis, além de normatizar as certificações de interoperabilidade e conformidade.

No que envolve à segurança, Wi-SUN adota a política de recomendações da NIST (*National Institute of Standards and Technology*), tomando por base os protocolos EAP-TLS e 802.1x para autenticação das entidades envolvidas e AES-128 para a encriptação dos dados trafegados. Vale destacar a alta interoperabilidade entre dispositivos de fabricantes distintos. Este fenômeno ocorre através da criação de uma subrede chamada HAM (*Home Area Network*) que segmenta as informações entre grupos comuns fazendo com que os dispositivos intermediários e de borda da subrede possam trabalhar conjuntamente.

A Figura 2.9 ilustra o escopo de uma subrede do tipo HAM. Observa-se nela a presença de um microcontrolador MB-RL7023-09/HSE responsável pelo gerenciamento dos sinais advindos da HAM e repassados ao controlador, que neste contexto possui função de transceptor. Vale destacar também que o papel exercido pelas redes do tipo HAM, não são descritos de modo nativo no padrão Wi-Sun, e sim como configuração opcional.

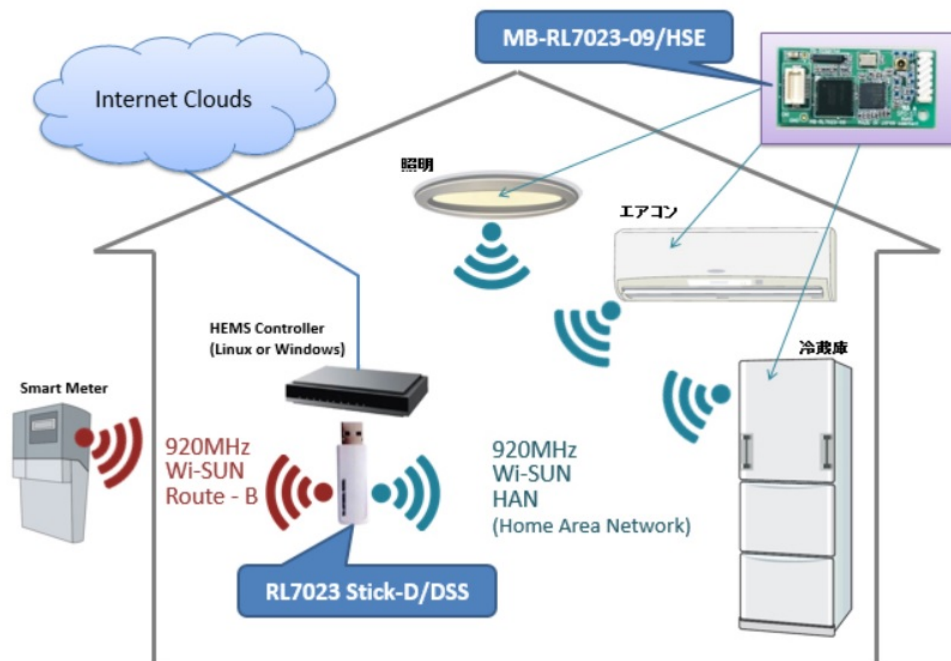


Figura 2.9: Infraestrutura da Sub-rede HAM. Adaptado de: [Tessera, 2018]

2.5 InfluxDB

Este é o banco utilizado para o registro dos dados obtidos a partir dos sensores. O InfluxDB consiste em um banco de dados de código aberto projetado para manipular séries temporais de um modo otimizado no que envolve desempenho e modelagem, desenvolvido pela empresa InfluxData. De sintaxe bastante semelhante ao SQL (*Structured Query Language*), esta plataforma foi criada com o intuito de armazenar vários tipos de dados que apresentam variações em função do tempo. De modo genérico, uma série é composta por 4 tipos de variações: tendência, ciclo, sazonalidade e variações irregulares. [Naqvi et al., 2017].

De acordo com [Cruz G. and Lean, 2017], temos duas estruturas essenciais ao se trabalhar com registros na plataforma InfluxDB:

- *Fields*: consistem em uma representação básica de estrutura de colunas as quais são nomeadas e armazenados dados relevantes para a representação da série baseados no esquema chave-valor. A *Field Key* é uma *string* e representa, por sua vez, o nome da coluna, tendo o *Field Value* como o valor do registro armazenado. A *Field* é tida como um índice de registro básico, ou seja, não se pode ter um registro sem ao menos um *Field*. Vale ainda destacar que este índice não é indexado, ou seja, a sua busca em um registro de banco implica na leitura de todos os valores.
- *Tags*: consistem em outra espécie de colunas que, assim como as *Fields* são armazenadas no formato chave-valor. Como o próprio nome diz, este tipo armazena uma espécie de metadado, o qual será alvo das consultas ao banco. Outro ponto de diferenciação das *Tags* é o fato das *Tags* serem indexadas.
- *Measurement*: trata-se de uma estrutura responsável pela alocação de *Tags* e *Fields*. Ou ainda, uma espécie de tabela de registros, fazendo-se uma analogia ao modelo de entidade-relacionamento.

Um ponto relevante a ser destacado é que não necessariamente precisa-se armazenar dados que variam conforme o tempo em um banco próprio de registro de sé-

ries temporais. Tais registros podem ser armazenados em um banco de dados entidade-relacionamento padrão. Porém, um fator que foi determinante para a escolha do InfluxDB é a questão do seu tempo de execução. As Figuras 2.10 e 2.11 ilustram o tempo médio de execução por número de consulta e o tempo médio de execução para todas as consultas em milissegundos, respectivamente. De acordo com os gráficos apresentados, pode-se observar o tempo de execução inferior nos dois casos em favor do InfluxDB, em comparação com o SQL Server.

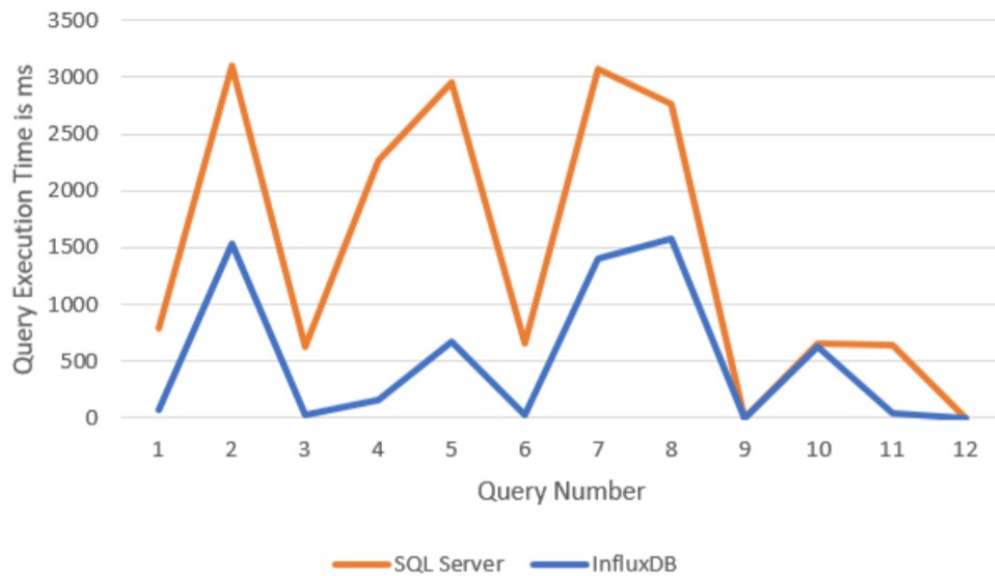


Figura 2.10: Tempo Médio de Execução por Número de Consulta. Adaptado de: [Naqvi et al., 2017]

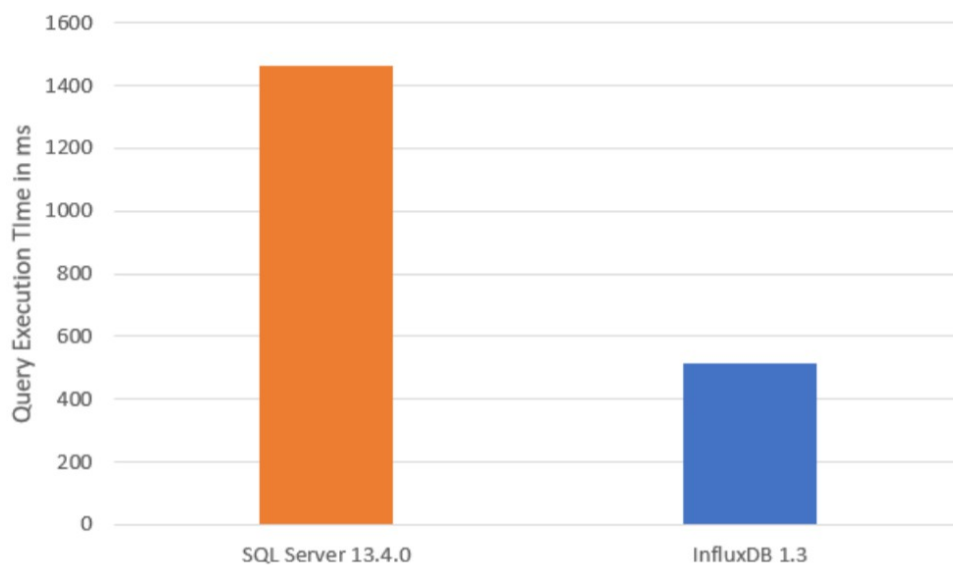


Figura 2.11: Tempo Médio de Execução para Todas as Consultas. Adaptado de: [Naqvi et al., 2017]

2.6 Protocolo MQTT

O MQTT (*Message Queuing Telemetry Transport*) consiste em um protocolo simplificado para troca de mensagens, desenvolvido pela IBM, com foco na aplicação em Internet das Coisas [Hunkeler et al., 2008]. Ele vem sendo largamente utilizado na integração de sensores presentes em microcontroladores e pequenos dispositivos móveis embarcados.

Este protocolo baseia-se na comunicação do modelo *publish/subscriber*, que consiste em um sistema no qual clientes interessados passem a assinar tópicos que sejam de seu interesse por intermédio de um servidor centralizado denominado *Broker* MQTT. Como o protocolo foi implementado principalmente para uso em IoT ele conta com um escopo principal de foco em economia de recursos dos dispositivos (exigindo um baixo nível de processamento), mínimo uso da largura de banda da rede e um satisfatório nível de confiabilidade e garantia de entrega. Estas características tornam-se relevantes em um cenário em que baixo poder de processamento e potência de bateria são de extrema importância. Além de aplicações voltadas a IoT, o protocolo MQTT também é utilizado nas chamadas "comunicações emergentes" (*machine-to-machine* - M2M) [Quincozes et al., 2019].

O MQTT surgiu da necessidade do estabelecimento de uma conexão rápida, simples e com grau de confiabilidade promissor. No final dos anos 90 a IBM (*International Business Machines Corporation*) teve a necessidade de desenvolver um protocolo simples e leve para comunicação entre pequenos sensores, microcontroladores e dispositivos com poder computacional limitado, agentes estes comumente encontrados em sistemas de IoT.

A Figura 2.12 ilustra a comunicação por meio do paradigma *publish/subscriber* utilizado pelo protocolo MQTT. É considerado um cenário em que 3 nós *publishers* alimentam um determinado *Broker*, que possui dois tópicos e dois assinantes. De acordo com a representação, o dispositivo assinante J deseja obter as mensagens providas pelo Tópico A, entidade responsável pela propagação das mensagens providas pelo dispositivo *publisher* 1. O *subscriber* K, recebe os dados publicados pelo dispositivo *publisher* 1 e por intermédio da assinatura do tópico A obtém acesso simultâneo aos dados disponibilizados pelos *publishers* 2 e 3, por meio da assinatura do tópico B. Desta forma, observa-se

que sempre que um nó *publisher* enviar uma mensagem para o *Broker*, tal mensagem será encaminhada em *Broadcast* para todos os dispositivos assinantes, não sendo necessário que estes estejam ativos no momento em que a mensagem é gerada para receber a comunicação, caracterizando assim uma comunicação assíncrona [Quincozes et al., 2019].

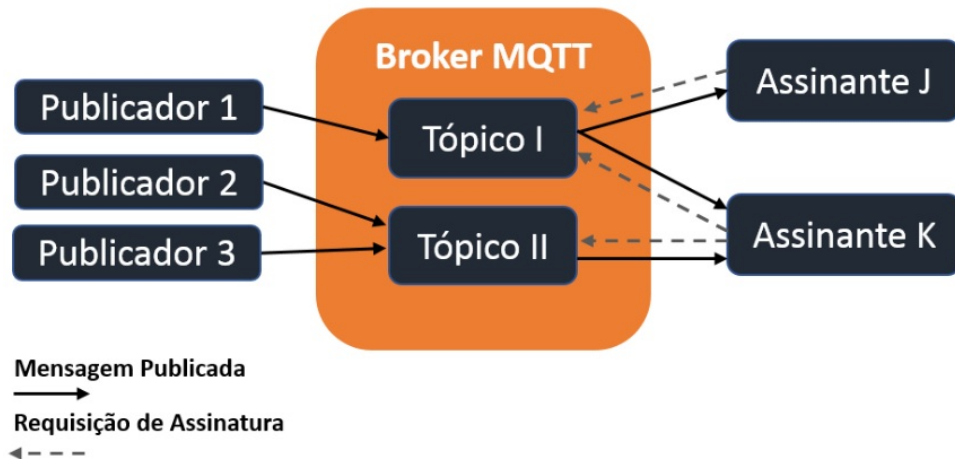


Figura 2.12: Representação do Broker MQTT. Adaptado de: [Quincozes et al., 2019]

Além de todas as características elencadas, o MQTT conta com alto grau de leveza e simplicidade em seu funcionamento. Estes aspectos são de grande relevância para utilização em dispositivos de IoT. HTTP (*Hypertext Transfer Protocol*) por exemplo, trata-se de um protocolo síncrono, normalmente um determinado cliente aguarda a resposta do servidor a um pedido de requisição de conexão. A grande maioria dos navegadores obedecem este requisito, embora traga como fator dificultante a baixa escalabilidade.

Especificamente quando pensamos no universo da IoT, a comunicação síncrona torna-se um pouco mais complexa devido ao grande número de nós conectados à rede, na maior parte das vezes, enfrentando problemas relacionados a falta de confiabilidade e latência. Deste modo, observa-se que protocolos de comunicação assíncronos podem vir a desempenhar melhor este papel, por permitir que sensores possam encaminhar leituras possibilitando que a rede fique responsável pelos aspectos relacionados a entrega aos dispositivos de destino.

Conforme citado anteriormente, HTTP é um protocolo com alto grau de eficiência. No entanto, quando pensamos no universo IoT, este possivelmente não seria a melhor opção, por tratar-se de um protocolo relativamente pesado para dispositivos com capa-

cidade de processamento limitada, o qual conta com muitos cabeçalhos e regras, sendo assim uma opção não apropriada para redes com algum tipo de restrição. Ainda sobre o HTTP, este atua com base no esquema P2P (*Peer-to-Peer*), em que o agente cliente solicita um determinado serviço e o servidor responde a esta solicitação, quando possível. Neste contexto, certamente temos um cenário mais complexo e custoso ainda, quando se necessita o envio de uma transmissão em *broadcast* para a rede, contexto este muito comum em aplicações IoT.

De acordo com a IBM, um dos protocolos de mensagens mais popular para sistemas de middleware corporativos denomina-se AMQP (*Advanced Message Queuing Protocol*), entretanto, em um ambiente de alto desempenho, com grande capacidade computacional, a latência da rede não configura-se como uma preocupação. Além do AMQP, outros protocolos populares foram cogitados para a função de agente integrador da comunicação na arquitetura de rede proposta neste estudo. Um outro exemplo seria o XMPP (*Extensible Messaging and Presence Protocol*) que consiste em um protocolo de troca de mensagens de modo instantâneo atuante no modelo ponto a ponto. Por contar com diversos atributos agregadores de funcionalidades, tais como, presença e adesão de anexos de mídia, em comparação com MQTT, o mesmo consome mais recursos a nível de dispositivo e rede.

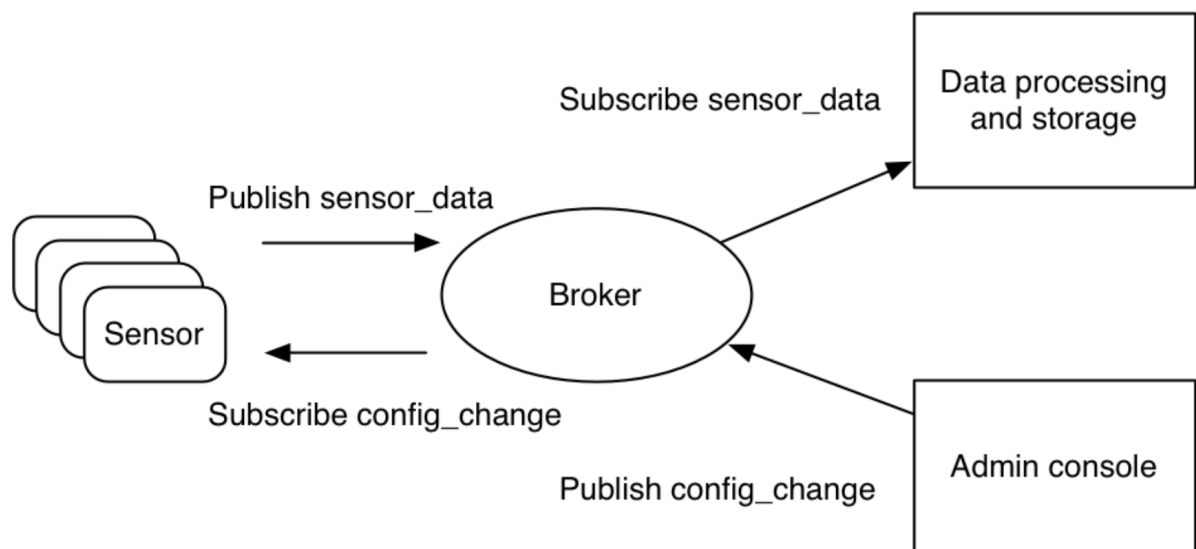


Figura 2.13: Modelo Representativo de Publicação-Assinatura do MQTT para IoT. Adaptado de: IBM Developer US.

De acordo com a Figura 2.13 observa-se a troca de comunicação a partir dos

sensores para o broker ilustrando o tipo de mensagem trocada. Verifica-se um cabeçalho bastante simples o qual especifica, um determinado tópico baseado em texto e a carga útil com as informações conduzidas. Ainda sobre a carga útil, é possível utilizar vários formatos de dados, tais como: JSON, XML ou Base64, por exemplo.

2.6.1 Principais Parâmetros e Níveis de QoS

O MQTT conta com diversos parâmetros para configuração das aplicações, além de contar com três níveis nativos para tratamento de QoS.

CONNECT: tipo de mensagem responsável pelo estabelecimento da conexão do cliente ao *broker*. Este padrão concebe parâmetros para especificar ações inerentes ao tipo da mensagem, por exemplo:

- i) `cleanSession`: flag que sinaliza se a conexão é do tipo persistente ou não. Neste contexto, a persistência determina o armazenamento de todas as assinaturas e o registro de informações possivelmente perdidas.

CONNACK: tipo de mensagem relacionada a confirmação da requisição repassada.

- i) `sessionPresent`: flag responsável por identificar se a conexão estabelecida já possui uma comunicação do tipo persistente.

SUBSCRIBE: refere-se ao processo estabelecimento da conexão aos tópicos cadastrados no *broker*. Permite também a especificação dos níveis de QoS na comunicação.

- i) `qos0`: não-confiável. Neste padrão a mensagem encaminhada é entregue, no máximo, uma vez. Se por ventura, o subscriber esteja offline no momento, a mensagem será perdida.

- ii) `qos1`: Neste padrão existe uma garantia mínima de entrega da mensagem, a qual deve ser entregue, ao menos uma vez.

iii) qos2: Neste terceiro padrão, o nível mais alto de QoS presente no modelo, a mensagem deve ser entregue exatamente uma vez.

2.7 Protocolo CoAP

O CoAP (*Constrained Application Protocol*) consiste em um protocolo amplamente utilizado em IoT, desenvolvido pelo grupo CoRE (*Constrained RESTful Environments*) da IETF (*Internet Task Force Engineering*). Possui funcionamento bastante semelhante ao modelo Cliente/Servidor utilizado pelo Protocolo HTTP e é executado em cima do Protocolo UDP, com uma troca de mensagens de tamanho reduzido, ideal para dispositivos que possuem limitações relacionadas a memória e consumo de energia [Shelby et al., 2014].

A efetiva troca de mensagens concebida pelo CoAP se dá por meio do UDP entre os nós finais utilizando uma estrutura de cabeçalho com comprimento fixo de 4 bytes. Cada mensagem é identificada por um ID de 16 bits utilizado como forma de detecção para duplicações e para confiabilidade. Vale destacar as principais mensagens trocadas entre os dispositivos: CON (*Confirmable*), indica a confiabilidade na transmissão de uma determinada mensagem até que o destino envie uma mensagem de *Acknowledgement* (ACK) com o detalhamento do mesmo ID referente aquela mensagem correspondente. Caso o destinatário não retorne a confirmação, uma mensagem RST (*Reset*) é retornada. Além da NON (*Non-Confirmable*) bastante utilizada em situações em que a aplicação não requer ou não necessita de transmissão confiável [Shelby et al., 2014].

No que diz respeito à segurança, o CoAP faz uso de DTLS (*Datagram Transport Layer Security*). Para este modo, existem quatro configurações possíveis: i) NoSec; ii) PreSharedKey; iii) RawPublicKey e iv) Certificate. No primeiro modo (NoSec) não existe DTLS ativo, já os três modos seguintes apresentam distintas configurações, sendo respectivamente: com chaves compartilhadas para proceder a indicação sobre quais os nós podem vir a ser utilizados, com pares de chaves assimétricos ou, no último modo, sem certificação [Shelby et al., 2014].

2.7.1 Modelo de Solicitação/Resposta

De acordo com a RFC 7252, informações relativas à solicitação e resposta, como por exemplo, a URI ou o tipo de mídia de carga, são transportados como possíveis opções CoAP. Métodos comumente utilizados no protocolo HTTP são também atribuídos às aplicações que utilizam CoAP, a exemplo dos métodos: GET, PUT, POST e DEL. Apesar do CoAP configurar-se como um tipo de protocolo M2M, uma única solicitação pode perfeitamente obter múltiplas respostas, como em uma transmissão de Multicast. Tendo uma determinada mensagem sendo retornada como Confirmável (CON) ou não-confirmável (NON), ou ainda ACK, como resposta a um pedido de confirmação. Essa resposta é chamada de *piggybacked* [Shelby et al., 2014].

Na Figura 2.14 são representados dois exemplos de um pedido GET para retorno da temperatura coletada por um sensor, em que, no primeiro exemplo, temos uma resposta bem sucedida, com o retorno de 22 °C e no segundo caso, com um retorno "vazio" ilustrando um erro 4.04 (não encontrado).

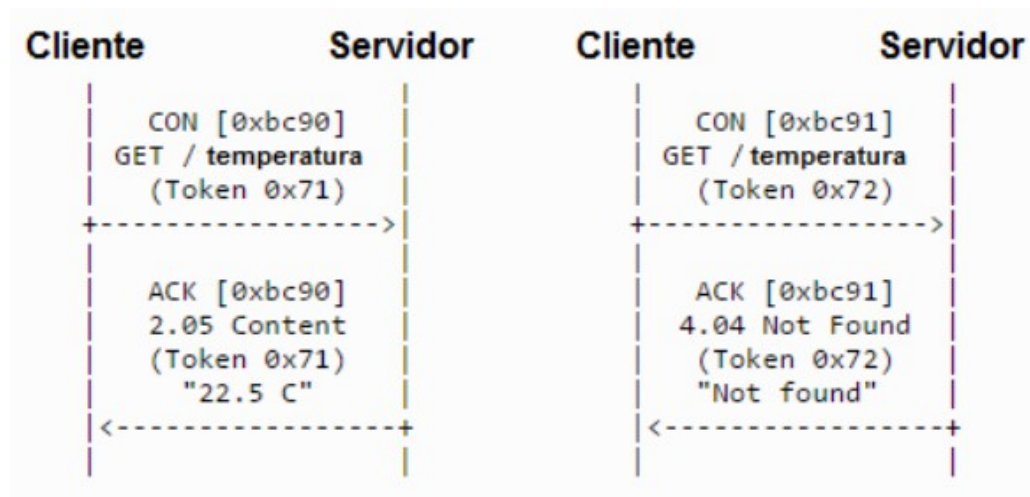


Figura 2.14: Representação dos Métodos GET e ACK. Adaptado de: [Shelby et al., 2014]

Existe também o cenário em que um determinado servidor não consegue fornecer uma resposta imediata para uma mensagem confirmável. Nesta situação, ele responde com uma mensagem vazia de confirmação, para que o cliente tome a ação de parar de retransmitir o pedido. Passado um período de tempo, quando a resposta está pronta, o servidor procede com o envio em uma nova mensagem confirmável. Este fato está

representado na Figura 2.15 denominado como "resposta separada".

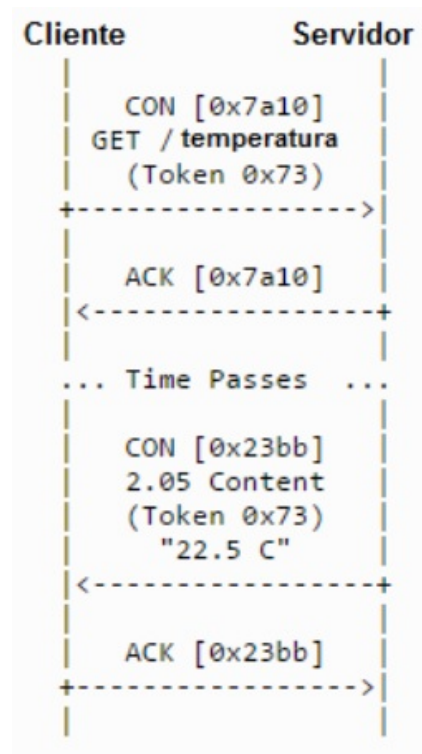


Figura 2.15: Representação do Método GET com Resposta Separada. Adaptado de: [Shelby et al., 2014]

Vale destacar também que embora o HTTP seja amplamente utilizado, ele não possui as características ideais no que diz respeito a se trabalhar com redes IoT, tendo em vista o tamanho relativamente grande dos seus pacotes para serem tratados por dispositivos com baixo poder de processamento. Diferentemente do HTTP que faz uso de TCP, CoAP usa UDP na camada de transporte para troca de mensagens. Pelo fato deste modo não necessitar de *handshaking*, torna-se assim mais leve e relativamente fácil de ser implementado em microcontroladores e dispositivos embarcados com baixa potência e baixa taxa de bits. Além destes aspectos, CoAP aproveita os mesmos códigos de resposta do HTTP, sendo responsável também pelos ACKs das mensagens, tratativas de congestionamento e duplicidade de detecções [Nazario et al., 2019].

3 Trabalhos Relacionados

Esta seção descreve trabalhos relacionados a aplicações de ambientes inteligentes, em especial, campi inteligentes com foco principal em estudos relacionados à comunicação com redes que utilizem sistemas embarcados, RSSF e projeto de IoT. Após a revisão bibliográfica, foi possível encontrar diversas abordagens por meio de dispositivos inteligentes, que associam eletrônica embarcada e sistemas computacionais. Para a composição da revisão bibliográfica, dois eixos principais foram pesquisados: i) aplicações de smart campi e ii) avaliação de desempenho de RSSF e sistemas de IoT.

[Del-Valle-Soto et al., 2019] realizou um experimento de avaliação de desempenho entre esquemas colaborativos e cooperativos em redes de sensores sem fio. Neste estudo, foi estabelecido um conjunto de métricas de desempenho para comparar, de modo objetivo, o desempenho de diferentes padrões de protocolos utilizados. Especificamente quatro dos protocolos mais populares de IoT foram selecionados: Zigbee, LoRa, Bluetooth e Wi-Fi. Objetivando-se testar e comparar seus desempenhos, vários sensores foram alocados em diferentes pontos de um campus universitário para criar uma rede esparsa capaz de simular com precisão uma cidade inteligente. Visando a obtenção de métricas pré-definidas, dois esquemas diferentes foram analisados, sendo um deles o modo colaborativo e também o cooperativo. Os resultados da análise demonstraram que no esquema cooperativo, o protocolo com menor consumo de energia é o ZigBee, o qual consome aproximadamente metade do protocolo LoRa. Por outro lado, BLE e Wi-Fi têm dados muito dispersos, chegando a consumir valores tão baixos quanto 1,1 Joules. No esquema colaborativo, os resultados se mostraram semelhantes, com o protocolo ZigBee tendo o menor consumo de energia. Tendo a principal diferença em relação aos dois modos de comparação avaliados, a dispersão dos dados no esquema cooperativo. Esta baixa dispersão demonstra que o consumo de todos os sensores são praticamente equiparáveis, com exceção do ZigBee que mostrou-se inferior nos dois casos.

[Wang et al., 2017] realizou um experimento concebido em um ambiente de campus inteligente na Universidade Nacional de Chiao Tung em Taiwan. No experimento foram implantados vários aplicativos IoT baseados em localização, incluindo LoRa e NB-

IoT. Avaliando-se o monitoramento de condições ambientais internas e externas, além de outros fatores relacionados, como temperatura e umidade. Ao longo do estudo proposto, foi investigado o desempenho de transmissão dos dispositivos finais LoRa para o *gateway*. Os autores ainda reportaram como as perdas de pacotes foram afetadas pela distância entre o dispositivo final e o *gateway*, a potência de transmissão, o comprimento da carga útil do pacote, o ângulo da antena, a hora do dia e as condições meteorológicas relacionadas. O padrão de perda de pacotes em LoRa também foi medido e descobriu-se que oscilações no desempenho relacionam dois ou mais agentes de atenuação de sinal. Além de também observar-se que mais de 99% das perdas de pacotes LoRa ocorrem com pelo menos três pacotes consecutivos.

[Marfievici et al., 2017] propõe a implementação de uma RSSF para avaliar um ambiente interno, composto por uma sala de *datacenter* de pequeno porte levando-se em consideração a existência de fatores dificultantes para um monitoramento preciso, por conta da localização dentro de um ambiente industrial tais como: temperatura, flutuações, ruído e grande quantidade de superfícies metálicas que causam alto nível de interferência eletromagnética. Foi montado um relatório baseado em 17 meses de observações de 30 nós sensores sem fio. Neste sistema foram medidos temperatura, umidade, nível de fluxo de ar, entre outros aspectos. Após um período inicial, uma avaliação de conectividade realizada na rede revelou um alto nível de ruído em alguns dos nós, ocasionados por conta da presença de diferentes fontes de interferência. Aumentando-se a configuração do CCA e realocando o posicionamento de alguns *sink nodes*, a rede conseguiu atingir uma confiabilidade de 99,2% nos últimos 8 meses de monitoramento. Deste modo, foi possível destacar a necessidade do uso adequado de ferramentas e protocolos confiáveis, além da definição de metodologias de projeto para gerenciar e implantar RSSF em ambientes do mundo real.

[Munoz et al., 2018] faz uma análise bastante abrangente acerca do padrão IEEE 802.15.4g demonstrando por meio de experimentos algumas aplicações e simulações em ambientes *indoor* e *outdoor*. Diversos testes foram realizados com o intuito de obter-se medições de alcance utilizando todo o escopo do padrão, realizando experimentos com distintos esquemas de modulação e abrangendo toda a parametrização na faixa que compreende o "*range*" de: 863-870 MHz, em quatro cenários diferentes. Os resultados obtidos

com os experimentos demonstraram que comunicações com alto nível de confiança e que usam taxas de dados de até 800 kbps podem ser totalmente alcançadas em ambientes urbanos a 540 m entre nós, desprezadas as incidências de interferência de menor porte.

[Neves, 2019] propõe a construção de uma aplicação para validar uma rede de sensores em ambiente urbano. O autor fez a implementação de um *smart meter*, em uma pequena rede LAN, com a função de medir o consumo de energia elétrica de uma instalação, enviando as leituras efetuadas para um sistema de banco de dados que armazena e processa estes dados convertendo-os em informações relevantes. Por meio da implementação de um contador de digital, desenvolvido através de um microcontrolador *Raspberry Pi* de baixo custo, o autor fez um monitoramento baseado em subgrupos, objetivando-se medir grandezas elétricas, tais como: potência real e equivalente, tensão e corrente. Ao final, o sistema foi aprimorado para não somente medir tais grandezas, mas também para mensurar se os níveis de consumo descritos em etiquetas dos principais eletrodomésticos encontrados em residências comuns, de fato, coincidem com os valores medidos pelo *hardware* proposto pelo autor. Para a transmissão das informações foi utilizado o protocolo Modbus para aquisição dos sinais de instrumentos e comandar os atuadores presentes no sistema.

[Pfeifer et al., 2018] propõe a utilização de uma Rede de Sensores sem Fio (RSSF) utilizando topologia em malha para a construção de uma aplicação de smart campi no edifício principal e prédios anexos da Universidade Federal de Santa Catarina (UFSC). O estudo foi conduzido entre os distintos prédios que compõem os Campi da instituição. Através do uso de um microcontrolador ESP32 programado em Linguagem C, foi desenvolvido um sistema sensorial que fez a coleta de grandezas como: corrente, tensão e temperatura. Na comunicação entre as aplicações e dispositivos medidores foi utilizado o protocolo MQTT. No que envolve a apresentação e avaliação dos dados obtidos, utilizando as ferramentas Node-Red e Blynk, foi desenvolvida uma página *WEB* e um aplicativo *mobile* em plataforma *Android* para o registro e apresentação dos dados em modo gráfico com o monitoramento em determinado instante e em função do tempo por meio de *logs* de registro. Neste estudo não foram realizados experimentos voltados a avaliação de desempenho, apenas testes de conectividade na rede em topologia *mesh* com suas possibilidades de configuração em um pequeno ambiente *indoor* para analisar o grau de eficiência do

sistema proposto em um período de 32 dias. Com relação ao campus universitário, os agentes responsáveis pela gerência do ambiente optaram pela não permissão de implantação do sistema, de fato. Deste modo, a rede foi validada somente com testes internos, voltados a experimentos testados em eletrodomésticos comuns a um ambiente residencial que concluíram níveis de consumo, de fato, próximos aos nominalmente descritos pelos fabricantes.

[Berouine et al., 2017] faz uma análise acerca da implementação de uma plataforma para monitoramento e processamento de dados em tempo real. Além destes aspectos, os autores também propõem um modelo para detecção, modelagem e visualização em tempo real. Um protótipo foi desenvolvido, baseado na criação de um *cluster* composto por cinco dispositivos, sendo um principal configurado como mestre e outros quatro como escravos. Com o foco em testar o interfaceamento entre prédios distintos, são detalhados a arquitetura montada para monitoração das grandezas, desde os nós sensores, módulos Wi-Fi, placas microcontroladoras e *cluster* para mensuração dos dados. O estudo propõe uma arquitetura baseada em uma topologia física LAN. Na página de visualização, Vários gráficos foram apresentados, ilustrando os dados obtidos em função da variação de tempo para cada aplicação. Como foram utilizados painéis fotovoltaicos para alimentação de parte do sistema, os gráficos também ilustram o baixo consumo de energia propiciado pelo aproveitamento energético na alimentação dos nós sensores. No experimento foram coletados dados em tempo real durante um período de 24 horas, sendo um intervalo de cinco segundos para cada ciclo de medição. Quatro métricas de avaliação foram geradas: i) O consumo diário instantâneo para cada aparelho monitorado, ii) O consumo agregado diário utilizado por todos os aparelhos presentes em uma guarita de segurança, iii) O consumo total desagregado de cada dispositivo, iv) O consumo total agregado por toda a barreira de segurança. Ao final, o sistema mostrou-se eficiente ao atingir os desafios elencados e com resultados satisfatórios no *cluster* implementado, exibindo um relatório das aplicações de controle com alto grau de autonomia.

[Metering et al., 2017] propõe um estudo que analisa a relação entre a crescente alta da cobrança tarifária energética e o "roubo" (desvio inapropriado) desta por parte da população. Por meio da utilização de um microcontrolador ATMEGA 328P para detectar e gerenciar o medidor de energia, este sistema implementado atua por meio da desconexão

Tabela 3.1: Relação de Trabalhos Relacionados

Autores	Propostas
Del-Valle-Soto et al., 2017	Um experimento de avaliação de desempenho entre esquemas colaborativos e cooperativos em redes de sensores sem fio
Wang et al., 2017	Investigado o desempenho de transmissão dos dispositivos finais LoRa para o gateway.
Marfievici et al., 2017	Sistema de monitoração de Datacenter em um intervalo longo de tempo avaliando qualidade e eficiência da transmissão em distintos níveis de interferência.
Muñoz et al., 2018	Avaliação detalhada acerca do padrão IEEE 802.15.4g realizando experimentos em ambientes indoor e outdoor em uma abrangente faixa do espectro eletromagnético a partir de distintas técnicas de modulação.
Neves, 2019	Proposta de um sistema de monitoração do consumo de energia em instalações elétricas por meio da utilização de um microcontrolador Raspberry Pi, medindo potência real, equivalente, tensão e corrente.
Pfeifer et al., 2018	Utilização de RSSF com topologia em malha para monitoração em tempo real do consumo de energia no edifício principal e prédios anexos da UFSC.
Berouine et al., 2017	Implementação de um sistema para modelagem e visualização de dados em tempo real, por meio de dispositivos sensores alimentados por painéis fotovoltaicos.
Metering et al. 2017	Proposta de dispositivo que monitora possível "roubo" de energia através de um sistema que utiliza rede de telecomunicações para prover camada de transmissão em tempo real.

e reconexão remota da linha de serviço de um consumidor específico. Por meio do envio de um SMS (*Short Message Service*) automático ao servidor central da concessionária através do módulo GSM (*Global System for Mobile Communications*) sempre que as atividades não autorizadas forem detectadas, uma mensagem separada será enviada de volta ao microcontrolador para efetuar a desconexão daquele agente não autorizado. A arquitetura dos dispositivos utiliza um transformador de tensão responsável pela conversão entre as entidades atuantes e um relé de estado sólido, que implementa o chaveamento entre conexão e desconexão dos dispositivos. Tomando-se por base este padrão descrito, os autores ainda propõem um método intercalando o recurso GSM em medidores inteligentes com os relés de estado sólido para lidar com possíveis perdas não técnicas e alterações provocadas por flutuação de tensão.

Baseado em um prévio levantamento bibliográfico e nos estudos elencados acima, observa-se que existe um quantitativo relativamente considerável de estudos que focam, de modo geral, em comunicação em *smart* campi e ambientes inteligentes de modo geral por meio da utilização de redes de sensores. Cada estudo desse tipo, propõe a sua própria arquitetura, focando nas técnicas utilizadas, realizando experimentos com protocolos

distintos de IoT e avaliando a eficiência no eixo de comunicação. Baseando-se nessa perspectiva, este trabalho busca ir além, sendo o ponto chave do estudo, a implementação do conceito de "reuso" da infraestrutura de monitoramento criada, aplicando essa infraestrutura em um conceito mais genérico voltado a adição de camadas adicionais no que diz respeito aos princípios da confiabilidade e disponibilidade, obtidas por meio das técnicas de diversidade e persistência de dados.

Vale ainda destacar que nenhum dos estudos citados neste capítulo de Trabalhos Relacionados utilizou o conceito de diversidade a nível de modulação e receptor associados a comunicação bidirecional com mais de um *gateway*. Já o estudo produzido nesta dissertação, explora um cenário mais complexo por trabalhar com comunicação simultânea em mais de um gateway, fator este que demanda um nível de eficiência mais preciso da aplicação de controle, para a verificação de ociosidade no meio de transmissão quando da ocorrência de possíveis colisões e para determinar a origem de cada pacote transmitido ou recebido. Outro diferencial que torna o cenário deste trabalho mais complexo que os demais estudos citados é com relação a comunicação com pacotes de confirmação (Pacotes ACK) entre o receptor e o transmissor.

4 Sistema de Gerenciamento de Aplicações de Smart Campi

Esta seção descreve a arquitetura proposta no trabalho, delimitando-a em três partes: i) arquitetura do sistema com o detalhamento dos componentes e agentes envolvidos; ii) detalhamento do RAW Manager com as suas características de comunicação e iii) o estudo de caso utilizado como estratégia para validação da rede.

4.1 Arquitetura do Sistema

Conforme descrito anteriormente, um dos objetivos específicos deste estudo é prover a capacidade de utilizar mecanismos de diversidade de transmissão, por meio do uso em modo simultâneo de diferentes esquemas de modulação, visando atingir um alto nível de disponibilidade e confiabilidade para os pacotes trafegados na rede. Para que esses mecanismos funcionem de forma adequada, se faz necessário um sistema centralizado para melhor condensar o gerenciamento das aplicações relacionadas e para realizar a configuração dos dispositivos da rede e organizar as informações coletadas por diferentes *gateways*, que por sua vez podem receber pacotes por meio de diferentes módulos de comunicação, embora saibamos que este fato também pode determinar um ponto único de falha. Com sua abordagem genérica, a infraestrutura de comunicação proposta nesta dissertação pode ser posteriormente reaproveitada para a implantação de diferentes aplicações, por exemplo, para monitoramento de consumo de energia e análise da rede elétrica, visando aumento de eficiência energética em um campus inteligente.

4.1.1 Descrição da Arquitetura Lógica

A Figura 4.1 ilustra a representação da comunicação dos nós finais com os *gateways* por meio da utilização do padrão IEEE 802.15.4g com configurações distintas de modulação. Essa arquitetura foi considerada inicialmente em [Tuset-Peiró et al., 2020], em que uma RSSF com 11 nós finais e um *gateway* foi implantada, usando três configu-

rações distintas de modulação (SUN-FSK, SUN-OQPSK e SUN-OFDM), todas provendo 50 Kbps de taxa de bits. O sistema de gerenciamento proposto nesta dissertação usa a perspectiva de um cenário mais genérico, com possibilidade de existência de diversos *gateways* e considerando a combinação do uso de diversidade a nível de modulação e receptor. Além disso, o trabalho descrito em [Tuset-Peiró et al., 2020] foi realizado em um ambiente industrial, enquanto que o foco do presente trabalho é em um ambiente de Campus Inteligente, com o diferencial da comunicação com mais *gateways* e com a utilização de pacotes de ACK entre o eixo de receptor e transmissor.

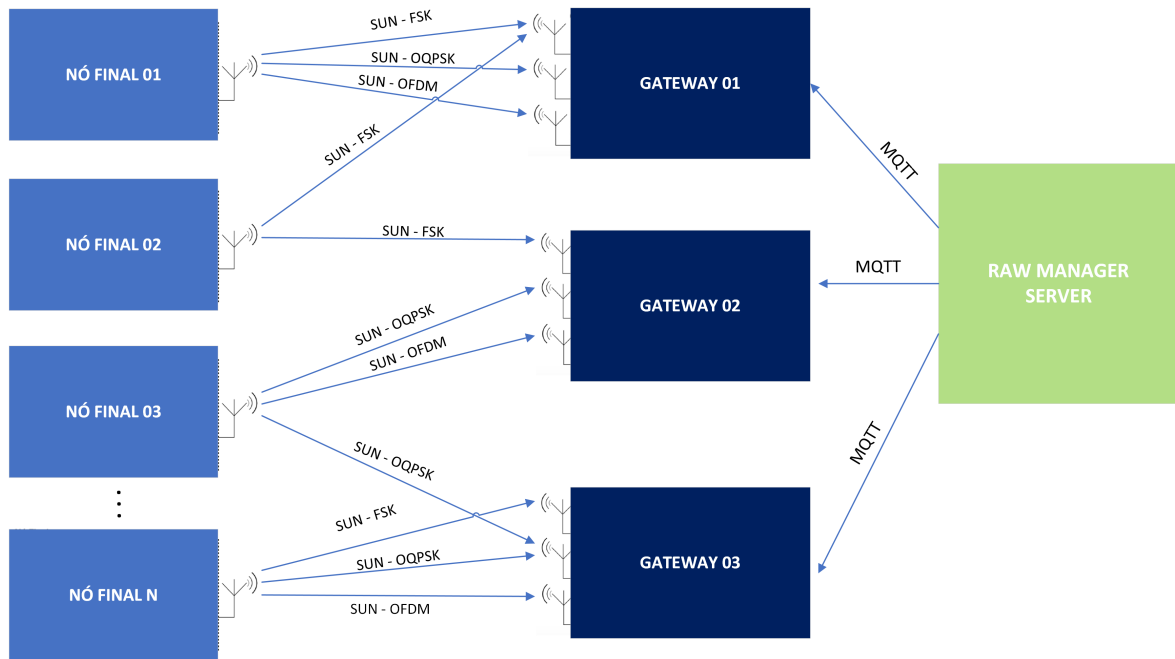


Figura 4.1: Comunicação Entre os Nós Finais e o Gateway. Adaptado de: [Gomes., 2020]

Com relação aos protocolos de camada de aplicação, propõe-se uma segmentação em dois grupos distintos, sendo um deles voltado a uma abordagem de mais baixo nível, que lida com a transmissão de bits em camada física e protocolos, que não são visíveis diretamente ao usuário final. Enquanto a outra parte aborda a camada de aplicação, do ponto de vista das aplicações que executam na nuvem e oferecem serviços aos usuários finais, seus processos e agentes envolvidos.

No primeiro grupo, as aplicações que executam na nuvem não são consideradas, sendo o foco neste ponto o provimento de um canal de comunicação confiável entre os nós finais, dotados de sensores ou atuadores, e o agente intermediário centralizador (*gateway*).

No *gateway*, os pacotes recebidos a partir dos nós sensores são convertidos utilizando-se a API para gerar novos pacotes em formato MQTT, que é o protocolo considerado para comunicação entre as aplicações que executam na nuvem com a rede de sensores/atuadores. Vale destacar que um fator impeditivo de usar um protocolo único de IoT para prover o canal de comunicação nesta infraestrutura proposta, deve-se ao fato de o MQTT necessitar de conexão à Internet para a troca de dados e os microcontroladores acoplados aos nós sensores considerados na rede proposta não possuem placa de rede para conexão direta com a Internet sem fio. Por este motivo, na infraestrutura criada, os módulos de rádio vão se comunicar por meio de uma rede local sem fio de baixa potência. A partir do *gateway* em diante é que a comunicação contará com acesso à Internet.

Vale destacar, também, o porquê da escolha de se trabalhar com diferentes tecnologias, em vez de unificar a comunicação em um único protocolo de IoT. Esta característica deve-se ao fato da ausência de conectividade com a *web* da cadeia de comunicação entre os nós finais (responsáveis pelo sensoriamento) até a chegada no *gateway*. Seria possível utilizar, neste trecho de comunicação, o protocolo HTTP, por exemplo, mas optou-se pela não utilização, pelo fato deste ser além de específico para IoT, várias vezes mais leve que o HTTP. Além do fato de já existirem bibliotecas prontas que facilitam amplamente a programação no protocolo.

No nó final, as informações a serem transmitidas são organizadas na carga útil do pacote. Quando o pacote é recebido no *gateway*, além dos dados dos sensores, outras informações também devem ser coletadas. Um pacote de dados customizável a partir das informações coletadas foi criado para encapsular todas as informações de interesse. Esta carga útil é composta por um *Array* contendo diversos campos de informações relevantes, tais como: RSSI, identificador do pacote e outras métricas referentes à qualidade da transmissão nos enlaces de comunicação. Com relação ao valor gerado de RSSI, foi utilizado o próprio conversor integrado do módulo de rádio, o qual já faz uma conversão automática transformando um dado inteiro, normalmente entre 0 a 255, para um valor em dBm, de modo a estimar a potência de recepção do pacote. Uma vez que estas informações chegam ao *gateway*, tais dados são encapsulados em um formato compreendido pelas aplicações, neste cenário, usando o MQTT.

A Figura 4.2 ilustra o detalhamento dos métodos de comunicação agregados entre o nó final e o *gateway*. De acordo com a imagem, pode-se verificar os dois principais métodos CoAP utilizados para troca de informações entre a rede de sensoriamento composta pelos nós finais e os *gateways*.

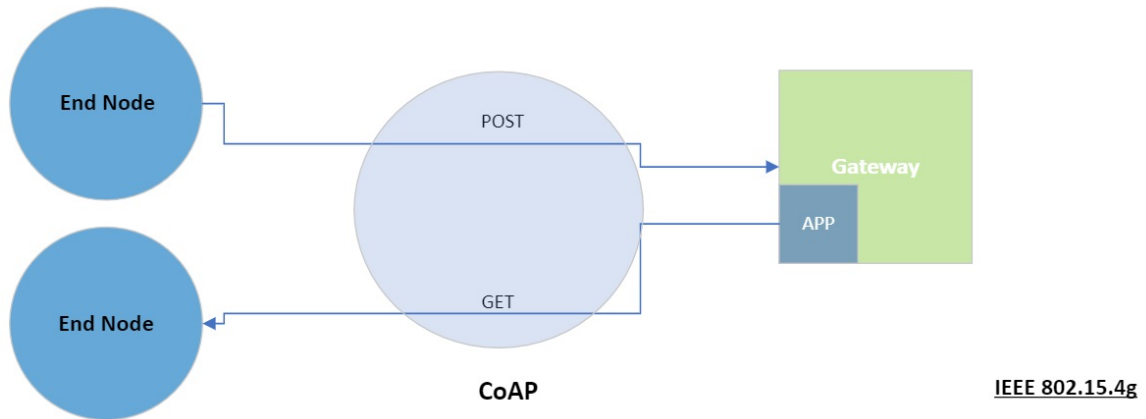


Figura 4.2: Detalhamento dos métodos de Comunicação entre o Nó Final e o *Gateway*. [Autoria Própria, 2020]

Em nível mais alto, as aplicações comunicam-se com o *gateway* através do protocolo MQTT. Tomando-se por base o modelo *publish/subscriber*. Neste modelo de comunicação, existe a figura do *Broker*, que uma vez alocado em nuvem, permite a criação de entidades chamadas de “Tópicos”, os quais permitem que uma determinada aplicação interessada naquele grupo de dados relativos a um determinado sensor cadastre-se para receber qualquer informação que seja cadastrada nesta lista.

4.1.2 RAW Manager

A Figura 4.3 representa o detalhamento do RAW Manager e da comunicação entre os *gateways* e as aplicações finais de IoT. Podemos descrever o RAW Manager como sendo uma aplicação de controle executada dentro de um servidor de aplicação na nuvem, se comunicando com os *gateways* para coletar informações processadas pelo banco de dados. Ainda na Figura 4.3 temos o detalhamento das funções envolvidas e associadas ao RAW Manager. A FN1 representa a verificação de inconsistências e duplicações dentro da aplicação de controle, a qual gerencia o pacote recebido. A FN2 descreve a comunicação entre o banco de dados e a aplicação de visualização, por meio da troca de mensagens

através de uma linguagem de consulta estruturada (SQL). Por sua vez, FN3 descreve a capacidade das aplicações finais de IoT também enviarem ações de controle ou requisição de dados, por meio dos seus atuadores. FN4 a representação do canal de comunicação responsável por acessar as informações dos pacotes que chegam pela rede.

Ainda a nível de aplicações, foi implementada uma funcionalidade responsável pela persistência de dados. Tal ferramenta é responsável pela leitura dos dados via MQTT e armazenamento das informações obtidas em um Banco de Dados de séries temporais (InfluxDB), que é uma ferramenta amplamente indicada para cenários em que necessita-se de armazenamento de dados que sofrem variações em função do tempo. Vale destacar também que um mesmo pacote pode ser recebido por mais de um *gateway*, caso eles estejam com modulações compatíveis e no alcance de um determinado nó final. Dessa forma, antes de cadastrar um pacote no *Broker*, o RAW Manager fica responsável, também, pelo tratamento dessas duplicações.

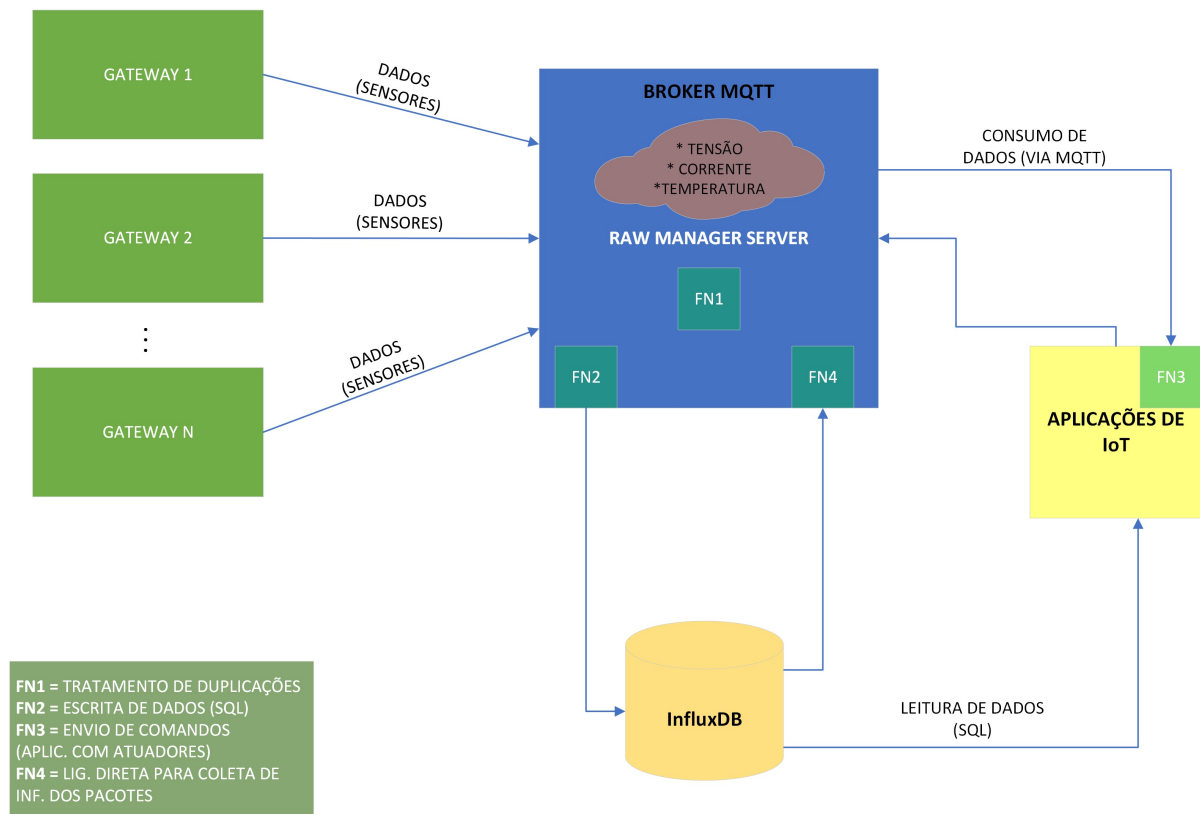


Figura 4.3: Detalhamento do RAW Manager com troca de Mensagens Via MQTT. [Autoria Própria, 2020]

No que envolve o *Broker* MQTT, cada caminho percorrido é formulado por um

pacote de dados (*Payload*). Este pacote é responsável pela transmissão das informações com ligação direta em tópicos MQTT. O módulo de software implementado para prover a comunicação neste ponto, atualiza dinamicamente o quantitativo de tópicos cadastrados para cada mensagem contida no pacote registrado no banco de dados. No exemplo da Figura 4.3, apenas para melhor entendimento da comunicação, são definidos três tópicos, com representação de sensoriamentos de: tensão, corrente e temperatura.

4.2 Definição da Arquitetura Física e Ambiente de Validação

Nesta subseção é detalhada a arquitetura física da rede e da aplicação proposta como estudo de caso, para validar a comunicação composta tanto pelos nós finais com seus sensores acoplados até os *gateways* e as aplicações integradas.

A UFPB conta com diversos centros e unidades de apoio para atender a mais de 39.000 estudantes de graduação e pós-graduação. O protótipo inicial desenvolvido contempla um dos centros da instituição, mais especificamente o CI (Centro de Informática), para realização dos testes iniciais. Para composição da estrutura, inicialmente foi concebida uma topologia em estrela com a inclusão de 8 nós para se avaliar a capacidade de transmissão entre eles e observar inicialmente a incidência de fenômenos relacionados a confiabilidade da transmissão, tais como: taxa de entrega de pacotes, atenuação de sinal e possíveis interferências em ambiente *outdoor*.

Existem no Campus I aproximadamente 80 transformadores de alta tensão. A distância média entre estes pontos é cerca de 35-40 metros. Esses pontos foram usados como referência pensando em uma aplicação de monitoramento de energia, que foi a aplicação que motivou o início dos estudos deste trabalho. Por conta da relativa proximidade entre os pontos de medição, optou-se pela implementação do sistema com topologia em estrela, objetivando assim uma maior simplificação do projeto como um todo.

O sistema embarcado utiliza um transceptor OpenMote B, que utiliza tecnologia IEEE 802.15.4g para permitir sua integração na rede implantada. Baseado no modelo proposto por [Tuset-Peiró et al., 2020], os *gateways* representados na Figura 4.4 são com-

postos por um Raspberry Pi (SBC - *Single Board Computer*) conectado a três módulos OpenMote B para o recebimento dos pacotes enviados a partir de 3 diferentes esquemas de modulação. Dentro do *gateway* é executado GNU / Linux com um *Script* Python responsável pela leitura individual de cada pacote recebido pelo transceptor e convertendo-o a partir do pacote bruto para uma carga útil, posteriormente encaminhando para o *back-end* da nuvem usando MQTT. A cada ciclo de transmissão, cada novo pacote gerado pelo nó final é enviado 3 vezes em cada modulação por minuto.

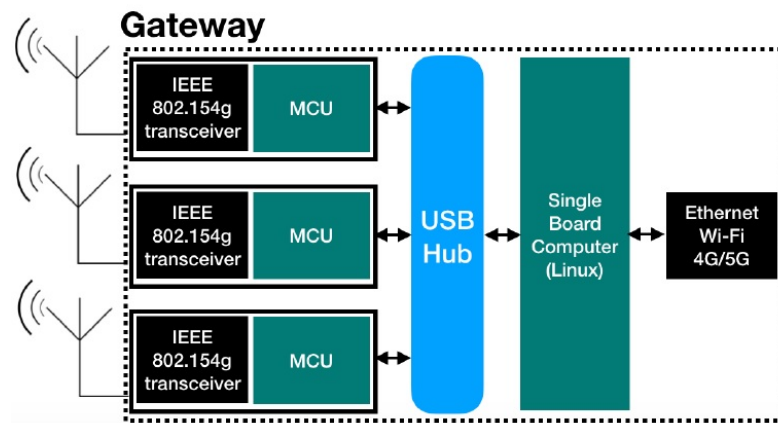


Figura 4.4: Gateway Internamente. Adaptado de: [Tuset-Peiró et al., 2020]

A Figura 4.5 ilustra o escopo geográfico com as respectivas medidas necessárias para a identificação do quantitativo de nós a serem utilizados na rede de sensores. Os *gateways* ficam alocados no prédio principal do Centro de Informática. Além disso, os *gateways* são posicionados em um local em que possam acessar à Internet preferencialmente por meio de uma rede cabeada. A partir da implantação dessa rede, foi possível validar o sistema proposto nessa dissertação e realizar por meio dele, a coleta de informações que permitirá, por exemplo, avaliar o nível de atenuação de sinal para cada nó e a qualidade dos enlaces, para as diferentes modulações consideradas, durante a operação da rede.

4.3 Tecnologias e Componentes Utilizados

Esta proposta de arquitetura toma como base o princípio da não-heterogeneidade, ou seja, utilizar o mínimo de tecnologias distintas possível. Decidiu-se por aplicar ferramentas que possuam uma integração nativa entre si, ou caso inexista esta característica, que ela se dê de forma mais transparente possível tanto para o usuário final como para o



Figura 4.5: Delimitação Geográfica da Área Teste. [Autoria Própria, 2020]

agente responsável pela administração e gerenciamento da infraestrutura.

Tendo em vista esta perspectiva, optou-se pela programação do sistema em linguagem Python por esta ser uma linguagem amplamente difundida, simples de se trabalhar e com um quantitativo bastante expressivo de programadores, o que permite a manutenção simplificada do sistema bem como a sua evolução futura.

No desenvolvimento das aplicações *back-end* optou-se pela integração da linguagem Python com o framework Flask, que pode ser compreendido como um micro framework que implementa o mínimo necessário para se otimizar o desenvolvimento de aplicações *web*. Sua estrutura é bastante simples e leve, além da capacidade de ir se adaptando por meio da instalação de módulos adicionais de forma incrementável para a exigência de cada aplicação distinta. Vale destacar que a “leveza” do framework é fator crítico para a sua utilização em qualquer sistema de IoT, por conta das limitações a nível de processamento e consumo inerentes a seus dispositivos.

4.3.1 OpenMote B

Na composição dos 8 nós finais utilizados na rede, são utilizados módulos OpenMote B, que contam com um transceptor Atmel AT86RF215. O transceptor Atmel AT86RF215 implementa todas as definições de camada física do IEEE 802.15.4g. Este dispositivo foi concebido com foco em IIoT (*Industrial Internet of Things*).

O OpenMote B é um *hardware* para desenvolvimento e prototipação de plataformas IoT, composto por um núcleo CC2538 de 512 kb Flash e 32 kb de RAM. Possui módulo de Rádio Atmel AT86RF215 SubGHz (868/915MHz), suportando todas as modulações IEEE802.15.4g e operação simultânea de rádio duplo. O OpenMote B pode ser integrado com a Raspberry e pode ser programado por plataformas *OpenSource*. Baseado no kernel do FreeRTOS, sua configuração é possível de acordo com as necessidades do sistema, sendo possível também a configuração de uma aplicação IoT usando a ferramenta *OpenSource*, “contiki” . Com relação ao aspecto de segurança, o OpenMote B possui suporte aos algoritmos criptográficos: HA2, AES-128/256, ECC-128/256 e RSA. Além deste aspecto, o seu alcance espectral é provido por dois conectores de antena SMA para antenas Sub-GHz e 2,4 GHz, sendo o conector da antena Sub-GHz diretamente conectado ao rádio no AT86RF215. Esta antena de 2,4 GHz é multiplexada usando um comutador RF nos transceptores de rádio CC2538 e AT86RF215. [Dallagasperina, 2017]

O código-fonte do *firmware* dos dispositivos está presente no repositório [Tuset-Peiró and Vilajosana, 2019] e nele foram realizadas algumas alterações, disponíveis no repositório em [Ferreira et al., 2021].

4.3.2 Raspberry Pi 4

Consiste em um pequeno computador com todo o seu *hardware* integrado em uma placa única SoC (*System on a chip*), com larga aplicação na indústria, casas inteligentes e em ambientes voltados a IoT. No contexto específico deste estudo, as referências atribuídas ao Raspberry Pi o denominam como SBC (*Single Board Computer*) e especificamente para os experimentos realizados foram utilizadas duas Raspberry Pi 4 Modelo B, que contam

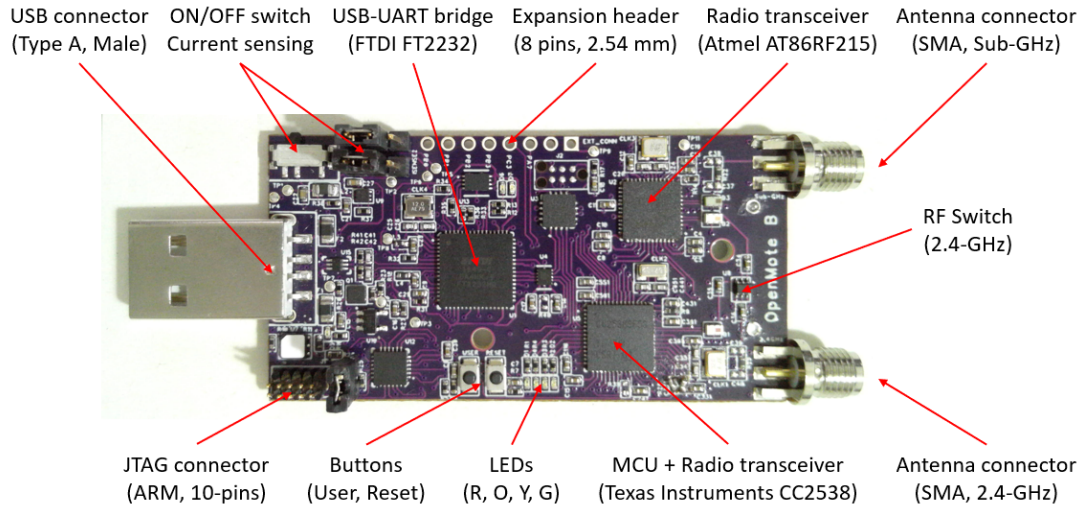


Figura 4.6: Exemplo de um OpenMote B [Tuset-Peiró et al., 2020].

com Processador Broadcom BCM2711 quad-core (4x Cortex-A72 a 1.5GHz) de 64 bits, armazenamento microSD de 32GB, chip gráfico VI de 500 MHz, OpenGL ES 3.0 e 4GB de RAM, ambas utilizando um ambiente Linux Raspbian que recentemente passou a denominar-se Raspberry Pi OS.

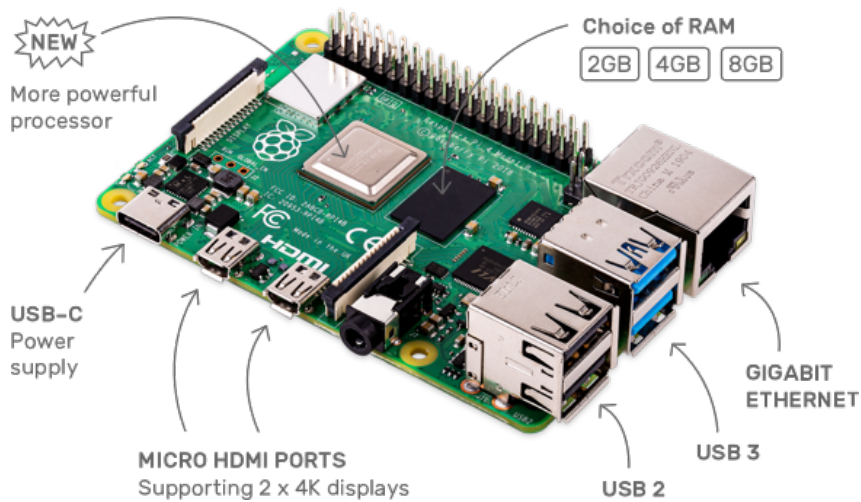


Figura 4.7: Raspberry Pi 4 Modelo B Adaptado de: [Inc., 2021]

4.4 Implementação do RAW-Manager

Este subcapítulo traz o detalhamento das etapas de transmissão de dados desde os nós finais até a entrada de dados via porta serial nos *gateways* da rede. Em seguida trazendo uma visão geral do processo de registro dos dados com fluxo bidirecional, sendo

uma destas saída de dados para o InfluxDB e outra para o Broker MQTT. Por fim, serão explicados detalhes adicionais referente a página WEB, responsável por exibir os dados coletados na rede.

4.4.1 Transmissão de Dados

Os nós finais foram configurados para transmitir os dados tomando-se por base um determinado ciclo de execução, conforme ilustrado na Figura 4.8. Neste nível de comunicação, os nós finais são compostos por placas transceptoras do tipo OpenMote B. Cada um dos 8 nós finais foram configurados através do processo de gravação de *firmware* como Tx (transmissores), para realizar o envio de pacotes de dados. Na construção dos nós finais, objetivando-se a simplificação do projeto em si com foco na economia de recursos, ao invés de utilizar um OpenMote B para transmitir em cada modulação distinta, o *firmware* utilizado permite o envio de pacotes nas três modulações utilizando o mesmo *hardware*, de maneira intercalada, sendo: SUN-FSK (Modulação 0), SUN-OQPSK (Modulação 1) e SUN-OFDM (Modulação 2).

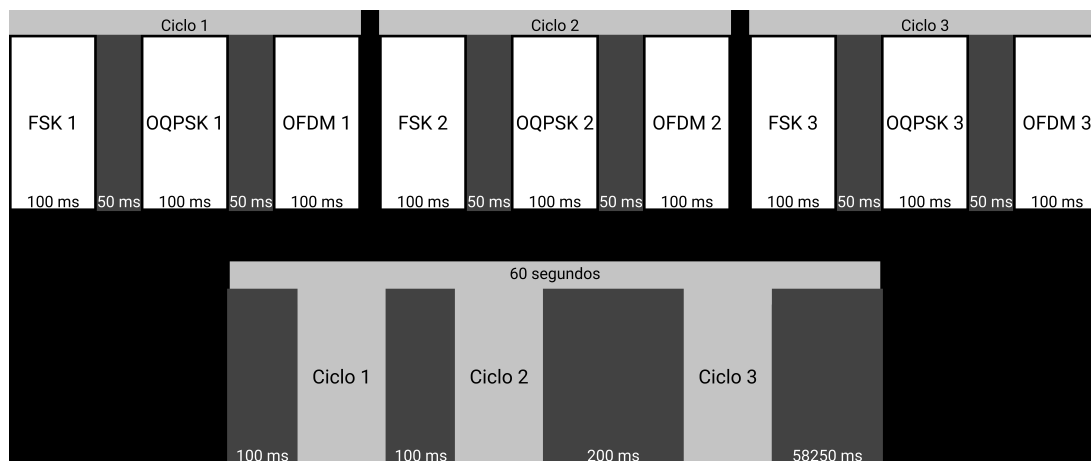


Figura 4.8: Representação do ciclo de transmissão. Adaptado de: [Bezerra Silva, 2021]

Em cada ciclo de transmissão, o dispositivo realiza o envio de três pacotes de dados, sendo uma transmissão para cada modulação com intervalo de 50 ms entre as transmissões. A cada ciclo existe um delay de 100 ms, entre o primeiro e o segundo ciclo, ou 200 ms entre o segundo e o terceiro ciclo. A cada envio nas três modulações, o que constitui um ciclo de transmissão completo, o dispositivo entra em espera por um período de 58250 ms. Com relação ao comprimento do payload, este, possui um total de 32 *bytes*.

4.4.2 Escrita no Banco com Saída Bidirecional

A Figura 4.9 ilustra a arquitetura simplificada que foi, de fato, aplicada. O processo de comunicação inicia-se com a aquisição dos dados via porta serial, dados estes que são posteriormente tratados pelo módulo de *software* desenvolvido e sequencialmente transmitidos via fluxo bidirecional, ou seja, repassados em paralelo tanto para o banco de dados InfluxDB como para o módulo que trata da comunicação via MQTT. Testes iniciais foram realizados utilizando a comunicação em ambos os protocolos CoAP e MQTT. No entanto, de maneira a aproveitar a base de código do repositório do OpenMote B já disponibilizado, foi utilizado um formato simplificado para a composição do pacote, sem a necessidade de inclusão dos cabeçalhos CoAP.

Ainda sobre a Figura 4.9, quando neste texto é referido a saída bidirecional, esta, diz respeito a característica de transmissão distinta entre o banco de dados de séries temporais InfluxDB e o *broker* MQTT. Enquanto o primeiro utiliza um registro padrão para dados não-relacionais, o segundo se comunica tomando-se por base uma estrutura de fila, fazendo-se uma analogia semelhante ao modelo FIFO (*First-in First-out*), ou seja, o primeiro que valor que entra, é o primeiro que sai da fila.

Em relação ao MQTT, o protocolo foi implementado tomando-se por base a utilização do *Broker*, sendo o protocolo em sua versão 5.0, versão esta mais atual até o momento da escrita deste estudo. Foi optada a utilização do *Broker* nesta versão por tratar-se de uma implementação bem documentada, de software livre (licenciado por EPL/EDL), ideal para ambientes que operam com sensores de baixa potência, dispositivos móveis, microcontroladores e para a construção da comunicação do lado cliente, foi utilizada a biblioteca PAHO que possui um excelente nível de integração com a linguagem Python.

Com o foco em determinar a origem do pacote recebido, foi desenvolvida uma estrutura de identificação baseada no endereço MAC de cada dispositivo responsável pela transmissão na rede. Essa estrutura de identificação toma por base a criação de um prefixo para cada pacote recebido. Com relação aos tópicos, foi atribuída a relação 1 para 1, ou seja, a cada campo contido no pacote trafegado, um tópico foi cadastrado para o

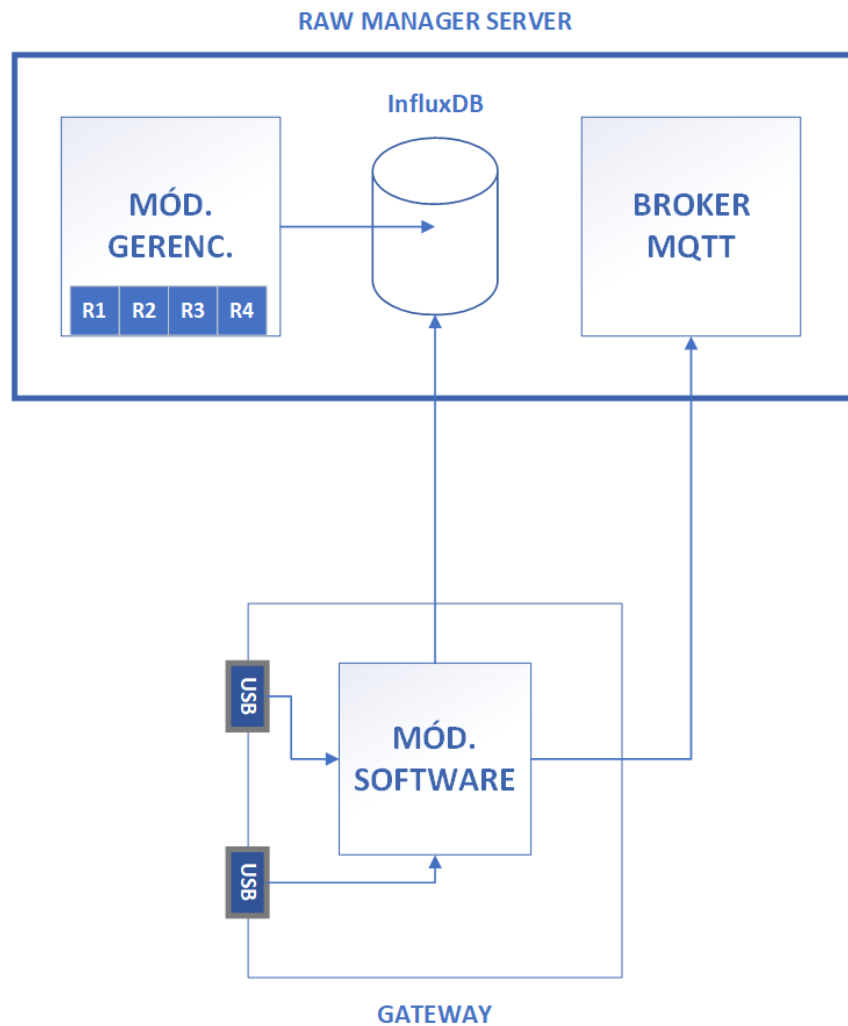


Figura 4.9: Comunicação com saída bidirecional. Autoria Própria, 2021

seu atributo correspondente. Por exemplo, se no pacote são transmitidos 6 valores de informações gerais, tais como: RSSI, tipo de modulação, tentativas de retransmissão, etc, para cada um destes 6 valores teremos um tópico vinculado.

4.4.3 Monitoramento em Página WEB

Para visualização dos dados transmitidos, foi desenvolvido um monitor de rede acessado via página WEB utilizando tecnologias HTML (*HyperText Markup Language*) e JavaScript. A referida página é capaz de exibir, em tempo real, o quantitativo de *gateways* ativos e que encontram-se escrevendo dados no banco, naquele instante. Uma verificação periódica de atividade é informada ao usuário em intervalos regulares a cada 5 s. Se por

ventura ocorrer uma desconexão, seja por falta de alimentação, ruptura na rede ou algum outro fator externo, um log de atividades adiciona o registro de falha com o detalhamento da desconexão e registro de data e hora da ocorrência. O monitor de rede desenvolvido ainda conta com diversos filtros capazes de refinar os resultados por: i) identificação do nó, tipo de modulação, identificador do *gateway*, filtragem temporal para início e fim da série.

Além da filtragem de resultados, o monitor de rede mostra um contador com informações do número de pacotes recebidos e esperados, taxa efetiva por minuto e métricas relevantes, tais como: RSSI e PDR. Em especial nestes dois últimos, o sistema também é capaz de mostrar graficamente as variações em função do tempo, por meio da utilização de filtros de consulta SQL que podem ser acessados a partir do delimitador que marca o início e fim de cada série. Por fim, o monitor de rede também é capaz de exibir os registros do banco por pacote recebido. Por meio da técnica de prefixo descrita anteriormente, além das informações contidas no banco, o usuário também tem acesso ao endereço MAC do *gateway*, para desta forma, ser capaz de determinar com precisão a origem do pacote.

5 Validação do Sistema Proposto

Neste capítulo são descritas as etapas executadas para validação da arquitetura de comunicação proposta. Desde a preparação do ambiente até a análise dos resultados obtidos.

5.1 Preparação do Ambiente

A rede é composta por 8 nós finais configurados como dispositivos transmissores, denominados ao longo deste capítulo como Tx, responsáveis pelo envio de 9 mensagens, logo, sendo 3 em cada variante de modulação do padrão IEEE 802.15.4g SUN, por minuto, por nó. Cada *gateway* conta com 3 receptores, denominados ao longo deste capítulo apenas como Rx, configurados para receber os dados em três modulações distintas em simultâneo.

Os nós Rx enviam os pacotes recebidos via IEEE 802.15.4g por uma porta USB para um computador que utiliza um sistema operacional de base GNU/Linux. Neste contexto específico foram utilizados dois SBC Raspberry Pi 4 modelo B (um por *gateway*) com sistema operacional Raspbian (atualmente denominado Raspberry Pi OS). Uma vez que os dados são recebidos localmente pelos módulos de rádio do *gateway*, um *script Python* é responsável pela leitura das mensagens recebidas, pelo processo de encapsulamento em um formato de dados manipulável, a persistência das informações no banco de dados InfluxDB e pela saída bidirecional da aplicação com escrita paralela no banco e no broker MQTT. A Figura 5.1 ilustra a arquitetura real utilizada no domínio do experimento. De acordo com a mesma, podemos observar a representação dos nós finais transmitindo pacotes de dados em três modulações distintas para serem capturados pelos nós receptores acoplados, via porta serial, aos dois *gateways*. Sequencialmente, temos o registro e posterior tratamento dos dados através do módulo de gerenciamento contido internamente ao RAW-Manager.

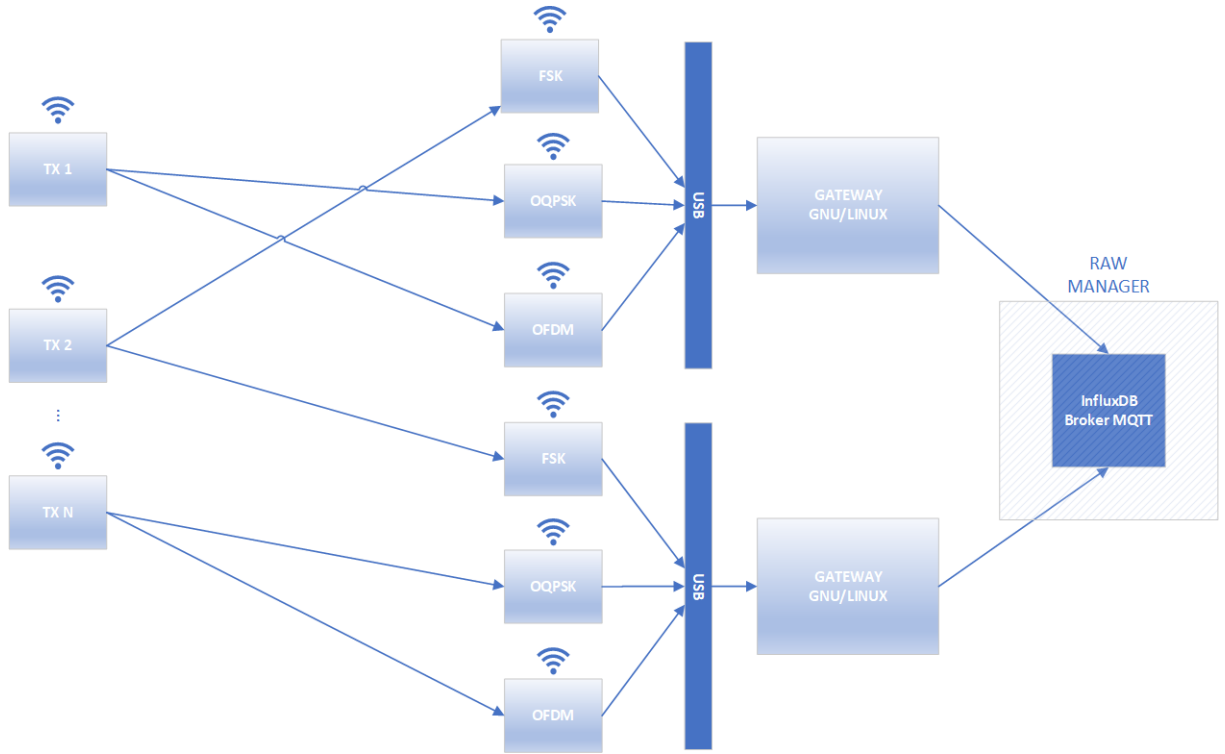


Figura 5.1: Arquitetura no Domínio do Experimento. [Autoria Própria, 2021]

5.2 Métricas de Avaliação de Desempenho

O objetivo geral desta arquitetura proposta foi avaliar a qualidade da comunicação em um ambiente externo por meio da utilização de técnicas de diversidade em nível de modulação e receptor. Deste modo, um dos objetivos específicos deste estudo foi observar o comportamento dos enlaces de comunicação baseados em métricas de avaliação de desempenho previamente definidas, tais como PDR e RSSI. Sobre o primeiro, a análise foi desdobrada em duas camadas distintas, sendo, camada física e camada de aplicação. Na camada física a análise do PDR leva em consideração a efetiva taxa de entrega de pacotes no meio, independente da quantidade de vezes que o pacote foi enviado e do quantitativo de retransmissões. Em relação a camada de aplicação, esta análise considera os pacotes recebidos, sendo indiferente, para a aplicação em si e para o agente consumidor dos dados, se os dados chegaram 1 ou 3 vezes, ou ainda em qual modulação o dado foi recebido. Deste modo, o objetivo da análise em camada de aplicação é abstrair a complexidade inerente ao meio físico, concentrando-se apenas em se a informação foi recebida e não em como esta informação é recebida. Com relação a segunda métrica predefinida, o RSSI, este avalia o quão bom um dispositivo consegue detectar, transmitir e ouvir sinais

vindos de algum elemento ativo na rede. Para a construção deste modelo de avaliação de desempenho, criou-se uma estrutura e cada mensagem transmitida é constituída pelos seguintes campos:

- Node ID: é o identificador do dispositivo. Constitui-se de um campo de 6 *bytes* que armazena um identificador entre "a" e "h", constituindo assim, 8 entradas, associadas a cada nó final;
- Identificador de Pacote: é o registro único associado a cada pacote transmitido na rede. Constitui-se de um campo de 8 *bytes* que armazena um contador responsável pela identificação do pacote;
- TX Mode: é o identificador do tipo de modulação. Constitui-se de um campo de 1 *byte* que registra qual modulação o pacote foi enviado;
- Identificador do Pacote do Transmissor: é o registro inerente ao ciclo de transmissão realizado, o qual compreende o intervalo entre "1" e "3". Este campo também possui 1 *byte*;
- Tentativas de CSMA: refere-se ao quantitativo de verificações de ociosidade do meio para iniciar a transmissão. Também possui um intervalo entre "1" e "3" tentativas. Caso este limite seja excedido sem ser possível a transmissão, o pacote é descartado;
- RSSI do CCA: é o identificador do nível de energia do canal. Constituído por um campo de 1 *byte*. Caso este valor exceda um limiar predefinido, o Tx aguarda um tempo aleatório, em milissegundos, e realiza outra tentativa de transmissão.

5.3 Condução do Experimento

Conforme descrito anteriormente, o experimento foi realizado no Campus V da UFPB, mais especificamente, entre os Centros de Informática (CI) e o Centro de Tecnologia e Desenvolvimento Regional (CTDR). Ao todo foram espalhados 8 nós finais distribuídos em ambientes internos e externos pertencentes ao Campus. Destes 8 nós finais, 4 deles foram alocados internamente, espalhados ao longo de salas/laboratórios do prédio

principal do CI e os 4 nós finais restantes distribuídos nas áreas externas com distâncias geográficas variadas entre eles.



Figura 5.2: Distribuição dos Nós no Campus V da UFPB. [Autoria Própria, 2021]

A Figura 5.2 ilustra a distribuição dos 8 nós finais (Tx) implementados ao longo do Campus V, além dos 2 *gateways* responsáveis pela centralização das informações coletadas. Cada um destes *gateways* são dotados de 3 receptores (Rx) responsáveis pela captura dos pacotes nas modulações SUN-FSK, SUN-OQPSK e SUN-OFDM. A distribuição dos nós espalhados ao longo do Campus foi definida da seguinte forma:

- Rx 1: localizado no 3º andar do prédio do CI. Alocado no interior da sala do LAR - Laboratório de Redes;
- Rx 2: localizado no piso térreo do CI. Alocado no hall de acesso as salas de aulas;
- Tx A: localizado no 3º andar do prédio do CI, cerca de 6m de distância em linha de visada do LAR;
- Tx B: localizado no 3º andar do prédio do CI, cerca de 11.3m de distância em linha de mira do LAR;

- Tx C: localizado no 2º andar do prédio do CI, cerca de 21.2m de distância em linha diagonal até o LAR;
- Tx D: localizado no 1º andar do prédio do CI, cerca de 24.8m de distância em linha diagonal até o LAR;
- Tx E: localizado no piso térreo do prédio do CI, cerca de 29.6m de distância em linha de mira do LAR. Destacando-se aqui uma forte presença (entre os nós finais D e E) de barreiras físicas atenuadoras de sinal, tais como, paredes, linhas de concreto e canaletas metálicas;
- Tx F: localizado no piso térreo do Campus V, mais especificamente na guarita de segurança, cerca de 42m de distância do *gateway* 2 (Rx2);
- Tx G: localizado no piso térreo do prédio externo do LIA - Laboratório de Inteligência Artificial, cerca de 72m de distância até o LAR;
- Tx H: localizado no 1º andar do prédio do CTDR, cerca de 92.8m de distância até o LAR;

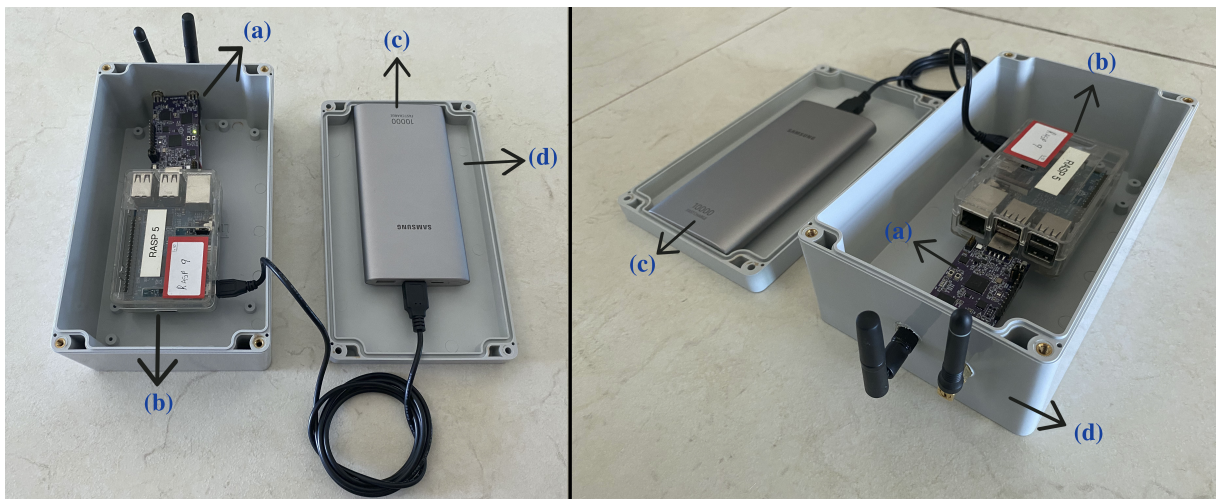


Figura 5.3: Arquitetura Interna dos Nós Finais. [Autoria Própria, 2021]

A Figura 5.3 representa a arquitetura interna de cada um dos nós finais utilizados no experimento. (a): OpenMote B: dispositivo responsável por transmitir os pacotes de dados nas modulações SUN-FSK, SUN-OQPSK e SUN-OFDM. (b): Raspberry Pi 3: computador do tipo SBC responsável pelo armazenamento dos pacotes ACK (pacotes de confirmação) enviados pelos dispositivos Rx e também pela alimentação do OpenMote B

acoplado ao sistema via porta serial. (c): Power Bank Samsung 10.000 mAh: fonte de energia responsável pela alimentação de todo o sistema contido no nó final. (d): Caixa plástica específica para prototipagem em material antichamas: utilizado para proteção do sistema como um todo, em especial os nós alocados em ambiente externo pelo fato de estarem sujeitos a precipitações pluviométricas. Vale ainda destacarmos uma adaptação feita (pequenos furos em pontos frontais) para que as antenas possam transmitir os sinais com maior liberdade de propagação, contrapondo-se a possíveis níveis de atenuação, caso durante o experimento ficassem presas fixamente no interior das caixas plásticas.



Figura 5.4: Nós Finais Alocados Internamente. [Autoria Própria, 2021]



Figura 5.5: Nó Final Externo e Gateway Receptor. [Autoria Própria, 2021]

As Figuras 5.4 e 5.5 ilustram, respectivamente, dois nós finais já nos pontos definitivos, transmitindo dados durante o experimento realizado, um nó final externo e um dispositivo *gateway* contendo a estrutura montada para a recepção dos pacotes enviados. De acordo com a imagem observa-se um *gateway* representado por um computador do

tipo SBC Raspberry Pi 4, conectado a três placas OpenMote B, cada uma delas recebendo pacotes em uma modulação específica. Sendo, Rx0: modulação SUN-FSK, Rx1: modulação SUN-OQPSK e Rx2: modulação SUN-OFDM.

Conforme citado nos subtópicos anteriores, a alimentação de todo o sistema concebido nos nós finais foi fornecida por *Power Banks* Samsung com capacidade nominal de 10.000 mAh. Objetivando-se que o experimento fosse executado pelo máximo período de tempo possível, foram aplicadas algumas estratégias para contenção/melhor eficiência energética nas Raspberry Pi. Algumas dessas estratégias foram:

- Desabilitar portas HDMI e P2;
- Desabilitar placas Wi-Fi e *Bluetooth*;
- Desativar leds de *Power* e Atividade;
- Desativar um dos controladores USB, tendo em vista que apenas um destes foi necessário durante a execução do experimento.

Tais medidas foram de extrema relevância para o aumento da vida útil da bateria portátil durante a condução do experimento.

Com a aplicação destas estratégias e tomando-se por base o log de registro gerado, os 8 nós finais apresentaram, em média, cerca de 22/23h de operação contínua. Ou seja, durante este intervalo de tempo, a rede permaneceu se comunicando e trocando informações de transmissão e confirmação, por meio dos pacotes ACK enviados.

Durante a execução do experimento, as mais variadas condições climáticas e meteorológicas foram observadas, desde a presença de precipitações pluviométricas, forte ventania que chegou a provocar leves oscilações na angulação das antenas, altos níveis de umidade e incidência direta de sol. Tais fatos serão melhor detalhados no subseção seguinte.

5.4 Análise dos Resultados

Esta subseção descreve um resumo e análise das métricas observadas durante a condução do experimento. Para tanto, cinco eixos principais foram pré-definidos para melhor organização dos dados coletados:

- Análise de PDR e RSSI geral por nó em camada física;
- Análise de PDR na camada de aplicação;
- Análise na camada de aplicação por meio de combinação de *gateways*;
- Análise de PDR em função do tempo e condições meteorológicas;
- Monitor de rede e comunicação via MQTT.

5.4.1 Análise em Camada Física

Inicialmente foi feita uma análise em camada física, avaliando-se o PDR geral por nó. Esta análise consiste em investigar a quantidade total de pacotes efetivamente enviados em cada modulação e a quantidade total de pacotes recebidos. A partir do processo de coleta dos dados adquiridos durante o experimento, inicialmente foi realizado um processo de preparação e limpeza dos dados a fim de garantir a completude e integridade das informações. Deste modo, foram gerados gráficos de PDR para cada *gateway* e para cada nó, sendo o eixo X responsável por exibir o identificador do nó e o eixo Y ilustrando o percentual de PDR médio em cada modulação analisada, de acordo com o que podemos observar nas Figuras 5.6 5.7 e 5.8

Baseado nestes gráficos, podemos observar que a comunicação na rede, via modulação SUN-FSK no *Gateway* 1 (Rx1) apresentou uma taxa de PDR mais equilibrada em relação as modulações SUN-OQPSK e SUN-OFDM igualmente analisadas. Já em relação ao *Gateway* 2 (Rx2), quando é utilizada a modulação SUN-OFDM, o fator distância e presença de barreiras físicas são determinantes quanto a qualidade na comunicação. Neste segundo ponto ainda vale destacar que a modulação SUN-OQPSK foi a única que

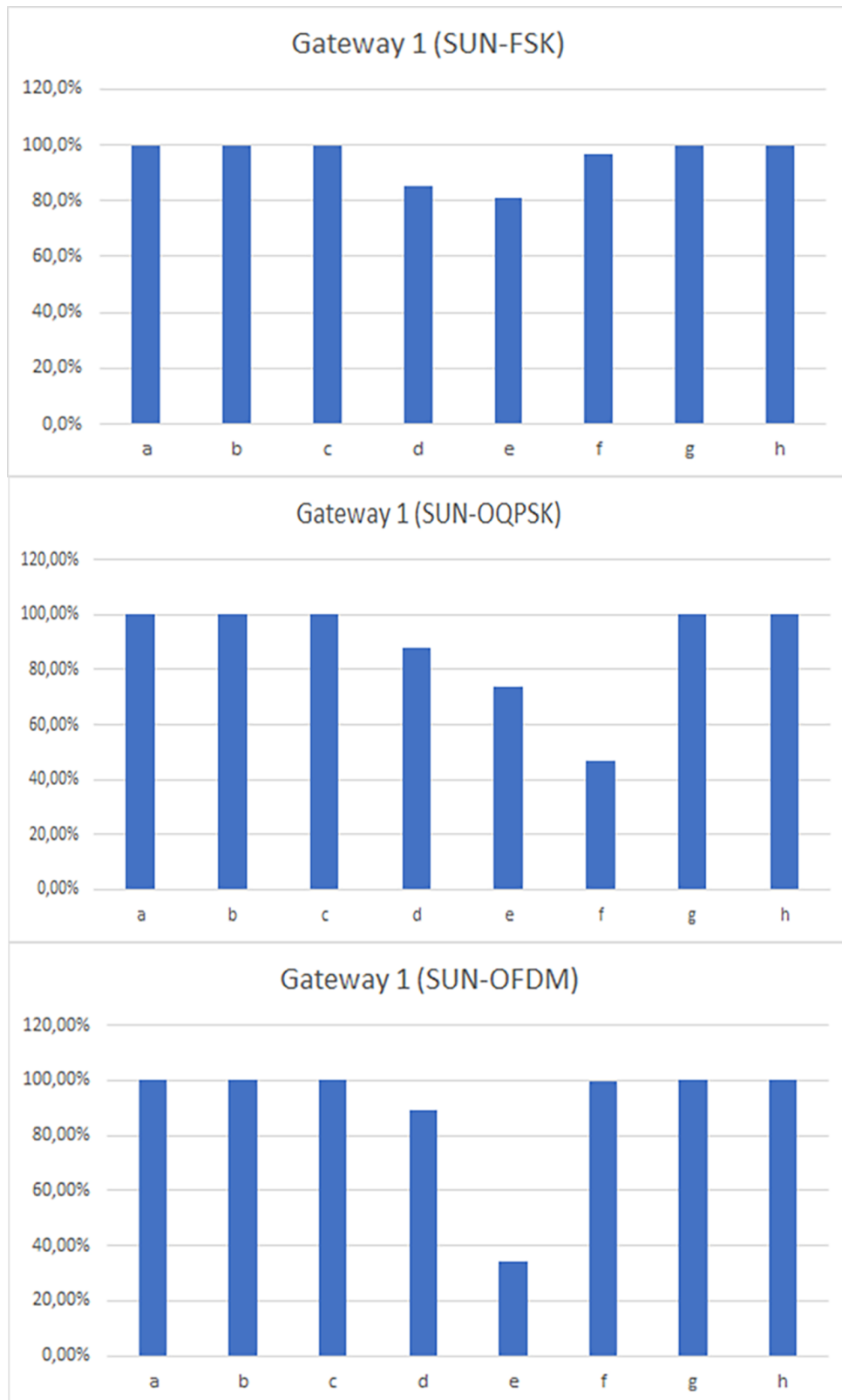


Figura 5.6: Análise em Camada Física no Gateway 1. [Autoria Própria, 2021]

conseguiu receber, em camada física, pacotes vindos de todos os nós finais, independente da distância geográfica entre os pontos. Na mesma Figura 5.8 observa-se a ausência no

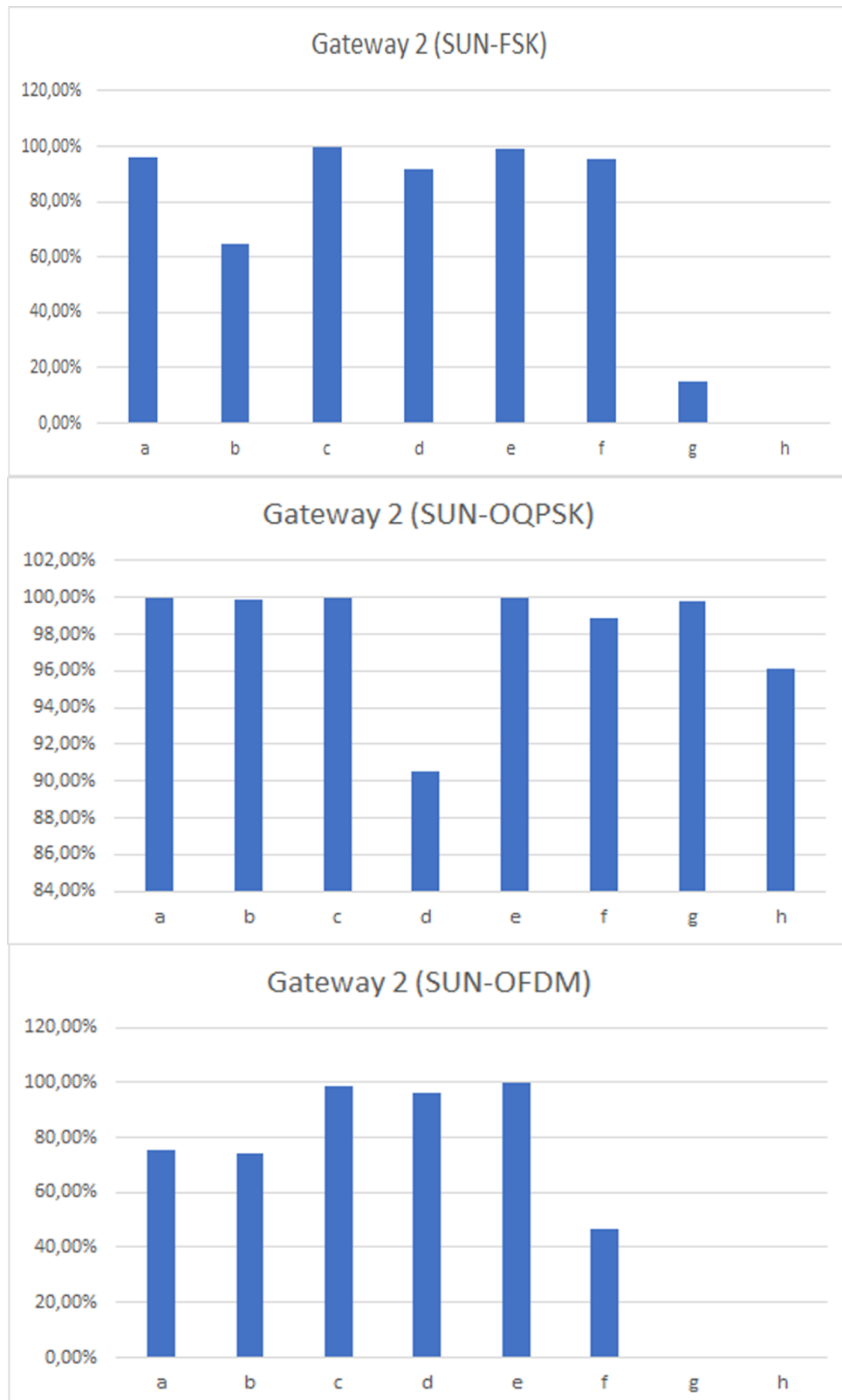


Figura 5.7: Análise em Camada Física no Gateway 2. [Autoria Própria, 2021]

gráfico CompTx (Compilado Tx) dos nós finais TxA, TxB e TxC. Esta ausência deve-se ao fato de que durante a condução do experimento, por algum motivo, até então desco-

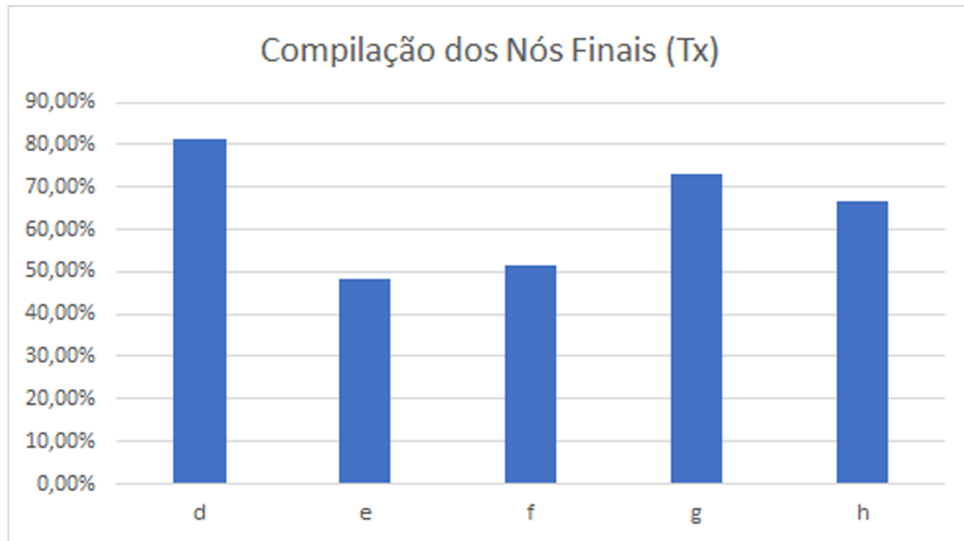


Figura 5.8: Compilação dos Nós Finais em Camada Física. [Autoria Própria, 2021]

nhcido, estes nós finais não capturaram os pacotes ACK. Deste modo, o *script* Python utilizado para captura dos dados, acabou gerando entradas vazias, quando da calota de dados, o que impossibilitou a adesão destes ao gráfico gerado.

5.4.2 Análise em Camada de Aplicação

Conforme citado anteriormente, também foi realizada uma análise a nível de camada de aplicação. Ou seja, considerando-se que um mesmo pacote é transmitido três vezes por modulação, por nó, para a perspectiva da aplicação em si, basta que este chegue apenas uma vez para ser contabilizado como recebido neste segundo cenário. Nesse caso, foram contados a quantidade de novos pacotes gerados e a quantidade de pacotes que chegaram, sendo indiferente, neste ponto em específico, ter chegado uma ou três vezes. Assim como na primeira análise, também foi feita para cada modulação em separado, considerando as três tentativas de transmissão. Deste modo, foram gerados gráficos de PDR para cada *gateway* e para cada nó, sendo o eixo X responsável por exibir o identificador do nó e o eixo Y ilustrando o percentual de PDR médio em cada modulação analisada, de acordo com o que podemos observar nas Figuras 5.9 5.10 5.11.

De acordo com os gráficos, pode-se observar uma representação similar a feita anteriormente. No entanto, quando avaliamos em perspectiva de camada de aplicação,

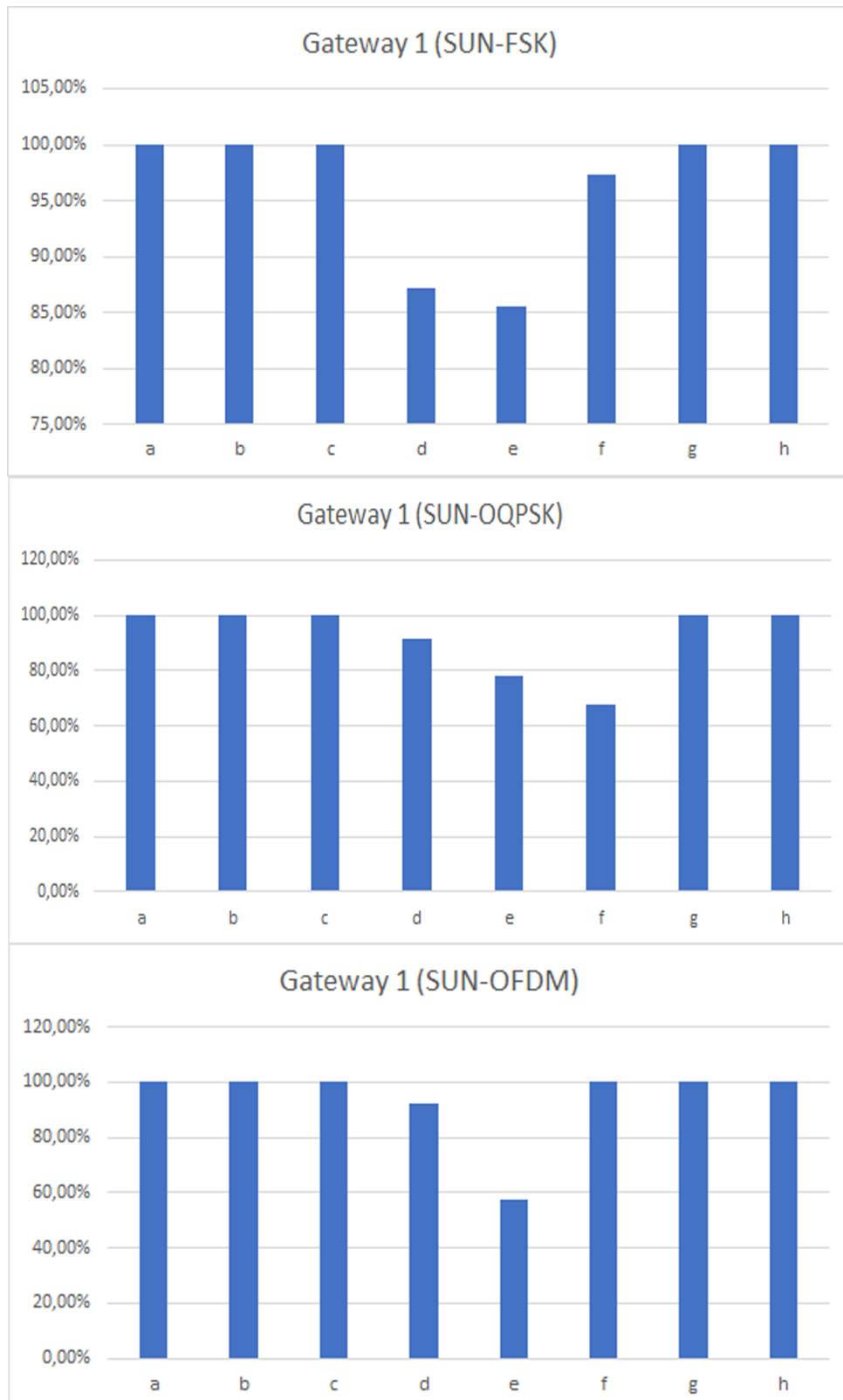


Figura 5.9: Análise em Camada de Aplicação no Gateway 1. [Autoria Própria, 2021]

podemos observar que a taxa de PDR, de um modo geral, apresenta uma tendência de crescimento positivo nos picos dos gráficos, em especial no Rx2. Fator este fruto de um

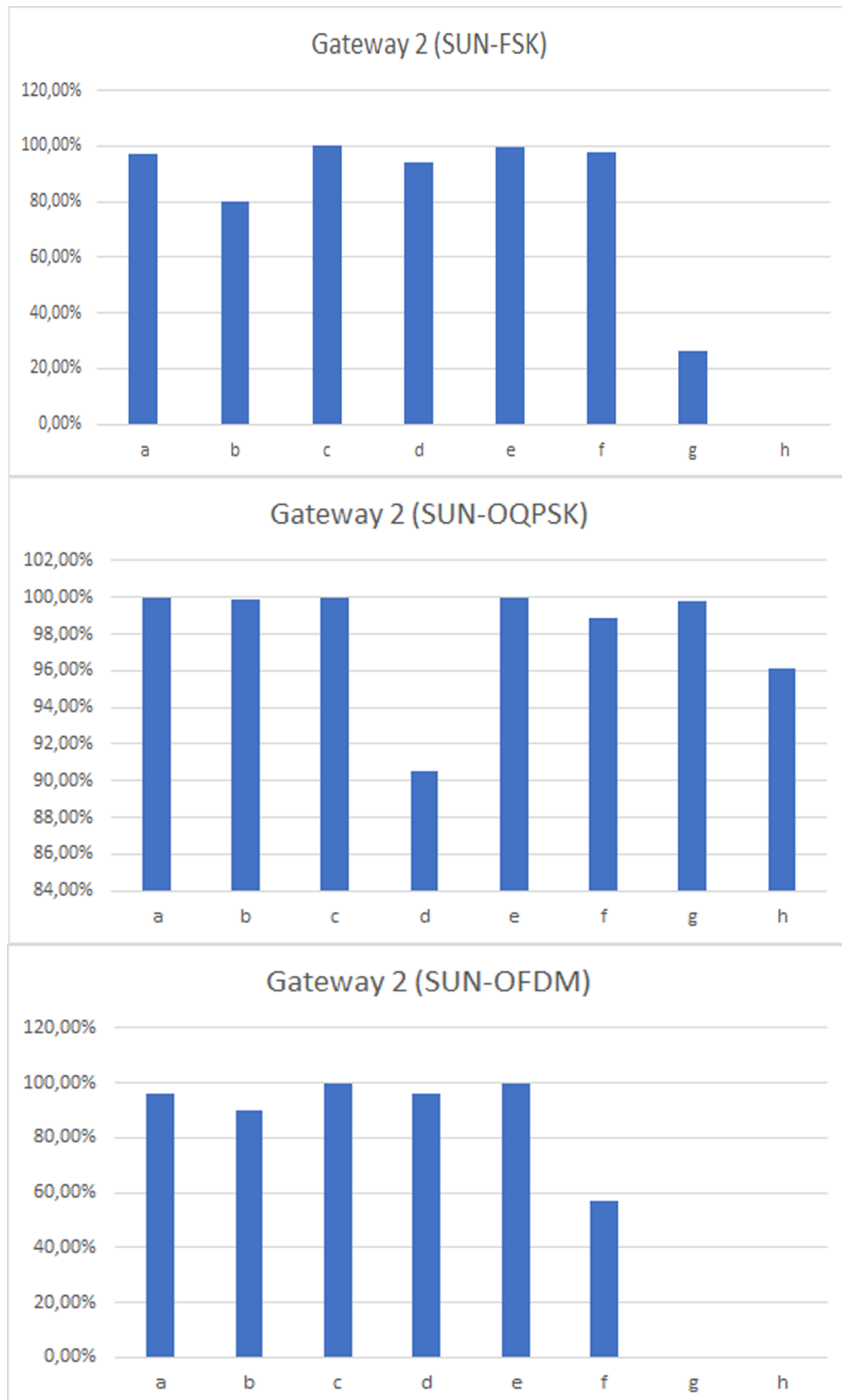


Figura 5.10: Análise em Camada de Aplicação no Gateway 2. [Autoria Própria, 2021]

processo que investiga a influência da diversidade de modulação para transmissão com o objetivo de avaliar a qualidade da comunicação na rede.

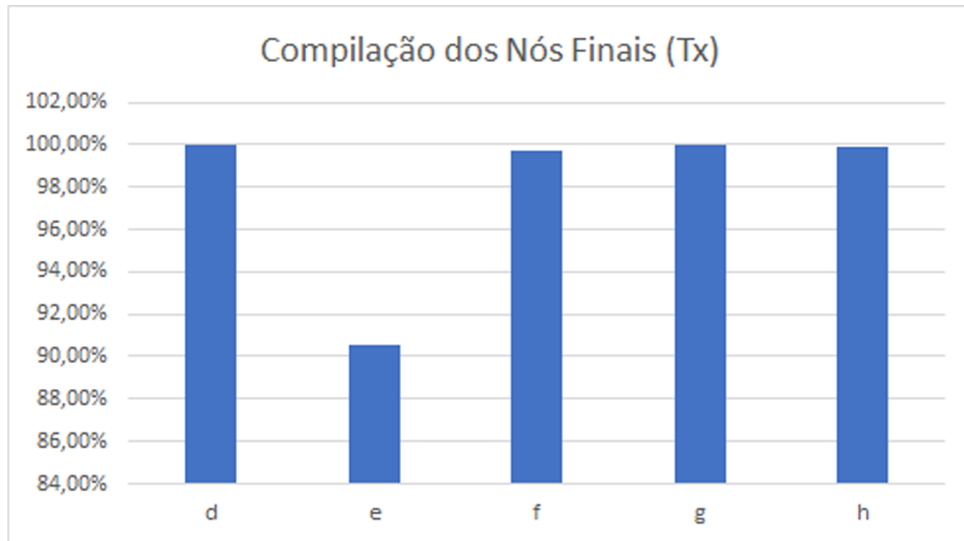


Figura 5.11: Compilação dos Nós Finais em Camada de Aplicação. [Autoria Própria, 2021]

5.4.3 Análise em Camada de Aplicação com Gateways Combinados

Posteriormente foi feita uma análise similar a do ponto anterior, no entanto, agora combinando diferentes *gateways*, conforme ilustra a Figura 5.12. Deste modo, para cada novo pacote transmitido, foi verificado se ao menos uma cópia chegou em um dos dois *gateways*. Por exemplo: se todos os pacotes transmitidos, em ao menos uma cópia chegar em um dos *gateways*, o PDR em camada de aplicação seria 100%.

Neste ponto, o objetivo principal do experimento foi investigar a influência direta da diversidade de modulação, igualmente em nível de camada de aplicação, no entanto, agora combinando ambos os *gateways* Rx1 e Rx2. Então foi feita uma análise individualizando as modulações e conforme pode-se observar, houve igualmente um ganho de desempenho, em especial na modulação OQPSK, a qual atingiu um PDR de 100% em praticamente todos os nós finais, a exceção do nó final TxF, que apesar de não estar tão distante de ambos os *gateways*, é um dos que mais enfrentou obstáculos físicos, tais como barreiras e paredes, o que novamente evidencia um forte impacto na avaliação de qualidade da rede.

Neste ponto, vale destacar que mesmo o TxF tendo uma taxa de entrega pior no geral, assim como o TxD, se considerarmos os dois *gateways* e o uso das três modulações,

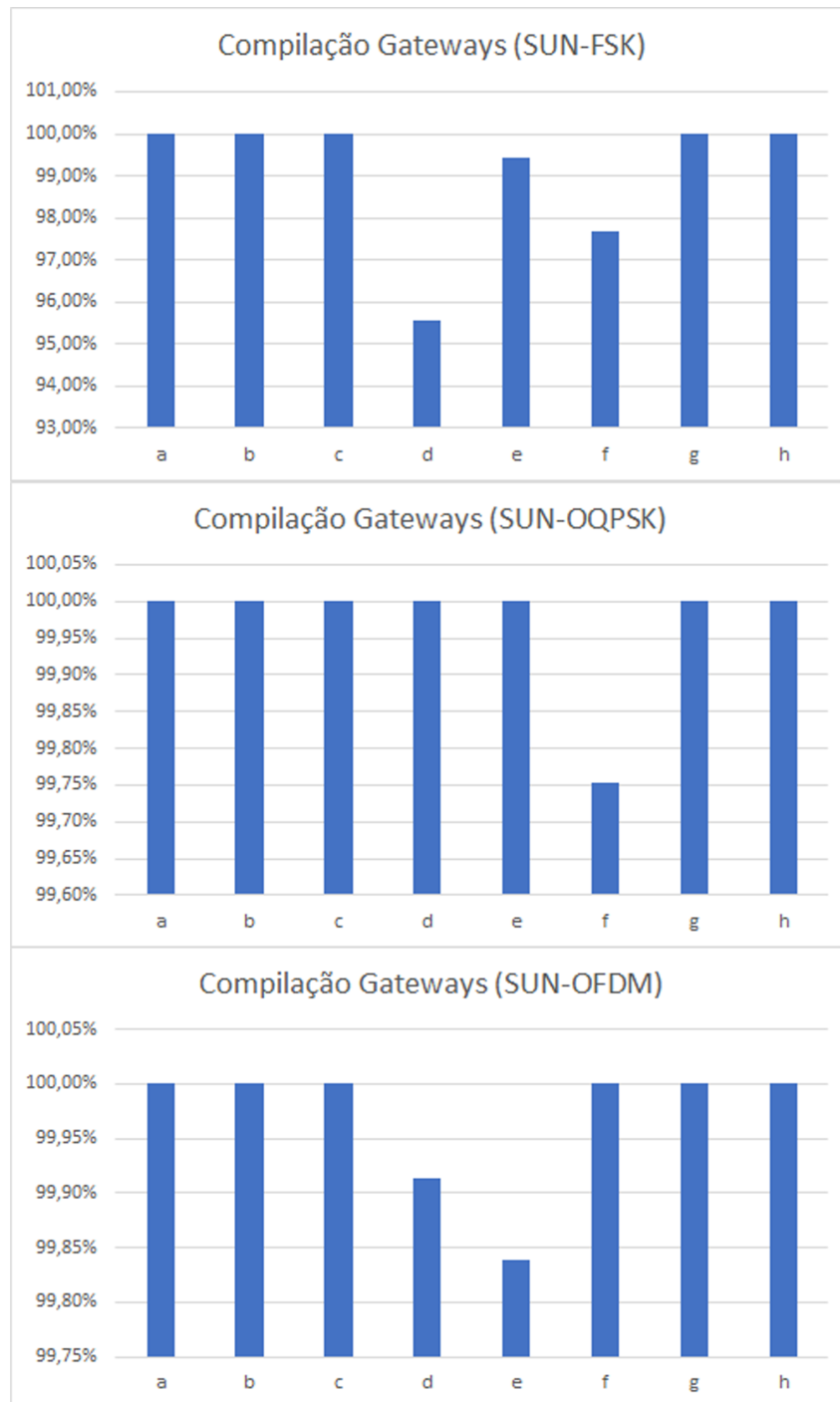


Figura 5.12: Análise com Gateways Combinados. [Autoria Própria, 2021]

ambos conseguiram 100% de entrega. Mais especificamente, o nó TxD apenas conseguiu atingir 100% de entrega ao usar SUN-OQPSK e a combinação de dados recebidos nos dois gateways. Por sua vez, o nó TxF atingiu 100% de taxa de entrega ao usar a modulação SUN-OFDM, o que reforça a importância do uso de diversidade a nível de modulação e receptor para atingir altos níveis de confiabilidade e disponibilidade para toda a rede. Isso ocorre uma vez que modulações diferentes podem apresentar diferentes níveis de qualidade

para diferentes nós, implantados em locais distintos.

5.4.4 Análise do RSSI

Além da análise voltada ao PDR, também foi investigado o Indicador de Intensidade do Sinal Recebido (RSSI), para cada *gateway* e cada nó final individualizando as suas modulações conforme pode-se observar nas Figuras 5.13 5.14 5.15.

A ideia neste ponto foi avaliarmos o quão bom um dispositivo pode ouvir, identificar e detectar sinais vindos de algum outro elemento ativo na rede. Baseado na análise conjunta com a do PDR, podemos inferir se a potência do sinal é suficiente para estabelecer uma comunicação sem fio com os demais elementos da rede. Além de que, com esta análise, foi possível validar se a adaptação feita na estrutura concebida para acomodar os nós finais foi bem sucedida de tal modo que não viesse a comprometer o sinal por meio de algum nível de atenuação, ou seja, perda de potência.

Ainda sobre esta análise do RSSI, foi calculado também o Dp (Desvio Padrão) para cada modulação em específico e em todos os *gateways*, além de uma compilação Tx_Co com os nós finais TxD até TxH:

- Rx1_0 = 10.281
- Rx1_1 = 10.691
- Rx1_2 = 08.434
- Rx2_0 = 31.377
- Rx2_1 = 16.244
- Rx2_2 = 41.789
- Tx_Co = 03.687

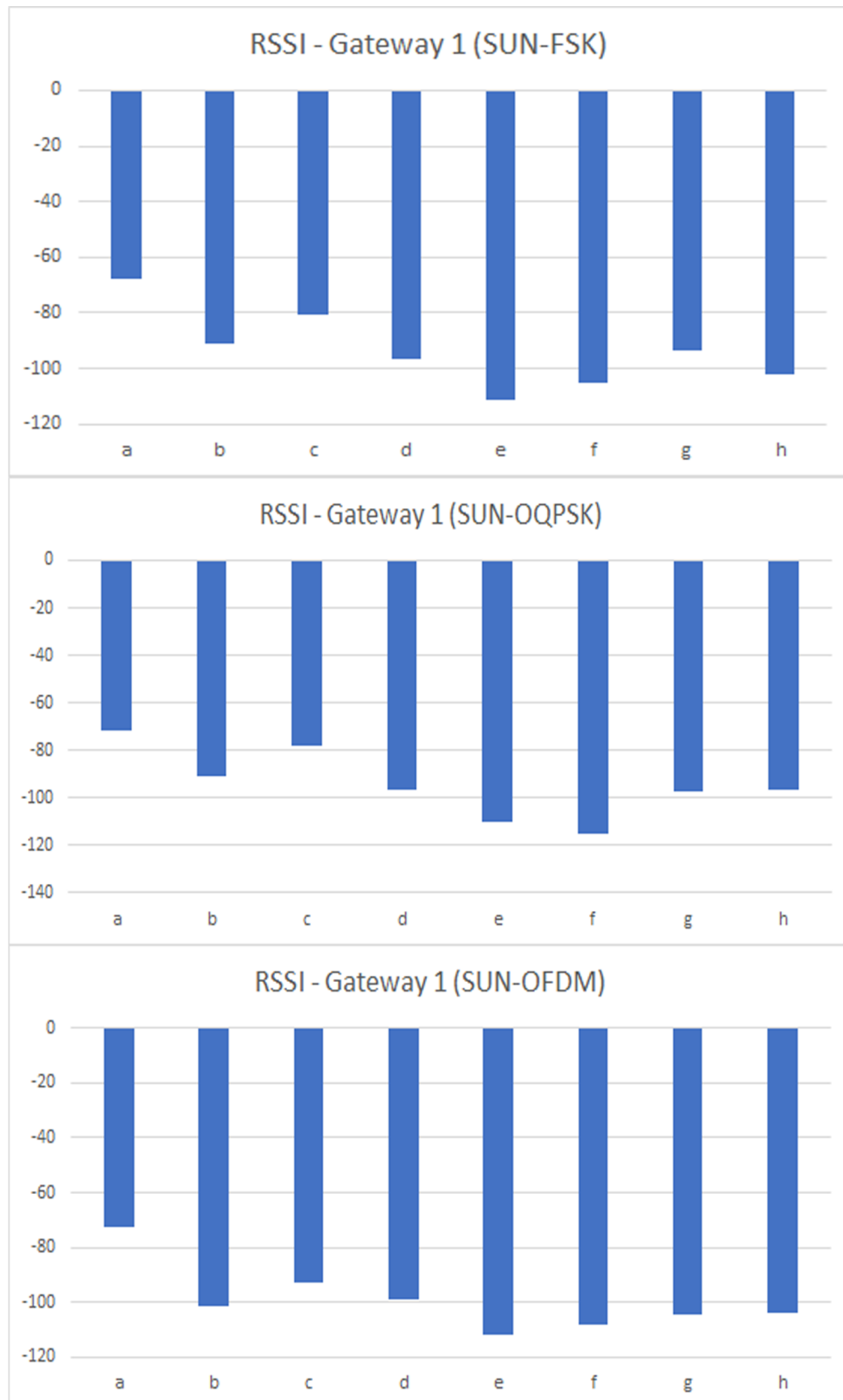


Figura 5.13: Análise do RSSI no Gateway 1. [Autoria Própria, 2021]

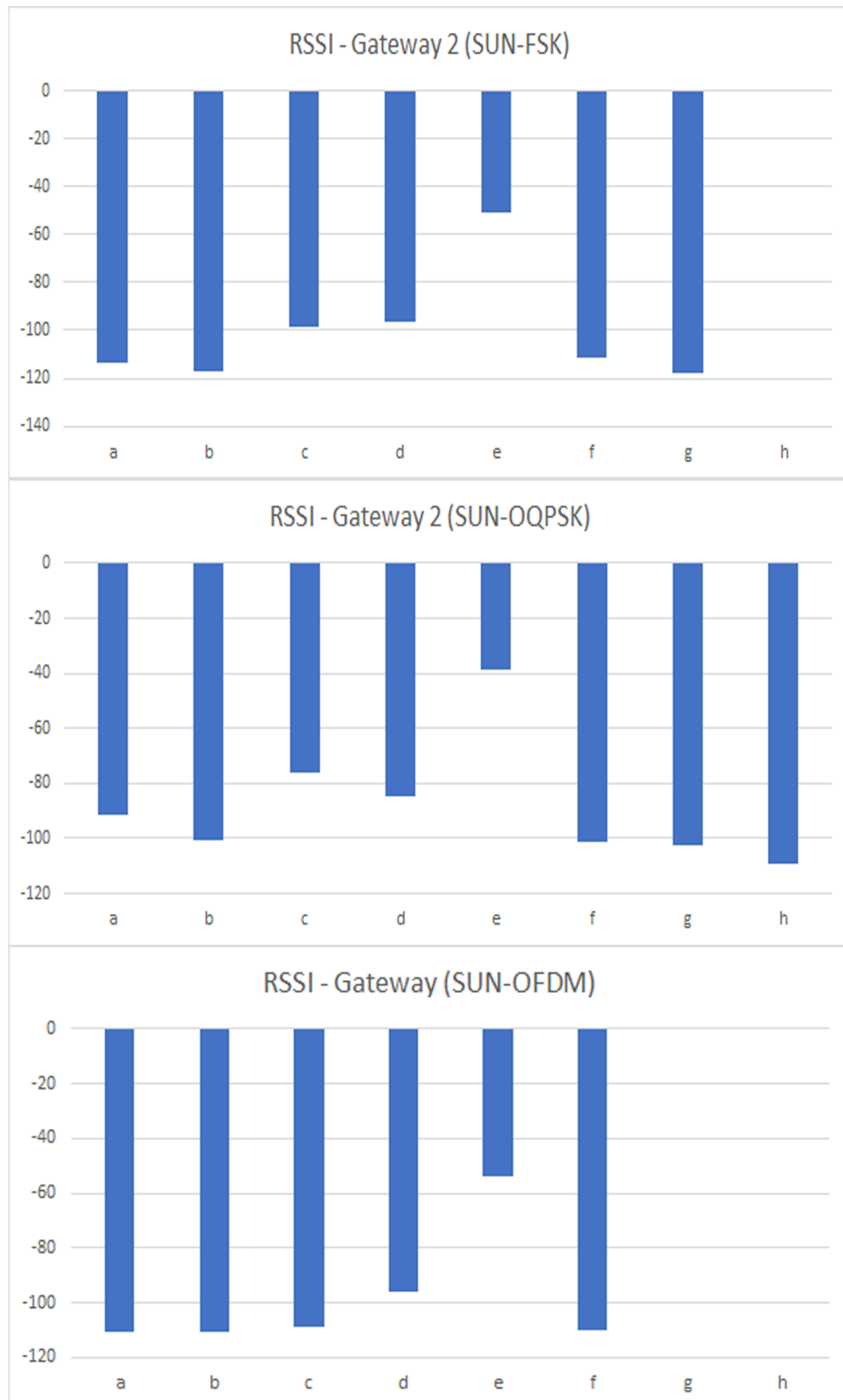


Figura 5.14: Análise do RSSI no Gateway 2. [Autoria Própria, 2021]

5.4.5 Análise de PDR em Função do Tempo e Condições Meteorológicas

A análise seguinte diz respeito ao horário (data e hora) dos pacotes recebidos e das condições físico/meteorológicas do ambiente estudado. Neste ponto, a ideia é investigar

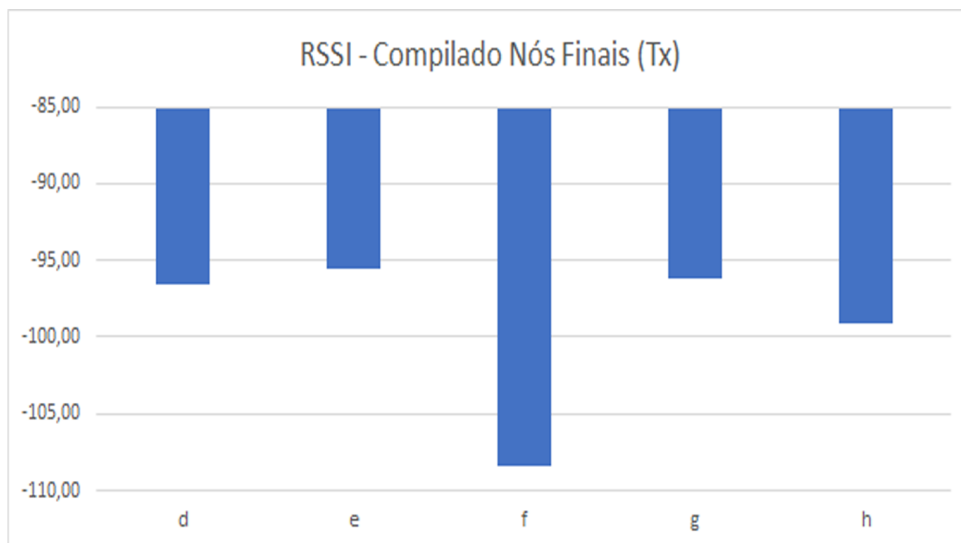


Figura 5.15: Análise do RSSI Compilado. [Autoria Própria, 2021]

um possível nível de influência das condições climáticas para o cálculo do PDR. Ou seja, buscamos verificar, de acordo com o horário do experimento, se a existência de alterações climáticas (chuva, luz solar direta, tempo nublado, ventania) de alguma forma interferem na qualidade da comunicação. Conforme citado anteriormente, o experimento foi realizado durante um período médio de 22h/23h, o qual iniciou-se no fim da tarde/início da noite, com forte ventania, entrando pela madrugada com presença de grande precipitação pluviométrica, ao amanhecer com o dia ensolarado e entrando novamente à tarde do dia seguinte com tempo nublado. Ou seja, durante o período o qual o experimento foi conduzido, tivemos condições climáticas adversas e, baseado no histórico dos eventos gerados, foi construído um gráfico conforme ilustra a Figura 5.16.

Baseado nesta análise que representa o PDR em função do horário, conseguimos identificar uma possível interferência baseada nas condições climáticas ou interferência de agentes externos. Como já era de se esperar, os nós finais que foram alocados dentro de ambiente *indoor* não foram afetados por estas alterações climáticas. No entanto, quando isolamos apenas os nós alocados externamente, conseguimos observar uma discreta diminuição na taxa efetiva de entrega de pacotes em camada física. Podemos destacar também que esta análise, em função do horário, nos permitiu estimar com boa precisão, o horário o qual os nós finais deixaram de enviar pacotes pela rede, fator este, muito provavelmente provocado pela descarga total dos *power banks* utilizados para alimentação do sistema.

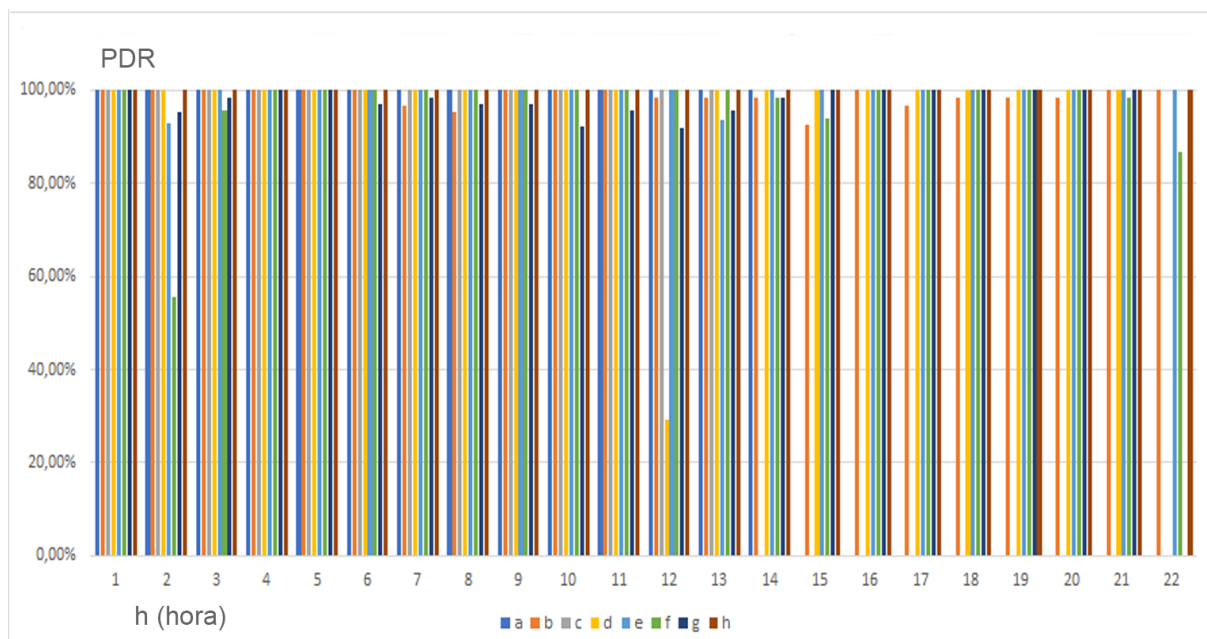


Figura 5.16: Dados Recebidos em Função do Horário. [Autoria Própria, 2021]

De modo geral, nos testes realizados percebeu-se que na maioria das situações, a modulação OQPSK obteve os percentuais de PDR mais elevados, com exceção da análise em camada de aplicação com os *gateways* combinados, fato este que pode ser considerado como um indicativo que, para um ambiente de um campus inteligente, possivelmente esta modulação seja a mais adequada, no entanto, não descartando a importância da diversidade de modulação como estratégia de aumento de confiabilidade na comunicação em uma RSSF.

5.4.6 Monitor de Rede e Comunicação via MQTT

Por fim, o ponto seguinte representa mais duas contribuições desta dissertação, as quais mostram, respectivamente, a representação da ferramenta de gerenciamento / monitoramento de rede *RAW-Manager* e uma simples aplicação MQTT *Mobile* responsável pelo consumo de dados gerados pelos nós finais da rede. Vale destacar que a aplicação *mobile* não foi desenvolvida pelo autor deste estudo, porém, foi utilizada como estratégia para validar a comunicação no trecho que utiliza MQTT. A ideia por trás da comunicação bidirecional é, de fato, promover um modelo de comunicação na rede que independente do conteúdo do pacote de dados enviado, seja possível que esses dados sejam consumidos

nos mais variados níveis de processamento, ou seja, desde um PC completo com acesso a banco de dados a praticamente qualquer dispositivo *mobile* de configuração mais simples possível para consumir as informações via MQTT. Neste último cenário teste, os dados foram transmitidos em formatos distintos na rede, desde formato de *Strings* a JSON (*JavaScript Object Notation*) e XML (*Extensible Markup Language*). Para não sobrecarregar esta subseção com muitas figuras, uma melhor representação de imagens referentes ao monitor de rede e ao consumo de dados via MQTT por uma aplicação *mobile*, são encontradas no Apêndice deste trabalho.

6 Conclusão

6.1 Considerações Finais

No experimento realizado foi possível avaliar e caracterizar o impacto e o desempenho da diversidade de modulação em camada física com o objetivo de compreender como tais pontos influenciam a qualidade e confiabilidade na comunicação de uma RSSF projetada em ambiente interno e ao ar livre. Baseado no estudo conduzido, foram exibidas algumas características relevantes dos canais sem fio descritos. Os dados gerados durante o experimento estão disponibilizados em uma plataforma aberta e de acesso público, no seguinte endereço: https://github.com/GComPI-IFPB/raw_manager/tree/master/data_experiment

Este trabalho teve o objetivo de propor e implementar uma arquitetura de rede sem fio para o monitoramento de aplicações de campus inteligente. Uma vez concebido o projeto da arquitetura, a mesma foi validada por meio de experimento em campo com uma posterior análise dos impactos provocados pela diversidade de modulação no meio físico, contribuindo também com a construção de um monitor de rede capaz de analisar, em tempo real, a ocorrência de eventos diversos relacionados ao tráfego da rede. Nos domínios investigados, o conceito de explorar diversidade de modulação mostrou-se como uma estratégia viável para a construção de redes de *smart* campus com topologia em estrela. Os experimentos sugeriram também que a exploração dos protocolos CoAP e MQTT configuram-se como boas opções para comunicação no domínio de *smart* campus, por meio da sua total integração com a arquitetura proposta e implementada neste trabalho, constituindo assim uma possível estratégia para aumento de confiabilidade em uma rede WI-SUN/802.15.4g.

6.2 Publicação Durante o Mestrado

Durante a construção deste estudo, as pesquisas até então executadas geraram o aceite de um artigo científico apresentado em Congresso Internacional em fevereiro de

2021, o qual inicialmente seria realizado em Viena, Áustria e que por questões de saúde pública, findou ocorrendo de forma remota, via videoconferência, no mesmo período acima citado.

Venceslau, F.; Gomes, R. and Fonseca, I. (2021). System for Supporting Implementation and Monitoring of Smart Campus Applications based on IoT Protocols. In Proceedings of the 10th International Conference on Sensor Networks, ISBN 978-989-758-489-3, ISSN 2184-4380, pages 113-119. DOI: 10.5220/0010346201130119.

6.3 Trabalhos Futuros e Dificuldades Encontradas

Em relação ao desenvolvimento de trabalhos futuros, este estudo visa o aprimoramento de alguns pontos executados no experimento, os quais são melhor detalhados a seguir:

- **Continuidade da análise com pacotes ACK:** esses pacotes de confirmação podem ser melhor explorados por meio da análise do comportamento em cenários com adição de mais *Gateways*, especificamente analisando questões relacionadas a colisão entre os dados trafegados na rede;
- **Adição de sensores aos nós finais:** por meio da medição de grandezas reais, pretende-se analisar a confiabilidade na rede adicionando variações no comprimento da carga útil do pacote, ao carregar dados por meio de sensores diversos, tais como: temperatura, pressão e luminosidade;
- **Reproduzir o experimento com uma duração maior:** a condução do estudo realizado na produção desta dissertação, limitou-se a vida útil da carga de energia dos *power banks* acoplados aos nós finais. Para tanto, pretende-se, em um momento futuro, utilizar redundância nessas baterias para que possamos investigar em um cenário com maior margem temporal;

Sobre as principais dificuldades encontradas na condução deste estudo, podemos destacar, primordialmente, fatores relacionados a pandemia do COVID-19 que vem asso-

lando o planeta desde janeiro/2020. Ao longo de vários meses, por restrições de isolamento social por parte das autoridades sanitárias, o acesso aos equipamentos, controladores, placas, entre outros, ficou restrito, obrigando a condução da maior parte deste estudo por meio de cenários emulados. Desde a concepção da arquitetura, a análise da simplificação e tempo de resposta no uso dos protocolos CoAP e MQTT, construção de simuladores, geradores de pacote e emuladores, demandaram vastas horas de trabalho, o que acarretou um certo nível de prejuízo temporal na construção deste trabalho.

Apêndice

Monitor de Rede implementado.



Figura .1: Monitor de Rede Construído. [Autoria Própria, 2021]

Aplicação mobile consumindo dados via MQTT.

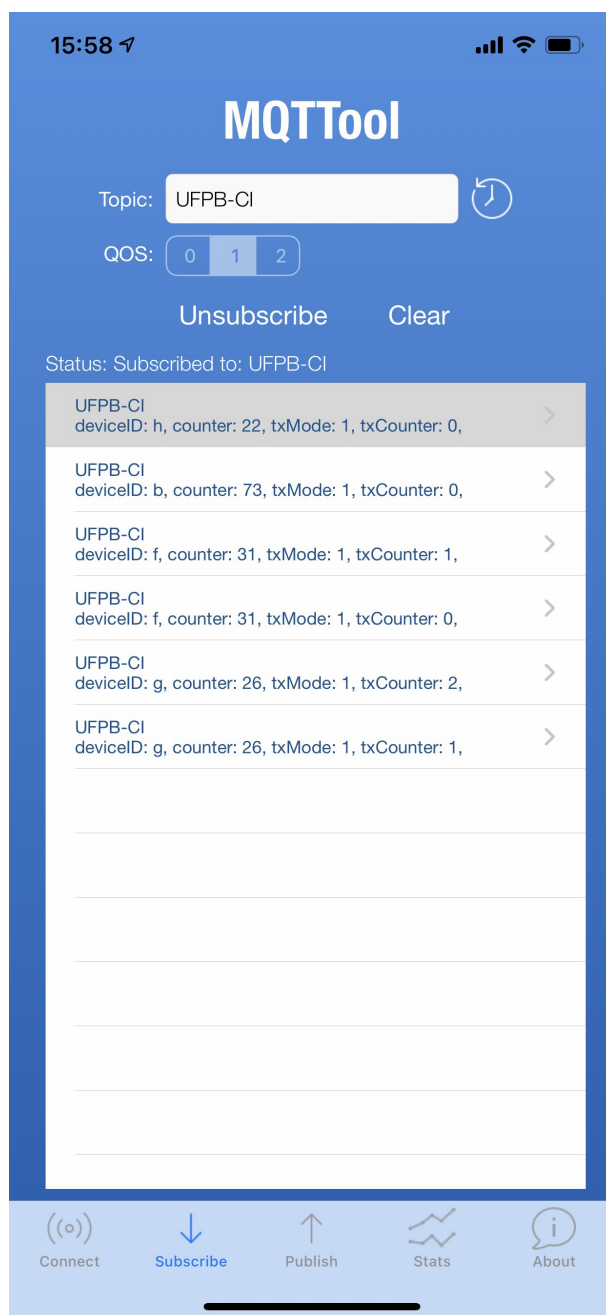


Figura .2: Consumo de dados via MQTT. [Autoria Própria, 2021]

Representação dos Nós Finais.



Figura .3: Visão Diagonal. [Autoria Própria, 2021]

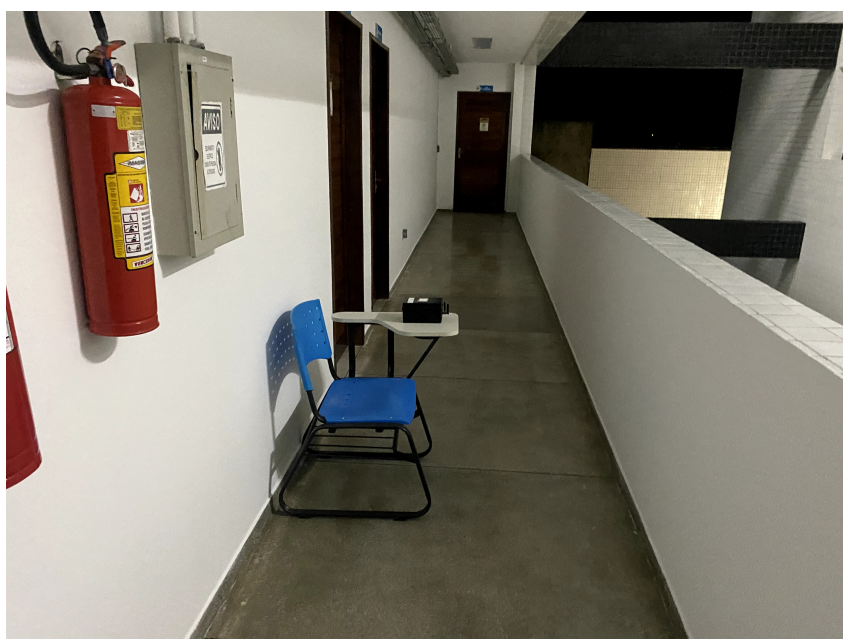


Figura .4: Visão Lateral. [Autoria Própria, 2021]

Representação dos Gateways.

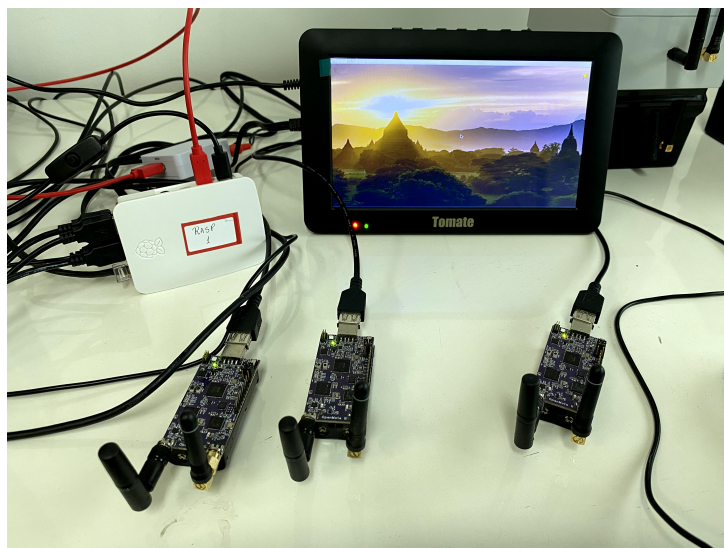


Figura .5: Gateway 01. [Autoria Própria, 2021]



Figura .6: Gateway 02. [Autoria Própria, 2021]

Referências

- [Abdul-Jabbar, 2015] Abdul-Jabbar, Y. A. (2015). *An implementation of peak to average power ratio reduction for multicarrier system (orthogonal frequency division multiple-xing)*. PhD thesis, Universiti Tun Hussein Onn Malaysia.
- [Al-Turjman and Abujubbeh, 2019] Al-Turjman, F. and Abujubbeh, M. (2019). Iot-enabled smart grid via sm: An overview. *Future Generation Computer Systems*, 96:579–590.
- [Bandeira, 2012] Bandeira, F. d. P. M. (2012). Redes de energia elétrica inteligentes. *Nota Técnica. Consultoria Legislativa. Câmara dos Deputados*.
- [Barai et al., 2015] Barai, G. R., Krishnan, S., and Venkatesh, B. (2015). Smart metering and functionalities of smart meters in smart grid-a review. In *2015 IEEE Electrical Power and Energy Conference (EPEC)*, pages 138–145. IEEE.
- [Berouine et al., 2017] Berouine, A., Lachhab, F., Malek, Y. N., Bakhouya, M., and Ouladsine, R. (2017). A smart metering platform using big data and iot technologies. In *2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech)*, pages 1–6. IEEE.
- [Bezerra Silva, 2021] Bezerra Silva, F. F. (2021). Impacto das modulações ieee 802.15.4g na qualidade de comunicação em ambiente de smart building. *IFPB*.
- [Bilic and Ozdemir, 2017] Bilic, H. G. and Ozdemir, S. (2017). The influence of the carrier frequency on the applications and the rfid general system characteristics. In *2nd International Mediterranean Science and Engineering Congress (IMSEC), Adana, Turkey*.
- [Chen et al., 2014] Chen, D., Nixon, M., Han, S., Mok, A. K., and Zhu, X. (2014). Wireless hart and ieee 802.15. 4e. In *2014 IEEE International conference on industrial technology (ICIT)*, pages 760–765. IEEE.
- [Cruz G. and Lean, 2017] Cruz G., G. and Lean, T. (2017). Séries temporais e o influxdb.
- [Dallagasperina, 2017] Dallagasperina, R. W. (2017). Uma plataforma de comunicação universal utilizando protocolo zigbee. *Embrapa Trigo-Tese/dissertação (ALICE)*.

- [de Souza et al., 2013] de Souza, S. C., Pereira Filho, J. G., and Spessimille, E. F. (2013). A rule-base approach for wsn application development in a cloud environment. *Journal of Advances in Computer Networks*, 1(4).
- [Del-Valle-Soto et al., 2019] Del-Valle-Soto, C., Valdivia, L. J., Velázquez, R., Rizo-Dominguez, L., and López-Pimentel, J.-C. (2019). Smart campus: an experimental performance comparison of collaborative and cooperative schemes for wireless sensor network. *Energies*, 12(16):3135.
- [Ejaz et al., 2017] Ejaz, W., Naeem, M., Shahid, A., Anpalagan, A., and Jo, M. (2017). Efficient energy management for the internet of things in smart cities. *IEEE Communications Magazine*, 55(1):84–91.
- [Ferreira et al., 2021] Ferreira, F., Tuset-Peiró, P., and Vilajosana, X. (2021). gcompifpb/openmote-fw. <https://github.com/GComPI-IFPB/openmote-fw>.
- [Ferret, 2013] Ferret, E. L. F. (2013). A iee 802.15. 4 como plataforma de comunicação de dados. *Revista Ilha Digital*, 4:97–105.
- [Gutierrez et al., 2001] Gutierrez, J. A., Naeve, M., Callaway, E., Bourgeois, M., Mitter, V., and Heile, B. (2001). Ieee 802.15. 4: a developing standard for low-power low-cost wireless personal area networks. *IEEE network*, 15(5):12–19.
- [Hanes et al., 2017] Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., and Henry, J. (2017). *IoT fundamentals: Networking technologies, protocols, and use cases for the internet of things*. Cisco Press.
- [Hunkeler et al., 2008] Hunkeler, U., Truong, H. L., and Stanford-Clark, A. (2008). Mqtt— a publish/subscribe protocol for wireless sensor networks. In *2008 3rd International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE’08)*, pages 791–798. IEEE.
- [Inc., 2021] Inc., R. P. (2021). Raspberry pi 4 model b. <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>.
- [Intelligence, 2018] Intelligence, A. B. (2018). More than 30 billion devices will wirelessly connect to the internet of everything in 2020. *ABI research news*, 9.

- [Mainwaring et al., 2002] Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., and Anderson, J. (2002). Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97.
- [Marfievici et al., 2017] Marfievici, R., Corbalán, P., Rojas, D., McGibney, A., Rea, S., and Pesch, D. (2017). Tales from the c130 horror room: A wireless sensor network story in a data center. In *Proceedings of the first ACM international workshop on the engineering of reliable, robust, and secure embedded wireless sensing systems*, pages 24–31.
- [Metering et al., 2017] Metering, A. S., Visalatchi, S., and Sandeep, K. K. (2017). Smart energy metering and power theft control using arduino & gsm. In *2017 2nd international conference for convergence in technology (I2CT)*, pages 858–961. IEEE.
- [Mochizuki et al., 2016] Mochizuki, K., Obata, K., Mizutani, K., and Harada, H. (2016). Development and field experiment of wide area wi-sun system based on ieee 802.15. 4g. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 76–81. IEEE.
- [Munoz et al., 2018] Munoz, J., Chang, T., Vilajosana, X., and Watteyne, T. (2018). Evaluation of ieee802.15.4g for environmental observations. *Sensors*, 18:3468.
- [Muñoz et al., 2018] Muñoz, J., Chang, T., Vilajosana, X., and Watteyne, T. (2018). Evaluation of ieee802.15.4g for environmental observations. *Sensors*, 18(10).
- [Naqvi et al., 2017] Naqvi, S. N. Z., Yfantidou, S., and Zimányi, E. (2017). Time series databases and influxdb. *Studienarbeit, Université Libre de Bruxelles*.
- [Nazario et al., 2019] Nazario, M. D., Fadul, F., and Silva, A. (2019). The constrained application protocol (coap). *Jun-2014 Available online. <http://https://www.gta.ufrj.br/ensino/eel878/redes1-2019-1/vf/coap/>*.
- [Neves, 2019] Neves, T. M. G. (2019). *Desenvolvimento de aplicações para monitorização de consumos de energia em instalações elétricas*. PhD thesis, Inst. Superior de Coimbra.
- [Pfeifer et al., 2018] Pfeifer, E. A. et al. (2018). Uma rede de sensores sem fio em malha para monitoramento on-line do consumo de energia nos edifícios da universidade federal de santa catarina. *UFSC*.

- [Quincozes et al., 2019] Quincozes, S., Emilio, T., and Kazienko, J. (2019). Mqtt protocol: Fundamentals, tools and future directions. *IEEE Latin America Transactions*, 17(09):1439–1448.
- [Ruiz et al., 2004] Ruiz, L. B., Correia, L. H. A., Vieira, L. F. M., Macedo, D. F., Nakamura, E. F., Figueiredo, C. M., Vieira, M. A. M., Bechelane, E. H., Camara, D., Loureiro, A. A., et al. (2004). Arquiteturas para redes de sensores sem fio. *Tutorial of the simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos(SBRC)*.
- [Saraiva., 2019] Saraiva., F. M. (2019). Redundancia de transceptores baseados no padrao iee 802.15.4g por meio do controle de potencia para a medicao de grandezas em sistemas de transmissao de energia eletrica. *Biblioteca de Teses e Dissertações*.
- [Shelby et al., 2014] Shelby, Z., Hartke, K., and Bormann, C. (2014). The constrained application protocol (coap)(rfc 7252). *Jun-2014 Available online. <http://www.rfc-editor.org/info/rfc7252>*.
- [Shkurti et al., 2017] Shkurti, L., Bajrami, X., Canhasi, E., Limani, B., Krrabaj, S., and Hulaaj, A. (2017). Development of ambient environmental monitoring system through wireless sensor network (wsn) using nodemcu and “wsn monitoring”. In *2017 6th Mediterranean Conference on Embedded Computing (MECO)*, pages 1–5. IEEE.
- [Soujeri et al., 2013] Soujeri, E. A., Rajan, R., Harikrishnan, A., et al. (2013). Design of a zigbee-based rfid network for industry applications. In *Proceedings of the 2nd international conference on Security of information and networks*, pages 111–116. ACM.
- [Sum et al., 2017] Sum, C.-S., Zhou, M.-T., Kojima, F., and Harada, H. (2017). Experimental performance evaluation of multihop iee 802.15. 4/4g/4e smart utility networks in outdoor environment. *Wireless Communications and Mobile Computing*, 2017.
- [Tanenbaum, 2010] Tanenbaum, A. S. (2010). *Computer Networks 5th By Andrew S. Tanenbaum (International Economy Edition)*. Prentice Hall, Indian International Ed.
- [Tessera, 2018] Tessera, T. I. (2018). *Connection Example for Wi-SUN Module for HEMS*. Yokohama-Jp, Disponível em: <https://bit.ly/2Pvf1KP>.
- [Tuset-Peiró and Vilajosana, 2019] Tuset-Peiró, P. and Vilajosana, X. (2019). openmote/openmote-fw. <https://github.com/openmote/openmote-fw>.

- [Tuset-Peiró et al., 2020] Tuset-Peiró, P., Gomes, R. D., Thubert, P., Cuerva, E., Egusquiza, E., and Vilajosana, X. (2020). A dataset to evaluate IEEE 802.15.4g SUN for Dependable Low-Power Wireless Communications in Industrial Scenarios. *MDPI Data*, XX(XX).
- [Vejlgaard et al., 2017] Vejlgård, B., Lauridsen, M., Nguyen, H., Kovács, I. Z., Mogenssen, P., and Sorensen, M. (2017). Coverage and capacity analysis of sigfox, lora, gprs, and nb-iot. In *2017 IEEE 85th vehicular technology conference (VTC Spring)*, pages 1–5. IEEE.
- [Visconti et al., 2017] Visconti, P., Primiceri, P., Ferri, R., Pucciarelli, M., and Venere, E. (2017). An overview on state-of-art energy harvesting techniques and choice criteria: a wsn node for goods transport and storage powered by a smart solar-based eh system. *Int. J. Renew. Energy Res*, 7:1281–1295.
- [Wang and Jiang, 2016] Wang, Q. and Jiang, J. (2016). Comparative examination on architecture and protocol of industrial wireless sensor network standards. *IEEE Communications Surveys & Tutorials*, 18(3):2197–2219.
- [Wang et al., 2017] Wang, S.-Y., Chen, Y.-R., Chen, T.-Y., Chang, C.-H., Cheng, Y.-H., Hsu, C.-C., and Lin, Y.-B. (2017). Performance of lora-based iot applications on campus. In *2017 IEEE 86th Vehicular Technology Conference (VTC-Fall)*, pages 1–6. IEEE.
- [Zheng et al., 2013] Zheng, J., Gao, D. W., and Lin, L. (2013). Smart meters in smart grid: An overview. In *2013 IEEE Green Technologies Conference (GreenTech)*, pages 57–64. IEEE.