

Avaliação da Influência de Interferências na Eficiência de Transmissão de Dados do Módulo ESP8266 em Redes WiFi

Rodrigo Ferraz Souza¹, Ítalo Manzine Amaral Duarte Garófalo¹

¹ Redes Sem Fio - DEC, UFSC Araranguá

Resumo. Este relatório apresenta uma análise detalhada do desempenho do ESP8266 em diversas condições de interferência, utilizando uma metodologia de explosão combinatória. As interferências testadas incluem Wi-Fi, Bluetooth, microondas e campos eletromagnéticos (CEM), em diferentes configurações e distâncias. Utilizamos regressões lineares, análise de variância (ANOVA) e testes de Tukey para avaliar a correlação entre o tipo de interferência e o tempo de transmissão. Também realizamos análises estatísticas, incluindo OLS, para investigar o impacto de cada condição de interferência. Os resultados revelam variações significativas nos tempos de transmissão sob diferentes condições de interferência, fornecendo insights valiosos para o desenvolvimento de dispositivos IoT mais resilientes.

1 Introdução

A Internet das Coisas (IoT) é um conceito que envolve a conexão de objetos físicos à internet, permitindo que dispositivos comuniquem autonomamente sem intervenção humana. Esses dispositivos podem coletar e trocar dados, criando um ecossistema de dispositivos inteligentes e interconectados, úteis em diversas aplicações, como automação residencial e monitoramento de saúde[1].

O ESP8266, um chip de baixo custo desenvolvido pela Espressif Systems, é notável em IoT por seu baixo custo, baixo consumo de energia e capacidade de Wi-Fi. Ele pode operar de forma independente ou como adaptador para outros microcontroladores, e é programável em linguagens como Arduino e MicroPython[2][3].

Neste contexto, nosso estudo foca no ESP8266, investigando como diferentes fontes de interferência afetam sua transmissão de dados em ambientes de IoT. Utilizaremos testes automatizados e análise de dados para avaliar cenários de interferência, desde competição em canais compartilhados até a presença de obstáculos físicos e dispositivos geradores de ruído eletromagnético.

Através de testes empíricos e análise estatística, como modelos de regressão linear e análises ANOVA e Tukey, buscaremos quantificar as interferências e oferecer insights sobre como mitigar seus efeitos negativos.

Este relatório detalhará o estudo desde a concepção, metodologia, resultados obtidos, até as conclusões, estruturado da seguinte forma: Seção 2 sobre

a metodologia, Seção 3 com os resultados, Seção 4 discutindo dificuldades e soluções, e a Seção 5 refletindo sobre as implicações dos resultados para sistemas IoT resilientes.

2 Metodologia

2.1 Equipamentos Utilizados

Para este estudo, utilizamos os equipamentos descritos na **Tabela 1**

Equipamento	Qtd	Motivo
Módulos ESP8266	2	Um configurado como cliente e outro como servidor.
Celular com Bluetooth	1	Utilizado para gerar interferência Bluetooth.
Caixa de som Bluetooth	1	Empregada em conjunto com o celular para intensificar a interferência Bluetooth.
Forno Micro-ondas	1	Utilizado para gerar interferência por meio de ondas eletromagnéticas.
Furadeira Elétrica	1	Empregada para criar um campo eletromagnético, simulando interferência industrial.

Tabela 1: Equipamentos Utilizados na Pesquisa

2.2 Locais dos Testes

Os testes foram realizados em três locais distintos:

- Estacionamento da Faculdade: Para testes com mínima interferência externa.
- Terreno Baldio em Araranguá: Utilizado por razões de segurança e baixa exposição a redes Wi-Fi.
- Laboratório de Redes (Lab 120) da UFSC Araranguá: Utilizado pela conveniência, apesar das interferências.

2.3 Configuração dos Testes

Para avaliar o desempenho do ESP8266, realizamos uma explosão combinatória de condições de interferência, transmitindo dados de diferentes tamanhos em cada configuração de teste. A seguir, a tabela mostra os testes planejados, incluindo a configuração das interferências e as distâncias testadas:

Número do Teste	Interferências	Configuração das Interferências
1	Sem Interferência	-
2	WIFI	Mesmo Canal, Perto do Cliente
3	WIFI	Mesmo Canal, Perto do Servidor
4	WIFI	Canal Diferente, Perto do Servidor
5	Bluetooth	Perto do Cliente
6	WIFI + Bluetooth	Mesmo Canal Perto do Server, Bluetooth Perto do Servidor
7	WIFI + Bluetooth	Canal Diferente Server, Bluetooth Perto do Cliente
8	WIFI + Parede	Mesmo Canal Perto do Cliente, 1 parede
9	WIFI + Parede	Canal Diferente Perto do Cliente, 1 parede
10	WIFI + Microondas	Mesmo Canal Perto do Cliente, Microondas Perto do Cliente
11	WIFI + Microondas	Canal Diferente Perto do Cliente, Microondas Perto do Cliente
12	WIFI + CEM	Mesmo Canal Perto do Cliente, CEM Perto do Cliente
13	WIFI + CEM	Canal Diferente Perto do Cliente, CEM Perto do Cliente

Tabela 2: Configurações de Testes Realizados

2.3.1 Estrutura dos Dados Coletados

Os dados foram registrados em tabelas CSV com as colunas descritas na **Tabela 3**

2.4 Espectrogramas de Redes

Os espectrogramas de rede são representações gráficas essenciais para avaliar o espectro de frequência de sinais Wi-Fi, permitindo-nos visualizar a presença e a intensidade de sinais e possíveis interferências. Abaixo, apresentamos e discutimos os espectrogramas coletados durante a nossa pesquisa.

Na Figura 1, observamos um ambiente sem interferências significativas, onde a rede **ESP_SERVER** opera sem muita competição no espectro de 2.4 GHz. Este cenário serve como nosso controle de linha de base sempre que consideramos um ambiente *Sem Interferência WiFi*.

A Figura 2 ilustra a sobreposição de espectro quando o ESP8266 e a rede doméstica "Romulo" operam no mesmo canal. Nota-se um aumento significa-

Nome da Coluna no CSV	Tipo de Dado	Valores Possíveis
Tamanho	Number	[128,10240]
Distancia	Number	[0,max]
Tempo	Number	[0,600000]
Paredes	Number	[0,3]
INT_WIFI	Bool	{0,1}
INT_WIFI-Mesmo_Canal	Bool	[0,1]
INT_WIFI-Proximo	String	{"CLIENT","SER-VER"}
INT_BT	Bool	{0,1}
INT_BT-Proximo	String	{"CLIENT","SER-VER"}
INT_MICROONDAS	Bool	{0,1}
INT_MICROONDAS-Proximo	String	{"CLIENT","SER-VER"}
INT_FURA-DEIRA	Bool	{0,1}
INT_FURADEIRA-Proximo	String	{"CLIENT","SER-VER"}

Tabela 3: Estrutura dos Dados Coletados

tivo na densidade espectral, o que pode causar perdas e retransmissões devido à interferência direta.

A Figura 3 apresenta uma situação onde o ESP8266 está configurado para operar no canal 11, distinto do canal 4 usado pela rede doméstica "Romulo". Embora não esteja competindo diretamente no mesmo canal, a rede ESP8266 ainda está sujeita à interferência das atividades na rede "Romulo", especialmente devido ao tráfego aumentado provocado pelos nossos testes. Esta configuração é representativa de uma situação real onde múltiplas redes operam em proximidade, mas em canais separados, e é crítica para entender como a coexistência de redes pode afetar o desempenho de dispositivos sem fio em um ambiente típico.

Cada espectrograma fornece uma visão detalhada de como as redes Wi-Fi coexistentes podem influenciar o desempenho do nosso dispositivo, sublinhando a importância de considerar essas condições ao desenvolver sistemas de comunicação sem fio.

2.5 Coleta e Análise de Dados

Para coletar e analisar os dados foi usada uma conexão TCP usando a biblioteca *ESP8266WiFi.h* em cada um dos ESPs, um se comportando como um cliente que requisitava um "arquivo" de tamanho variável e um servidor que devolvia bytes sem significado, apenas para simular o tamanho. Diagrama exempli-

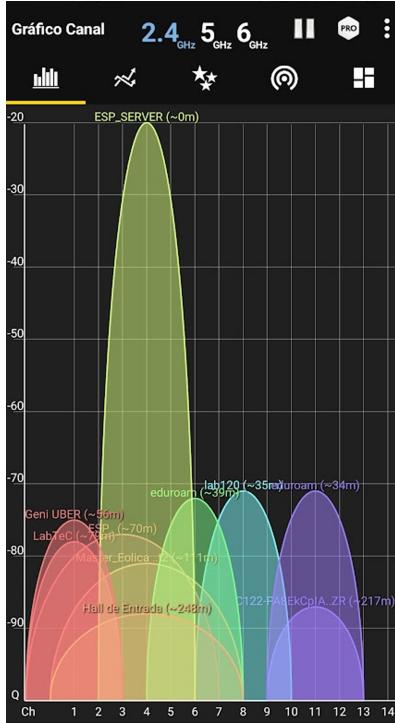


Figura 1: Espectrograma representando o ambiente sem interferência.

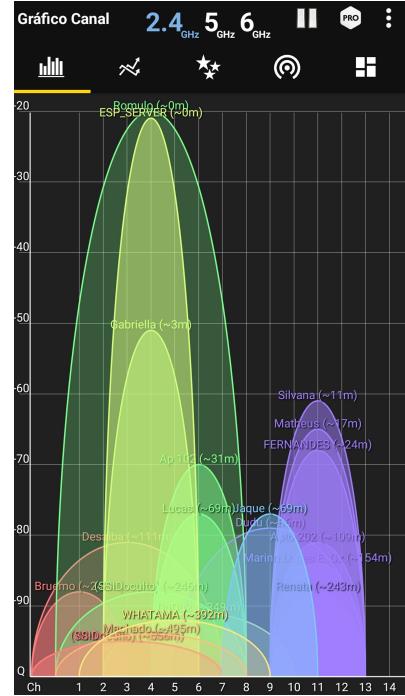


Figura 2: Espectrograma com sobreposição no mesmo canal da rede "Romulo".

ficando o processo do experimento pode ser visto na Figura 4 A coleta e análise de dados foram realizadas utilizando um conjunto de scripts Python, desenvolvidos especificamente para este projeto. Estes scripts foram programados para operar em **Python 3.11**, utilizando as bibliotecas mais recentes disponíveis em outubro de 2023, incluindo *argparse*, *pandas*, *numpy*, *serial*, *time*, *re* e *tqdm*. Eles foram criados para automatizar os testes, facilitar a comunicação serial com o *ESP8266* e registrar os resultados em arquivos CSV.

EspSerial.py: Gerenciava a comunicação serial com o *ESP8266*. O trecho do código abaixo é apresentado para ilustrar a função `run_experiment` usada para executar os testes.

```

1  """Executa o experimento e
2  retorna os tempos de transferência
3  para cada tamanho, realizando 3 envios.
4  return [int]
5  """
6
7  # Tamanhos dos arquivos
8  sizes = [i for i in range(128, 10241, 128)]
9  data = {}
10 time.sleep(2)
11 for size in tqdm(sizes):
12     times = []
13     for _ in range(3):
14         # Faz 3 envios para cada tamanho
15         # de arquivo
16         self.serial.write(f"{size}\n".encode())
17
18         # Aguarda para processamento
19         time.sleep(2)
```

```

line = self.aguardar_substring(
    "SUCCESS"
)
time_match = re.search(
    'Duration:(\d+) ms',
    line
)
if time_match:
    times.append(
        int(
            time_match.group(1)
        )
    )
else:
    print("Tempo não encontrado")
    time.sleep(2)
data[size] = times
return data
```

ExperimentArguments.py e ExperimentWrapper.py: Facilitaram a configuração e execução dos testes. main.py: Script principal que orquestrava a execução dos testes e a coleta de dados.

2.6 Códigos dos Módulos *ESP8266*

A programação dos módulos *ESP8266* foi realizada utilizando a Arduino IDE, onde foram desenvolvidos códigos específicos para as funções de cliente e servidor no contexto de comunicação de rede. No processo, o cliente inicia o envio de uma solicitação HTTP ao servidor através da rota `/file`. Esta solicitação inclui uma query que especifica o tamanho do arquivo a ser transmitido, como por exemplo, `?size=<qtd bytes>`. Ao receber a solicitação, o servidor responde

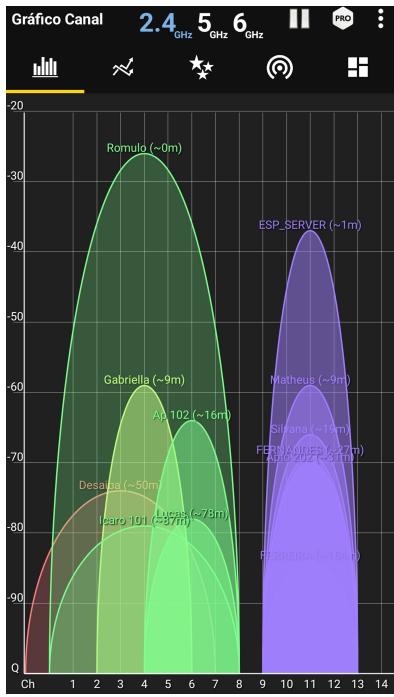


Figura 3: Espectrogrma com a rede ESP8266 operando no canal 11 e a rede domstica "Romulo"no canal 4.

enviando uma sequêcia de caracteres 'A', em que cada 'A' representa 1 byte. A transmissão é feita em blocos de 1024 caracteres para evitar sobrecarga da memória RAM do dispositivo, um ponto crítico, especialmente em situações onde o tamanho requisitado excede a capacidade de memória disponível.

Durante esse processo, o cliente também inicia a medição do tempo de transmissão. Para o desenvolvimento desses códigos, foi utilizada exclusivamente a biblioteca ESP8266WiFi.h. Um exemplo chave deste processo é o método de envio de dados do servidor, implementado na função send_data() mostrada abaixo, que gerencia eficientemente a transmissão dos dados conforme a solicitação do cliente.

```

1 void handle_buffer(
2     WiFiClient client,
3     String outputBuffer,
4     int *progress,
5     int fileSizeRequested,
6     int i)
7 {
8     // Enviar o buffer quando
9     // atingir um tamanho específico
10    if (outputBuffer.length() >= 1024)
11    {
12        client.print(outputBuffer);
13        outputBuffer = "";
14        if (100 * i / fileSizeRequested >= *progress)
15        {
16            *progress = 100 * i / fileSizeRequested;
17        }
18    }

```

```

19    }
20
21    void send_data(
22        WiFiClient client,
23        int fileSizeRequested)
24    {
25        // Bufferizando a saída para enviar
26        // vários caracteres de uma vez
27        int progress = 0;
28        if (fileSizeRequested > 0)
29        {
30            String outputBuffer = "";
31            for (
32                int i = 0;
33                i < fileSizeRequested;
34                i++)
35            {
36                // Conteúdo do arquivo
37                // (caractere 'A' repetido)
38                outputBuffer += 'A';
39                handle_buffer(
40                    client,
41                    outputBuffer,
42                    &progress,
43                    fileSizeRequested,
44                    i);
45            }
46            // Enviar o restante do buffer
47            if (outputBuffer.length() > 0)
48            {
49                client.print(outputBuffer);
50            }
51        }
52    }

```

O trecho de código abaixo mostra como este envio de pedido e recebimento de informações é feito no cliente para que seja possível calcular o tempo da transmissão:

```

1 // limpa o buffer do serial monitor
2 while (Serial.available())
3 {
4     Serial.read();
5 }
6 Serial.println("FILE SIZE (Bytes):");
7 while (!Serial.available())
8 {
9     delay(10);
10 }
11 int requestedFileSizeB = Serial.parseInt();
12 if (requestedFileSizeB == 0)
13     return;
14 if (client.connect(serverIP, serverPort))
15 {
16     + 5)unsigned long startTime = millis();
17     client.println(
18         "GET /file?size=" +
19         String(requestedFileSizeB) +
20         " HTTP/1.1");

```

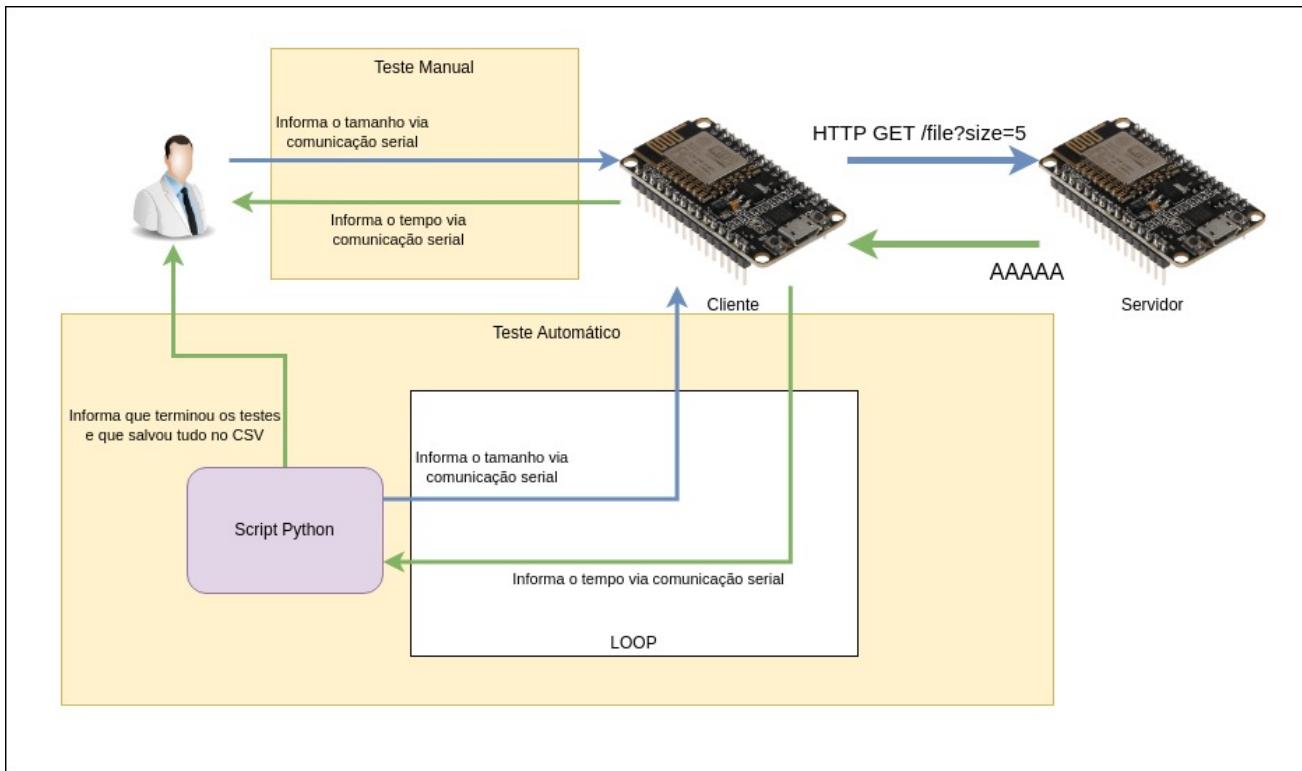


Figura 4: Diagrama do Experimento

```

22 int bytesRead = 0;
23 // Define um timeout de 600 segundos
24 long timeout = millis() + 600000;
25 bool received = false;
26 // clear buffer
27 client.flush();
28 while (
29     millis() < timeout
30     &&
31     bytesRead < requestedFileSizeB)
32 {
33     while (client.available())
34     {
35         received = true;
36         char c = client.read();
37         bytesRead++;
38     }
39 }
40
41 unsigned long endTime = millis();
42 unsigned long duration = endTime - startTime;
43
44 Serial.print("SUCCESS:");
45 Serial.print("Size:");
46 Serial.print(bytesRead);
47 Serial.print(" bytes;");
48 Serial.print("Duration:");
49 Serial.print(duration);
50 Serial.println(" ms");
51 client.stop();
52 }
53

```

Todos os códigos citados podem ser conferidos na íntegra no GitHub¹

2.7 Adaptações e Decisões de Teste

Durante os testes, tivemos que fazer ajustes significativos, especialmente nos testes envolvendo a furadeira (CEM) e o micro-ondas. Optamos por alterar o tamanho dos dados enviados de 128 bytes para 1280 bytes, com o maior tamanho sendo **9088** bytes ao invés de 10240 bytes. Essas mudanças foram motivadas pela preocupação em não danificar o ESP8266 e pela necessidade de reduzir o tempo de exposição ao ruído da furadeira, evitando problemas com vizinhos.

2.8 Análise Estatística dos Tempos de Transmissão

Para avaliar o impacto das diversas interferências no desempenho do ESP8266, realizamos uma análise estatística rigorosa dos dados de tempo de transmissão coletados.

2.8.1 Regressão Linear

Inicialmente, empregamos a técnica de regressão linear, uma abordagem estatística que permite modelar e analisar a relação entre o tamanho dos dados transmitidos (variável independente) e o tempo necessário para a conclusão de cada transmissão (variável dependente), em diferentes configurações de interferência.

¹<https://github.com/TarefasUFSC/Analise-de-Transferencia-de-dados-com-ESP8266>

2.8.2 ANOVA e Tukey

Com modelos de regressão linear estabelecidos para cada cenário, avançamos para a análise de variância (ANOVA), um método estatístico comparativo que nos ajudou a discernir se as diferenças nos tempos de transmissão observados entre os vários tipos de interferência eram de fato estatisticamente significativas. A ANOVA foi uma etapa decisiva para identificar as interferências que exerciam um impacto substancial no desempenho da rede.[4]

Depois da ANOVA, implementamos testes post hoc de Tukey, que são utilizados para realizar comparações múltiplas entre médias de grupos permitiu especificar entre quais condições de interferência essas diferenças eram estatisticamente reais, oferecendo uma compreensão detalhada da magnitude das diferenças entre as condições de interferência. Esses testes são fundamentais, pois, após o ANOVA indicar diferenças significativas, o teste de Tukey novamentes.[5][6]

2.8.3 Análise da Regressão Linear Ordinária (OLS)

A análise da Regressão Linear Ordinária (OLS) é uma estratégia de otimização usada em modelos de regressão linear para encontrar a linha que melhor se ajusta aos pontos de dados. Nesse modelo, o **coeficiente (coef)**, representando o β no modelo OLS, indica a mudança esperada na variável dependente para uma unidade de mudança na variável independente, com outras variáveis mantidas constantes [7].

O **intercepto (const)**, ou α no modelo OLS, é o valor da variável dependente quando todas as variáveis independentes são zero, representando onde a linha de regressão intercepta o eixo Y [7].

O **t-value (t)** é usado para testar a significância do coeficiente no modelo. Se o p-valor associado ao t-value for menor que um limiar (por exemplo, $= 0.05$), isso indica uma relação estatisticamente significativa entre as variáveis [8].

A relação entre coeficientes e t-values é interpretada da seguinte maneira:

- **Ambos Positivos ou Ambos Negativos:** Indica uma relação significativa.
- **Um Negativo e Outro Positivo:** Requer uma análise mais cuidadosa, podendo indicar uma relação não significativa ou complexa.

O modelo OLS é uma técnica estatística de regressão usada para analisar a relação entre variáveis dependentes e independentes. Neste estudo, aplicamos o modelo para entender como diferentes condições de interferência e distâncias afetam o tempo médio de transmissão de dados.

Esses procedimentos estatísticos, combinados, forneceram insights valiosos sobre a correlação

Tipo de Interferência	Tempo Médio de Transmissão (ms)
Sem Interferência (0m)	366.13
WiFi Diferente + 1 Parede (0m)	370.95
WiFi Diferente + Cliente (0m)	354.29
WiFi + Microondas (0m)	353.25
WiFi Diferente + Bluetooth (0m)	368.57
WiFi Diferente (0m)	373.33
WiFi Mesma Rede + 1 Parede (0m)	361.13
WiFi Mesma Rede + Cliente (0m)	350.71
WiFi Mesma Rede + Microondas (0m)	354.08
WiFi Mesma Rede + Bluetooth (0m)	363.72
WiFi Mesma Rede (0m)	366.05
Bluetooth (0m)	375.05
Sem Interferência (1m)	366.27
WiFi Diferente + Bluetooth (1m)	375.97
WiFi Mesma Rede (1m)	362.20
Sem Interferência (5m)	376.29

Tabela 4: Tempos médios de transmissão sob diferentes interferências

entre o tipo de interferência e o desempenho da rede, capacitando-nos a formular conclusões sólidas a respeito do efeito de distintos cenários de interferência sobre a eficiência de dispositivos IoT como o ESP8266. Assim, conseguimos não apenas quantificar o efeito das interferências, mas também entender o grau de impacto de cada uma delas, contribuindo para uma compreensão mais ampla e aplicável ao desenvolvimento de sistemas de comunicação sem fio resilientes.

3 Resultados

3.1 Análise da Eficiência da Transmissão de Dados

Esta análise foca na eficiência da transmissão de dados do módulo ESP8266 em diferentes cenários de interferência. Na Tabela 4 apresenta os tempos médios de transmissão sob várias condições de interferência. Estes tempos são cruciais para compreender como diferentes fontes de interferência afetam a eficiência de transmissão do módulo ESP8266.

A Tabela 4 detalhada apresenta uma visão abrangente dos efeitos de diferentes tipos de interferência na eficiência de transmissão de dados do módulo ESP8266. As distâncias (0m, 1m, 5m) indicadas junto aos tipos de interferência representam a distância entre os módulos ESP8266 durante os testes.

Distância	Tamanho dos Dados (bytes)	Tempo de Transmissão (ms)
50m	50	331
50m	1024	357
100m	50	281
100m	1024	426
160m	50	252
160m	1024	falha

Tabela 5: Desempenho da transmissão de dados em função da distância e do tamanho dos dados

Nota-se variações significativas nos tempos médios de transmissão, o que evidencia a influência das condições ambientais e da proximidade dos dispositivos na eficiácia da comunicação sem fio.

Expandimos nossos testes para avaliar a eficiência da transmissão de dados a maiores distâncias, observando o comportamento do módulo ESP8266 em seu limite operacional. Os resultados desses testes estão resumidos na Tabela 5, que destaca o desempenho do módulo em distâncias de 50m, 100m e 160m para cargas de dados de 50 bytes e 1024 bytes.

Os testes adicionais revelaram aspectos críticos da eficiência de transmissão do módulo ESP8266. Foi constatado que para cargas de dados de 1024 bytes, o alcance efetivo de transmissão não ultrapassa 100 metros. Além deste limite, o módulo falha ao tentar transmitir pacotes deste tamanho, evidenciando uma restrição quanto à quantidade de dados que podem ser enviados à medida que a distância aumenta. Por outro lado, para volumes menores de dados, especificamente pacotes de 50 bytes, o módulo manteve a eficiência de transmissão até 160 metros. Esta descoberta sublinha a influência da distância na performance da transmissão, que se torna especialmente significativa com o aumento do volume de dados.

A capacidade de transmissão de dados em longas distâncias é um fator essencial na concepção de redes IoT, e os resultados obtidos enfatizam a necessidade de adaptar o design de sistemas Wi-Fi para acomodar as limitações identificadas. As variações observadas nos tempos de transmissão também destacam a resiliência do módulo frente a diferentes tipos de interferência. Surpreendentemente, algumas condições específicas, como a presença de 'WiFi Diferente + Cliente' e 'WiFi + Microondas', mostraram-se menos impactantes do que o cenário sem interferência, o que pode indicar que determinados tipos de interferência podem não comprometer significativamente a eficiência da transmissão do módulo.

É crucial reconhecer que o desempenho do módulo ESP8266 é afetado por uma complexa interação de fatores, que vão desde o tipo e a intensidade das interferências até a configuração física do ambi-

ente em que está operando. A compreensão desses elementos é vital para otimizar a eficácia da comunicação sem fio em ambientes com diversas fontes de potencial interferência.

3.1.1 Interpretação das Regressões Lineares

A Figura 5 apresenta as regressões lineares para cada configuração de teste, oferecendo uma representação visual da relação entre o tamanho dos dados transmitidos e o tempo de transmissão. As linhas de regressão mostram a tendência do tempo de transmissão à medida que aumentamos a carga de dados, permitindo uma análise preditiva do comportamento do sistema sob variadas condições de interferência.

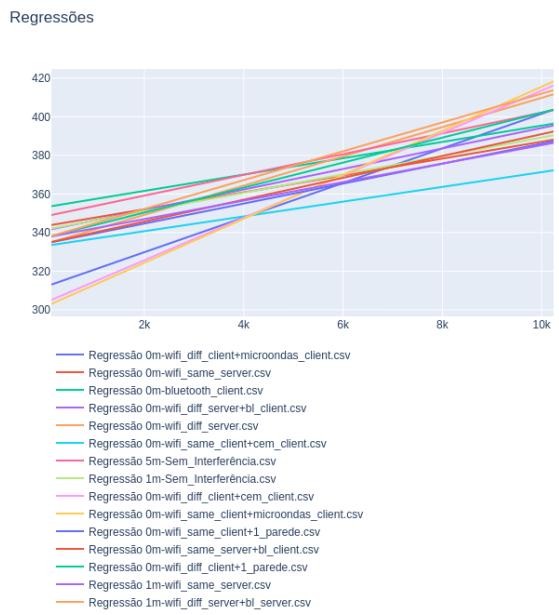


Figura 5: Regressões lineares dos tempos de transmissão para diferentes cenários de interferência.

As linhas de regressão demonstram que, para cada cenário de interferência, há um aumento linear no tempo de transmissão com o incremento no volume de dados enviado. Este comportamento linear é esperado, contudo, as variações nas inclinações e interseções das linhas de regressão entre os diferentes cenários são de particular interesse. Estas diferenças quantificam o impacto das interferências no desempenho do módulo e são cruciais para o planejamento de sistemas de IoT eficientes.

Analisamos mais profundamente as condições onde o tempo de transmissão é minimizado, como nas situações de 'WiFi Diferente + Cliente' e 'WiFi + Microondas', que paradoxalmente mostram tempos menores do que o cenário sem interferência. Uma hipótese para este fenômeno poderia ser a adaptação dinâmica do módulo ESP8266 às condições do ambiente, possivelmente ajustando a potência de transmissão ou mudando para canais menos congestionados, um comportamento que requer investigação adicional.

3.1.2 Comparação entre Regressões

Ao comparar as linhas de regressão, observa-se que a interferência de dispositivos Bluetooth parece ter um efeito mais pronunciado no aumento do tempo de transmissão, conforme representado pela maior inclinação na sua linha de regressão. Por outro lado, a interferência de uma rede Wi-Fi diferente não parece ter um impacto tão significativo, indicando que o módulo ESP8266 possui alguma resiliência contra interferências de redes Wi-Fi coexistentes.

3.1.3 Implicações Práticas das Regressões Lineares

A regressão linear fornece uma ferramenta valiosa para prever o desempenho do módulo ESP8266 em aplicações do mundo real. Por exemplo, ao integrar o módulo em um dispositivo IoT que precisa operar em um ambiente com muitos dispositivos Bluetooth, poderíamos prever e mitigar o impacto potencial na eficiência da transmissão. As implicações desta análise são significativas para o desenvolvimento de soluções robustas em ambientes de IoT que são dinâmicos e repletos de fontes potenciais de interferência.

3.1.4 Considerações Finais sobre a Análise de Regressões Lineares

Em suma, a análise das regressões lineares em conjunto com os tempos médios de transmissão traz à luz a complexidade e a adaptabilidade do módulo ESP8266. A capacidade do módulo de manter a eficiência da transmissão, apesar das variadas fontes de interferência, é um testemunho da sua adequação para implementações de IoT em ambientes desafiantes. Estes resultados são um ponto de partida para futuras investigações e otimizações na comunicação sem fio e no projeto de sistemas de IoT resilientes.

3.2 Impacto das Interferências e Distância

A análise do impacto das interferências e da distância na eficiência da transmissão de dados foi conduzida com base em uma série de experimentos detalhados neste estudo. Como ilustrado na Figura 6, os resultados demonstram uma variação considerável dos tempos de transmissão em função das diferentes fontes de interferência e distâncias testadas.

Os dados obtidos revelam que a transmissão de dados é mais afetada em ambientes onde o módulo ESP8266 está sujeito a interferências de dispositivos como micro-ondas e Bluetooth, especialmente quando localizados a uma distância de 1 metro. As interferências de redes Wi-Fi concorrentes também influenciam a transmissão, mas em menor grau, sugerindo que o ESP8266 mantém uma transmissão de dados relativamente estável mesmo em canais compartilhados, a menos que obstáculos físicos estejam presentes.

Além disso, observou-se que o aumento da distância entre os módulos ESP8266 resulta em um incremento dos tempos de transmissão, sublinhando a

Comparação entre os tempos máximo, mínimo e médio

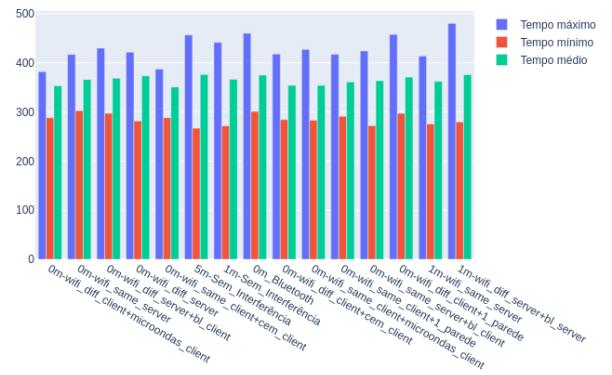


Figura 6: Comparação entre os tempos máximo, mínimo e médio de transmissão sob diferentes condições de interferência e distância.

importância da proximidade física para a eficiência da comunicação sem fio. As medições de tempo máximo, mínimo e médio apresentadas no gráfico fornecem um panorama quantitativo do desempenho sob várias condições, fundamentais para a compreensão e o desenvolvimento de estratégias de mitigação das interferências em sistemas de IoT.

Esses resultados são cruciais para a implementação de redes sem fio eficientes, enfatizando a necessidade de uma análise criteriosa das condições ambientais durante a fase de planejamento e execução de infraestruturas de comunicação sem fio.

3.3 Análise de Tukey e ANOVA

Nossa investigação sobre a influência das interferências na eficiência de transmissão do módulo ESP8266 utilizou uma abordagem estatística rigorosa para avaliar os dados de tempo de transmissão coletados. Através da técnica de regressão linear, estabelecemos modelos para analisar a relação entre o tamanho dos dados transmitidos e o tempo de transmissão em diferentes cenários de interferência. Esses modelos serviram de base para a subsequente análise de variância (ANOVA), que indicou uma Estatística F de 2.295292 e um p-valor de 0.003339, apontando para diferenças significativas no desempenho sob várias condições de interferência.

Prosseguindo com o teste de Tukey, buscamos especificar quais condições apresentavam diferenças estatisticamente significativas. Apesar da indicação inicial da ANOVA, o teste de Tukey não rejeitou a hipótese nula em nenhuma das 120 comparações realizadas. Com um p-valor ajustado médio de aproximadamente 0.91 e uma variação de p-valor ajustado de 0.14 a 1.0, as diferenças médias observadas de cerca de 3.33 não foram suficientes para estabelecer uma diferença estatística entre os grupos testados.

Estes resultados sugerem que, embora o mó-

Métrica	Valor
Dep. Variable	MaxTime
R-squared	0.708
Adj. R-squared	0.269
Model	OLS
Method	Least Squares
F-statistic	1.613
Prob (F-statistic)	0.289
Log-Likelihood	-66.543
AIC	153.1
Df Residuals	6
Df Model	9
Covariance Type	nonrobust

Tabela 6: Dados Gerais da Análise OLS do Max-Time

dulo ESP8266 possa ser afetado por diferentes tipos de interferência, as variações nos tempos de transmissão não podem ser consistentemente atribuídas a uma fonte específica com base nos critérios estatísticos adotados. A robustez do módulo em manter a eficiência de transmissão em diferentes cenários é evidenciada por essa falta de diferenças significativas, uma descoberta que reforça a versatilidade do ESP8266 para uso em ambientes IoT diversificados.

3.4 Análise dos Resultados da Regressão Linear Ordinária (OLS)

3.4.1 Escolha do Tipo de Redução de Dados

A eficácia do modelo OLS foi avaliada em explicar as variáveis dependentes MinTime, MaxTime e MeanTime. Observou-se que o modelo é mais eficaz para MeanTime (0.745 - Tabela 10), com R-squared ajustado significativamente mais alto em comparação com MinTime (0.012 - Tabela 8) e MaxTime (0.269 - Tabela 6), indicando que as condições de teste têm uma influência mais pronunciada no tempo médio de transmissão.

3.4.2 Avaliação dos Coeficientes para MeanTime

Os coeficientes do modelo OLS para MeanTime, Tabela 11 revelam nuances interessantes. O coeficiente constante de 277.1729 representa o tempo médio estimado quando todas as variáveis independentes são zero, o seja, sem interferência. Em relação às distâncias, coeficientes positivos e significativos para 'dist_1m', 'dist_0m' e 'dist_5m' mostram que o aumento da distância está associado a um aumento no tempo médio. Para interferências como 'cem_client', 'microondas_client', 'wifi_diff_client' e 'wifi_same_client', coeficientes negativos e significativos indicam uma relação inversa com o tempo médio. Já os coeficientes para Bluetooth e variáveis 'wifi_same_server' e 'wifi_diff_server' apresentam resultados mistos e não significativos.

3.4.3 Considerações Estatísticas

O modelo para MeanTime mostra boa adequação, porém a presença de coeficientes significativos e não significativos demanda uma interpretação cuidadosa. Isso inclui a consideração de multicolinearidade e outras complexidades inerentes ao modelo OLS.

3.4.4 Considerações Finais sobre a Análise dos Resultados da Regressão OLS

Essa análise oferece insights valiosos sobre como diferentes condições de interferência e distâncias afetam o tempo médio de transmissão de dados. Os resultados destacam a importância da distância como um fator determinante, enquanto sugerem que a presença de certas condições de interferência pode reduzir o tempo médio. No entanto, devido à complexidade das interações, mais investigações Em pesquisas futuras são necessárias para compreender completamente essas dinâmicas.

3.5 Discussões Tecnológicas

A eficiência na transmissão de dados é uma pedra angular no desenvolvimento de tecnologias de comunicação sem fio, especialmente em aplicações de Internet das Coisas (IoT), onde a confiabilidade e a velocidade são vitais. Nossa estudo sobre o módulo ESP8266, amplamente utilizado em soluções IoT, destaca a importância de considerar o ambiente de operação como um fator crítico na eficácia da comunicação.

A análise estatística dos tempos de transmissão aponta para uma robustez notável do módulo ESP8266 frente a diversas fontes de interferência. Contudo, a ausência de diferenças significativas entre os grupos de interferência, conforme revelado pelo teste de Tukey, sugere uma capacidade do dispositivo de manter a eficiência de transmissão, mesmo em condições que teoricamente poderiam comprometer a comunicação. Esta observação é encorajadora, porém, traz consigo uma série de considerações técnicas que merecem discussão.

Primeiramente, a imunidade do módulo a diferentes tipos de interferência pode ser interpretada como uma indicação da maturidade das tecnologias de comunicação sem fio atuais. Protocolos de comunicação avançados e técnicas de mitigação de interferência, como saltos de frequência e modulação adaptativa, estão se tornando cada vez mais eficientes, tornando os dispositivos menos suscetíveis a perturbações comuns.

Por outro lado, a alta densidade de dispositivos eletrônicos em ambientes urbanos e industriais é uma realidade que continua a desafiar os engenheiros de telecomunicações. O crescente número de dispositivos que competem pelo espectro de rádio gera um cenário onde a gestão de interferência se torna cada vez mais complexa. Isso nos leva a questionar: estamos realmente medindo todas as variáveis relevantes que afetam a transmissão de dados em ambientes reais?

Variável	Coef	Std Err	T	P> t	[0.025 0.975]
const	441.89	14.601	30.264	0.000	[406.161, 477.616]
wifi same client	-19.67	14.890	-1.321	0.235	[-56.102, 16.768]
cem client	-23.39	16.259	-1.439	0.200	[-63.173, 16.396]
wifi same server	-16.9841	22.075	-0.769	0.471	[-70.999, 37.031]
bl client	-19.71	23.414	-0.842	0.432	[-77.006, 37.577]
1 parede	13.11	16.259	0.806	0.451	[-26.673, 52.896]
bluetooth client	18.44	29.202	0.632	0.551	[-53.010, 89.899]
wifi diff server	26.97	25.885	1.042	0.338	[-36.370, 90.306]
microondas client	-21.39	16.259	-1.316	0.236	[-61.173, 18.396]
wifi diff client	-12.00	14.890	-0.806	0.451	[-48.435, 24.435]
bl server	11.4762	33.11	0.347	0.741	[-69.546, 92.498]

Tabela 7: Coeficientes e Estatísticas Associadas da Análise OLS para MaxTime

Métrica	Valor
Dep. Variable	MinTime
R-squared	0.605
Adj. R-squared	0.012
Model	OLS
Method	Least Squares
F-statistic	1.020
Prob (F-statistic)	0.510
Log-Likelihood	-53.271
AIC	126.5
Df Residuals	6
Df Model	9
Covariance Type	nonrobust

Tabela 8: Dados Gerais da Análise OLS para Min-Time

Além disso, a metodologia estatística aplicada, embora rigorosa, pode não capturar a totalidade das nuances do desempenho em campo. A modelagem e simulação de cenários de interferência, juntamente com testes em ambientes controlados, são essenciais, mas a validação em cenários reais de aplicação é inestimável. Questões como a interferência intermitente ou a degradação do sinal ao longo do tempo podem não ser totalmente abordadas em estudos labororiais.

As descobertas levantam também discussões sobre a necessidade de desenvolvimento contínuo de protocolos de comunicação que possam adaptar-se dinamicamente a ambientes altamente variáveis. A capacidade de um módulo de comunicação de ajustar sua estratégia de transmissão em tempo real, baseando-se na análise de condições ambientais em constante mudança, é um aspecto que poderia ser explorado mais profundamente em pesquisas futuras.

Em última análise, enquanto os resultados demonstram a resiliência do ESP8266, eles também evidenciam a complexidade inerente aos sistemas de comunicação sem fio modernos e a importância de abordagens multidisciplinares que integrem conhecimentos de engenharia de radiofrequência, processamento de

sinais e ciência da computação para otimizar a transmissão de dados em um mundo cada vez mais conectado.

4 Dificuldades Encontradas e Soluções Adotadas

4.1 Planejamento Excessivo de Testes

Inicialmente, enfrentamos um desafio logístico significativo com um planejamento de teste excessivamente ambicioso. A explosão combinatória das situações de interferência resultou em 80 casos de teste distintos, que, quando multiplicados por 6 distâncias diferentes (0m, 1m, 5m, 10m, 20m, e a distância máxima), totalizavam 480 testes individuais. Considerando uma duração média de 16 minutos por teste, isso implicava em aproximadamente 128 horas de testes ininterruptos, uma carga de trabalho claramente impraticável. Diante disso, realizamos adaptações no planejamento: apenas o cenário sem interferência foi testado a 5m; os cenários sem interferência, WiFi no mesmo canal, e WiFi + Bluetooth a 1m; e os demais apenas em 0m. Além disso, nem todos os casos foram testados, com a seleção dos cenários executados detalhada na seção de metodologia.

4.2 Contenção de Interferência WiFi Incontrolável

A ubiquidade do WiFi em ambientes urbanos apresentou uma barreira significativa para a realização de testes em condições de baixa interferência. Para contornar essa limitação, selecionamos locais estratégicos com mínima exposição a redes WiFi, como estacionamentos e terrenos baldios, além de realizar testes em horários alternativos, onde a atividade de rede tendia a ser menor.

4.3 Riscos Associados ao Uso do Micro-ondas e Furadeira

O uso do forno micro-ondas impôs riscos de danos ao ESP8266 devido à potente emissão de ondas eletromagnéticas, enquanto a furadeira gerava ruído

Variável	Coef	Std Err	T	P> t	[0.025 0.975]
const	271.00	6.37	42.54	0.000	[255.41, 286.59]
wifi_same_client	9.53	6.50	1.47	0.193	[-6.36, 25.43]
cem_client	4.63	7.09	0.65	0.538	[-12.72, 21.99]
wifi_same_server	13.48	9.63	1.40	0.211	[-10.09, 37.04]
bl_client	-3.10	10.22	-0.30	0.772	[-28.09, 21.90]
l_parede	12.63	7.09	1.78	0.125	[-4.72, 29.99]
bluetooth_client	30.00	12.74	2.36	0.057	[-1.17, 61.17]
wifi_diff_server	20.05	11.29	1.78	0.126	[-7.59, 47.68]
microondas_client	4.13	7.09	0.58	0.581	[-13.22, 21.49]
wifi_diff_client	11.87	6.50	1.83	0.118	[-4.03, 27.76]
bl_server	-11.38	14.45	-0.79	0.461	[-46.73, 23.97]

Tabela 9: Coeficientes e Estatísticas Associadas da Análise OLS para MinTime

Métrica	Valor
Dep. Variable	MeanTime
R-squared	0.898
Adj. R-squared	0.745
Model	OLS
Method	Least Squares
F-statistic	5.865
Prob (F-statistic)	0.0216
Log-Likelihood	-38.073
AIC	96.15
Df Residuals	6
Df Model	9
Covariance Type	nonrobust

Tabela 10: Dados Gerais da Análise OLS para MeanTime

acústico excessivo, incompatível com a realização de testes em ambiente residencial. As soluções adotadas incluíram a redução do tamanho máximo dos dados transmitidos e a alteração dos intervalos de teste, minimizando o tempo de exposição aos fatores de risco e o incômodo acústico, respectivamente.

4.4 Aprendizados e Descobertas

A complexidade dos desafios enfrentados proporcionou uma curva de aprendizado íngreme, onde adquirimos um conhecimento profundo sobre a operação do WiFi no ESP8266. A experiência culminou na descoberta da biblioteca *ESP8266WebServer.h*, que simplificou a criação de servidores de ponto de acesso para comportar-se como APIs REST, uma técnica que foi empregada com sucesso em projetos subsequentes, mas não neste para manter a homogenidade dos testes.

5 Conclusão

Este relatório explorou a eficácia do módulo ESP8266 em ambientes de redes Wi-Fi sob várias condições de interferência. Utilizando uma abordagem metodológica rigorosa, combinamos testes empíricos e

análises estatísticas, incluindo a regressão linear ordinária (OLS), para avaliar a influência de diferentes interferências e distâncias no tempo de transmissão de dados.

Durante nossos testes, observamos que a distância máxima efetiva para a transmissão de pacotes varia de acordo com o tamanho do pacote. Foi possível transmitir pacotes de até 50 bytes a uma distância de até 160 metros, enquanto a transmissão de pacotes de até 1024 bytes foi eficaz até uma distância máxima de 100 metros. Esses resultados são cruciais para entender as limitações físicas do módulo ESP8266 e suas aplicações práticas em diferentes cenários.

Nossos resultados revelaram que, embora o modelo OLS tenha sido eficaz na análise do tempo médio de transmissão (MeanTime), as variáveis independentes apresentaram um desempenho limitado em explicar as variações nos tempos mínimo (MinTime) e máximo (MaxTime) de transmissão. Isso sugere a necessidade de abordagens de modelagem mais robustas e uma coleta de dados mais extensa para capturar todos os fatores influentes.

Além disso, os testes mostraram que, em muitos casos, a interferência Wi-Fi não afetou significativamente a eficiência da transmissão, desafiando algumas premissas comuns sobre a vulnerabilidade do ESP8266 a interferências externas. Entretanto, as limitações logísticas e as adaptações necessárias no planejamento dos testes, especialmente a redução no número e na variedade dos cenários testados, apontam para a necessidade de mais pesquisas.

Para pesquisas futuras, recomendamos a ampliação do escopo dos testes, incluindo mais variações de tamanho de dados e distâncias, para obter um conjunto de dados mais representativo. Isso ajudará a melhorar a precisão dos modelos estatísticos e proporcionará insights mais confiáveis sobre o comportamento do ESP8266 em diferentes ambientes de rede. Essa abordagem mais abrangente é crucial para o desenvolvimento de sistemas IoT mais resilientes e adaptáveis às condições reais de operação.

Variável	Coef	Std Err	T	P> t	[0.025 0.975]
const	369.56	2.46	149.996	0.000	[363.54, 375.59]
wifi_same_client	-9.39	2.51	-3.738	0.010	[-15.54, -3.24]
cem_client	-9.77	2.74	-3.560	0.012	[-16.48, -3.05]
wifi_same_server	-4.82	3.73	-1.294	0.243	[-13.93, 4.30]
bl_client	-2.27	3.95	-0.575	0.586	[-11.94, 7.39]
1_parede	3.77	2.74	1.376	0.218	[-2.94, 10.49]
bluetooth_client	5.49	4.93	1.113	0.308	[-6.57, 17.54]
wifi_diff_server	2.52	4.37	0.578	0.584	[-8.16, 13.21]
microondas_client	-8.60	2.74	-3.135	0.020	[-15.31, -1.89]
wifi_diff_client	-5.20	2.51	-2.070	0.084	[-11.35, 0.95]
bl_server	3.88	5.59	0.694	0.514	[-9.79, 17.55]

Tabela 11: Coeficientes e Estatísticas Associadas da Análise OLS para MeanTime

Referências

- [1] TechTarget. Iot basics: A guide for beginners. <https://www.techtarget.com/whatis/feature/IoT-basics-A-guide-for-beginners>, 2023.
- [2] Random Nerd Tutorials. Getting started with esp8266 wifi transceiver review. <https://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/>, 2023.
- [3] Random Nerd Tutorials. Getting started with esp8266 wifi transceiver review: Micropython. <https://randomnerdtutorials.com/getting-started-with-esp8266-wifi-transceiver-review/#:~:text=,for%20microcontrollers%20and%20embedded%20systems>, 2023.
- [4] Qualtrics. What is anova?, 2023. Acessado em: 01/12/2023.
- [5] Sciencing. What is the tukey hsd test?, 2023. Acessado em: 01/12/2023.
- [6] Statistics How To. Tukey test / honest significant difference, 2023. Acessado em: 01/12/2023.
- [7] Built In. Understanding ordinary least squares (ols) regression. <https://builtin.com/data-science/ols-regression>, 2023.
- [8] Statology. Understanding the t-test in linear regression. <https://www.statology.org/t-test-linear-regression/>, 2023.