

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ
PRÓ-REITORIA DE PESQUISA, PÓS-GRADUAÇÃO E INOVAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Rui Guerra Falcão Moreno

**Avaliação de Desempenho de Redes em Malha Sem Fio em Cenários
de Cidades Inteligentes**

Fortaleza – CE

27 de abril de 2023

Rui Guerra Falcão Moreno

Avaliação de Desempenho de Redes em Malha Sem Fio em Cenários de Cidades Inteligentes

Dissertação submetida ao Programa de Pós-Graduação em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia do Ceará, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Linha de Pesquisa: Redes de Computadores

Orientador: Dra. Carina Teixeira de Oliveira

Instituto Federal de Educação, Ciência e Tecnologia do Ceará (IFCE)

Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação (PRPI)

Programa de Pós-Graduação em Ciência da Computação (PPGCC)

Fortaleza – CE

27 de abril de 2023

Dados Internacionais de Catalogação na Publicação

Instituto Federal do Ceará - IFCE

Sistema de Bibliotecas - SIBI

Ficha catalográfica elaborada pelo SIBI/IFCE, com os dados fornecidos pelo(a) autor(a)

MORENO, RUI GUERRA FALCÃO.

Avaliação de Desempenho de Redes em Malha Sem Fio em Cenários de Cidades Inteligentes / RUI GUERRA FALCÃO MORENO. - 2023.

64 f. : il. color.

Dissertação (Mestrado) - Instituto Federal do Ceará, Mestrado em Ciência da Computação, Campus Fortaleza, 2023.

Orientação: Profa. Dra. Carina Teixeira de Oliveira.

1. Avaliação de Desempenho. 2. Simulação. 3. Cidades Inteligentes. 4. WMN. I. Título.

CDD 004



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO CEARÁ
Av. 13 de maio, 2081 , - Bairro Benfica - CEP 60040-531 - Fortaleza - CE - www.ifce.edu.br

AVALIAÇÃO

Processo: 23256.001209/2023-99

Interessado: Rui Guerra Falcão Moreno

RUI GUERRA FALCÃO MORENO

Avaliação de Desempenho de Redes em Malha Sem Fio em Cenários de Cidades Inteligentes

Dissertação submetida à Coordenação do Curso de Pós-graduação em Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia do Ceará, como requisito parcial para a obtenção do título de mestre em Ciência da Computação, área de concentração Sistemas de Computação.

Aprovada em **07/03/2023**.

BANCA EXAMINADORA

Profa. Dra. Carina Teixeira de Oliveira (Orientador)

Instituto Federal de Educação, Ciência e Tecnologia do Ceará - PPGCC/IFCE

Profa. Dra. Atslands Rêgo da Rocha

Universidade Federal do Ceará - UFC

Prof. Dr. Antonio Wendell de Oliveira Rodrigues

Instituto Federal de Educação, Ciência e Tecnologia do Ceará - PPGCC/IFCE



Documento assinado eletronicamente por **Carina Teixeira de Oliveira, Professora do Ensino Básico, Técnico e Tecnológico**, em 07/03/2023, às 15:39, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Atslands Rego da Rocha, Usuário Externo**, em 07/03/2023, às 16:43, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Antonio Wendell de Oliveira Rodrigues, Professor do Ensino Básico, Técnico e Tecnológico**, em 16/03/2023, às 17:16, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade do documento pode ser conferida no site https://sei.ifce.edu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 informando o código verificador **4565161** e o código CRC **8377C8C3**.

*Aos meus pais, à minha esposa, aos meus familiares e
à todos os que sonharam em se tornar cientistas.*

Agradecimentos

Primeiramente, agradeço a Deus (Pai) pela sua infinita bondade e lealdade aos seus filhos, que, por meio de Cristo, tornou-se possível este vínculo. Sem Ele eu nada seria, faria e venceria.

Aos meus pais por todo amor, apoio, e paciência, desde a infância. E, pelos valores e princípios éticos e morais aprendidos.

À minha amada, querida e esposa Luciana pelo seu amor, carinho, motivação, paciência e companheirismo durante toda minha fase de mestrado e aos próximos caminhos a seguirmos juntos. À minha família, por parte dos meus pais e da minha esposa, e aos meus amigos por todo apoio, carinho e consolo.

À professora Dra. Carina pela sua imensa capacidade de incentivar, de motivar e de assistir seus alunos com maestria. À banca pelas considerações contribuindo na qualidade desta obra.

“Sei que nada sei.”
(Sócrates)

Resumo

A evolução dos meios de comunicações digitais ao longo dos últimos anos, principalmente das redes sem fio, resultou na comunicação dos mais diversos tipos de dispositivos eletrônicos, tais como *laptops*, *smartphones*, *tablets*, dispositivos de Internet das Coisas (*Internet of Things* - IoT), etc. Além disso, a evolução de aplicações e mídias digitais melhoraram, consideravelmente, a experiência dos usuários, podendo ser direcionada, apenas, para conteúdos almejados. Nesse contexto, é fácil perceber que uma cidade evolui a medida que serviços digitais são procurados e ofertados, permitindo que as cidades se transformem em serviços de inteligência conhecidos como cidades inteligentes (*smart cities*). Para uma cidade ser considerada inteligente é preciso que ela faça o uso de Tecnologias Digitais de Informação e Comunicação (TDIC), de serviços, de monitoramento, de gestão de forma mais consciente, interativa e eficiente. Com o grande desafio da heterogeneidade das informações e da interoperabilidade entre os dispositivos inteligentes que uma cidade inteligente exige, padrões de comunicações (ex: IEEE 802.11) foram melhorados e estão em melhora continuamente. Neste trabalho, estuda-se o padrão IEEE 802.11s para redes em malha sem fio (*Wireless Mesh Networks* - WMN) como solução de infraestrutura de redes para cidades inteligentes. Mais especificamente, avalia-se o desempenho do protocolo de roteamento proposto pelo IEEE 802.11s, o *Hybrid Wireless Mesh Protocol* (HWMP), em cenários típicos de cidades inteligentes (ex: com grande ou pequeno volume de informações, tamanhos diferentes de pacotes, diferentes topologias). Nos trabalhos relacionados, foram encontradas limitações como a falta de avaliação dos modos reativo e proativo e a ausência de análise das combinações de transmissão de mensagens de controle chamadas de somente destino (*Destination Only* - DO) e de resposta e encaminhamento (*Reply and Forward* - RF). Os modos reativo e proativo do HWMP, assim como os mecanismos do modo proativo, foram avaliados neste trabalho conforme as diferentes combinações de transmissão de mensagens de controle de roteamento DO e RF. Como o aumento da escalabilidade é um fator relevante em cidades inteligentes, pois não só mais dispositivos podem estar conectados como também mais mensagens podem ser trocadas entre os dispositivos, realizou-se a avaliação no *Network Simulator 3* (NS-3), que disponibiliza um módulo *mesh* seguindo o padrão IEEE 802.11s. Porém, foram realizadas alterações para atender aos objetivos da avaliação de desempenho. Os resultados do trabalho mostram que o modo reativo atuando juntamente com o proativo apresentou pior desempenho que o modo reativo atuando isoladamente, pois o número de mensagens de controle de roteamento foi superior no modo reativo com proativo. Os resultados e discussões podem ser usados por projetistas ou pesquisadores de redes para implantação ou melhoria de WMN em cidades inteligentes.

Palavras-chave: Avaliação de Desempenho. Simulação. Cidades Inteligentes. WMN.

Abstract

The evolution of digital means of communication over the last few years, mainly wireless networks, gave rise to in communication of the most diverse types of electronic devices, such as laptops, smartphones, tablets, IoT devices, etc. In addition, the evolution of applications and digital media have considerably improved the user experience, which can only be directed to targeted content. In this context, it is easy to see that a city evolves as digital services are sought and offered, allowing cities to become intelligence services known as smart cities. For a city to be considered smart, it needs to use Digital Information and Communication Technologies (DICT), services, monitoring, management in a more conscious, interactive and efficient way. As the great challenge of information heterogeneity and interoperability between smart devices that a smart city requires, communications standards (e.g. IEEE 802.11) have been improved and are in continuous improvement. In this work, the IEEE 802.11s standard for Wireless Mesh Networks (WMN) is studied as a network infrastructure solution for smart cities. More specifically, the performance of the standard IEEE 802.11s routing protocol, Hybrid Wireless Mesh Protocol (HWMP), is evaluated in typical scenarios of smart cities (e.g., with large or small volume of information, different sizes packages, different topologies). In the related works, the lack of evaluation of the reactive and proactive modes and the lack of analysis of the transmission combination of control messages called Destination Only (DO) were fully addressed and Reply and Forward (RF). The HWMP reactive and proactive modes, as well as the proactive mode switches, were evaluated in this work according to different combinations of transmission of DO and RF routing control messages. As the increase in scalability is a relevant factor in smart cities, because not only more devices can be connected but also more messages can be exchanged between devices, the evaluation was carried out in Network Simulator 3 (NS-3), which provides a mesh module following the IEEE 802.11s standard. However, alterations were changed to meet the objectives of the performance evaluation. The results of the work show that the reactive mode occurred simultaneously with the proactive one, presented worse performance than the reactive mode orally, since the number of routing control messages was higher in the reactive and proactive mode. The results and discussions can be used by network designers or researchers to implement or improve WMN in smart cities.

Keywords: performance evaluation. ns-3 simulator. hwmp. smart cities.

Lista de ilustrações

Figura 1 – Exemplo de Arquitetura IoT em 3 camadas.	21
Figura 2 – Exemplo de Arquitetura IoT em 5 camadas.	22
Figura 3 – Arquitetura IoT orientada a serviços (IoTaaS).	23
Figura 4 – Arquitetura em camadas baseada na heterogeneidade das redes sem fio no cenário de cidades inteligentes.	26
Figura 5 – Exemplo de Rede em Malha sem Fio.	28
Figura 6 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d1-r0.	31
Figura 7 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d0-r0.	32
Figura 8 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d0-r1.	33
Figura 9 – Diagrama de classe do módulo <i>mesh</i> do NS-3.	35
Figura 10 – Exemplos de Topologias <i>mesh</i> com 81 nós em um círculo de raio de 300m.	43
Figura 11 – Fluxograma do <i>script mesh</i>	44
Figura 12 – Fluxograma do script <i>runplot-flows</i>	45
Figura 13 – Resultado da métrica taxa de entrega para 71 nós.	47
Figura 14 – Resultado da métrica taxa de entrega para 81 nós.	48
Figura 15 – Resultado da métrica atraso médio para 71 nós.	50
Figura 16 – Resultado da métrica atraso médio para 81 nós.	50
Figura 17 – Resultado da métrica vazão agregada para 71 nós.	51
Figura 18 – Resultado da métrica vazão agregada para 81 nós.	51
Figura 19 – Resultado da métrica soma de mensagens de controle na rede para 71 nós.	52
Figura 20 – Resultado da métrica soma de mensagens de controle na rede para 81 nós.	53
Figura 21 – Resultado da métrica total de mensagens PREQ na rede para 71 nós.	54
Figura 22 – Resultado da métrica total de mensagens PREQ na rede para 81 nós.	55
Figura 23 – Resultado da métrica total de mensagens PREP na rede para 71 nós.	56
Figura 24 – Resultado da métrica total de mensagens PREP na rede para 81 nós.	56
Figura 25 – Resultado da métrica total de mensagens PERR na rede para 71 nós.	57
Figura 26 – Resultado da métrica total de mensagens PERR na rede para 81 nós.	57

Lista de tabelas

Tabela 1 – Aplicações em Cidades Inteligentes e suas características	27
Tabela 2 – Comparativo de trabalhos	40
Tabela 3 – Parâmetros de Entrada para as Simulações.	42

Lista de abreviaturas e siglas

ALM	<i>Airtime Link Metric</i>
AODV	<i>Ad hoc On-demand Distance Vector</i>
AP	<i>Access Point</i>
API	<i>Application Programming Interface</i>
BATMAN	<i>Better Approach To Mobile Ad-hoc Networking</i>
DO	<i>Destination Only</i>
DSN	<i>Destination Sequence Number</i>
DDoS	<i>Distributed Deny of Services</i>
DICT	<i>Digital Information and Communication Technologies</i>
DSDV	<i>Destination-Sequenced Distance Vector</i>
DSR	<i>Dynamic Source Routing</i>
FANETs	<i>Flying Ad hoc Networks</i>
HWMP	<i>Hybrid Wireless Mesh Protocol</i>
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
IoT	<i>Internet of Things</i>
IoTaaS	<i>Internet of Things as a Service</i>
ITS	<i>Intelligent Transportation Systems</i>
NS-2	<i>Network Simulator versão 2</i>
NS-3	<i>Network Simulator versão 3</i>
OLSR	<i>Optimized Link State Routing</i>
PHY	<i>Physical layer</i>
PERR	<i>Path ERRor</i>
PREP	<i>Path Reply</i>

PREQ	<i>Path Request</i>
QoS	<i>Quality of Service</i>
RANN	<i>Root ANNouncement</i>
REST	<i>Representational State Transfer</i>
RF	<i>Reply and Forward</i>
RFID	<i>Radio Frequency Identification Devices</i>
SAODV	<i>Secure AODV</i>
SLRR-HWMP	<i>Scalable Local Route Repair-HWMP</i>
TDIC	Tecnologias Digitais de Informação e Comunicação
TTL	<i>Time-To-Live</i>
VANETs	<i>Vehicular Ad hoc Networks</i>
WLAN	<i>Wireless Local Area Networks</i>
WMN	<i>Wireless Mesh Network</i>
WSN	<i>Wireless Sensor Network</i>
UAV	<i>Unmanned Aerial Vehicles</i>
UN	<i>United Nations</i>

Sumário

1	INTRODUÇÃO	16
1.1	Objetivos	18
1.1.1	Objetivo Geral	18
1.1.2	Objetivos Específicos	18
1.2	Artigos publicados	18
1.3	Estrutura da Dissertação	19
2	FUNDAMENTAÇÃO TEÓRICA	20
2.1	Internet das Coisas	20
2.2	Cidades Inteligentes	25
2.3	Redes em Malha Sem Fio	28
2.3.1	Conceitos Básicos do IEEE 802.11s	28
2.3.2	Roteamento em Redes <i>Mesh</i> IEEE 802.11s	29
2.4	Simulador de redes NS-3	33
3	TRABALHOS RELACIONADOS	36
4	AVALIAÇÃO DE DESEMPENHO	41
4.1	Etapas 1: Entendimento do domínio de estudo	41
4.2	Etapas 2: Preparação do <i>script</i> de entrada para as simulações no NS-3	41
4.3	Etapas 3: Simulações no NS-3	43
4.4	Etapas 4: Preparação de um <i>script</i> para exibição dos resultados das simulações em gráficos amigáveis	45
4.5	Etapas 5: Análise dos resultados das simulações	46
5	RESULTADOS	47
5.1	Taxa de Entrega	47
5.2	Atraso Médio	49
5.3	Vazão agregada	50
5.4	Soma das mensagens de controle na rede	51
5.5	Total das mensagens de controle na rede	53
5.5.1	PREQ	53
5.5.2	PREP	55
5.5.3	PERR	56
6	CONCLUSÃO E TRABALHOS FUTUROS	58

REFERÊNCIAS 59

1 Introdução

A forma como nos comunicamos através dos meios digitais mudaram profundamente ao longo dos últimos anos, principalmente com o avanço das redes de comunicação sem fio e dos dispositivos eletrônicos, que, por meio delas, se comunicam entre si. Além das altas taxas de transferência de dados, a disseminação das informações proporcionou o desenvolvimento de aplicações em dispositivos como *laptops*, *smartphones*, *tablets*, *smart TVs*, *smart watches*, dispositivos de Internet das Coisas (*Internet of Things* - IoT) em geral, etc. Ademais, a evolução de aplicações e mídias digitais melhoraram, consideravelmente, a experiência dos usuários, podendo ser direcionada, apenas, para conteúdos digitais almejados.

Nesse contexto, as redes locais sem fio (*Wireless Local Area Networks* - WLAN), popularmente conhecidas como *Wireless Fidelity* (Wi-Fi), ampliaram a conectividade dos dispositivos sem a necessidade de uma infraestrutura cabeada, reduzindo custos de instalação da rede. Além disso, é cada vez maior o volume de dados gerados pelas aplicações instaladas nesses dispositivos (FELDMANN et al., 2021).

A família de padrões IEEE 802.11 é utilizada pelas redes WLAN (DENG; CHEN; CHENG, 2014). Esses padrões podem descrever, por exemplo, a camada física (PHY) e a camada de controle de acesso ao meio (*Medium Access Control* - MAC) para diferentes frequências de comunicação sem fio, como para as faixas de 2.4 GHz e 5 GHz. Na verdade, desde o seu lançamento em 1997, várias gerações surgiram, como o IEEE 802.11 a, b, g, n, ac, e para suporte a qualidade de serviços (*Quality of Service* - QoS), i para segurança, p para ambientes veiculares e, no contexto desse trabalho de dissertação, o IEEE 802.11s para redes em malha sem fio (*Wireless Mesh Networks* - WMN) (IEEE... , 2011).

As WMNs têm conquistado, de forma crescente, a atenção nos últimos anos em virtude de seu baixo custo, facilidade de implantação, robustez e cobertura de serviço confiável. Através desta tecnologia é possível ofertar uma cobertura sem fio em banda larga para grandes áreas (como centros urbanos), sem grandes exigências em termos de infraestrutura. Além disso, é capaz de assegurar a conectividade em ambientes com dinamicidade das condições do meio sem fio e com usuários móveis. Por todas estas vantagens, as WMNs têm sido implementadas em diversos tipos de ambientes, tais como residências, campus universitários, edifícios empresariais, sistemas de transporte inteligentes (*Intelligent Transportation Systems* – ITS), sistemas de veículos aéreos não tripulados e, no contexto desta dissertação, em **idades inteligentes** (*smart cities*) (HAKIRI; GOKHALE; BERTHOU, 2019)

Segundo Alrikabi e Jasim (2021), uma cidade inteligente pode ser compreendida

como um ambiente que faz uso de Tecnologias Digitais de Informação e Comunicação (TDIC), de serviços, de monitoramento e de gestão de forma mais consciente, interativa e eficiente. Conforme Washburn et al. (2009), uma cidade inteligente é capaz de agir, de forma ágil, à problemas relacionados ao transporte, à saúde, à educação, à segurança, à economia e aos demais problemas urbanos através do gerenciamento de diversos dispositivos espalhados pela cidade.

Em um ambiente de cidade inteligente, é esperado um número crescente de dispositivos heterogêneos, tanto dos dispositivos “convencionais” (*laptops, smartphones, etc.*) quanto, principalmente, de dispositivos no contexto de Internet das Coisas (*Internet of Things* - IoT). Por exemplo, um relatório de Transforma Insights (2020) prevê que mais de 29 bilhões de dispositivos IoT estarão conectados no mundo todo em 2030. Um aumento de quase três vezes mais da quantidade de dispositivos IoT em relação a 2020, que é de 9,7 bilhões. Além disso, estima-se que, em 2030, 60% dos dispositivos, presentes no setores industriais e nos setores de consumo, serão de IoT.

Com o crescente aumento do número de dispositivos, não restam dúvidas de que são necessárias infraestruturas de redes robustas capazes de dar o suporte adequado ao grande volume de dados gerados por tais dispositivos heterogêneos. Um dos principais desafios do uso dessas infraestruturas de redes quando há um aumento da escalabilidade é reduzir, ao máximo, a perda do desempenho da rede.

Neste trabalho, propõe-se o estudo das WMN como uma das infraestruturas de redes para cidades inteligentes, com foco na avaliação de desempenho do protocolo de roteamento híbrido de redes em malha sem fio (*Hybrid Wireless Mesh Protocol* - HWMP) do IEEE 802.11s. As características e o comportamento foram analisados diante de comparações entre cenários com alto e baixo fluxo de dados, com diferentes modos de operações (reativo, proativo), entre mensagens de controle de roteamento e entre pacotes com tamanhos diferentes, representando as aplicações típicas de cidades inteligentes, conforme será visto no Capítulo 2. A avaliação foi realizada em um simulador de redes de *software* livre, que permite simular ambientes reais de redes com a ajuda de bibliotecas *helpers* contendo as mais variadas implementações de protocolos, atuais, do mundo real. O simulador utilizado é o *Network Simulator 3* (NS-3)¹, que já disponibiliza um módulo *mesh* seguindo o padrão IEEE 802.11s. Porém, foi necessário realizar alterações para atender aos objetivos do trabalho.

Embora exista uma vasta quantidade de pesquisas relacionadas às cidades inteligentes e IoT, poucos trabalhos relacionam-as com as WMN (HAKIRI; GOKHALE; BERTHOU, 2019; PARVIN, 2019; KASHEF; VISVIZI; TROISI, 2021; RETHFELDT et al., 2021; TONG et al., 2021; WANG et al., 2021). Em particular, poucos trabalhos abordam o funcionamento do protocolo HWMP em cenários de cidades inteligentes, conforme

¹ <https://www.nsnam.org/>

mostrado no Capítulo 3. E, quando abordam, ou estão relacionados a outros cenários, ou estão com métricas e parâmetros de avaliações que não medem o impacto das mensagens de roteamento. Além disso, em nenhum dos trabalhos pesquisados foram realizados estudos que abordassem, analiticamente, o impacto dos modos reativo e proativo do HWMP em conjunto com as diferentes possibilidades de definição de *flags* de roteamento do HWMP.

1.1 Objetivos

1.1.1 Objetivo Geral

O objetivo geral do trabalho é realizar uma avaliação de desempenho de redes malha sem fio em cenários de cidades inteligentes.

1.1.2 Objetivos Específicos

Os objetivos específicos do trabalho são:

- Fazer uma avaliação de desempenho de WMN utilizando o padrão IEEE 802.11s em cenários de cidades inteligentes, com foco especial no protocolo de roteamento (*Hybrid Wireless Mesh Protocol* - HWMP);
- Averiguar o desempenho do HWMP em WMN com o auxílio de uma ferramenta de simulação de redes (simulador NS-3);
- Avaliar o desempenho do HWMP em função do volume de dados injetado na WMN em cenários de cidades inteligentes;
- Aferir o efeito do tamanho de pacotes transmitidos na WMN em cenários de cidades inteligentes;
- Avaliar o desempenho do HWMP em função da quantidade de nós na WMN em cenários de cidades inteligentes;
- Avaliar os impactos das mensagens de controle do HWMP no desempenho das WMNs em cenários de cidades inteligentes;
- Avaliar os impactos das *flags* somente destino (*Destination Only* - DO) e responde e encaminha (*Reply and Forward* - RF) do HWMP no desempenho das WMNs em cenários de cidades inteligentes.

1.2 Artigos publicados

- MORENO, R. G. F.; OLIVEIRA, C. T. **Avaliação de Desempenho de Redes em Malha Sem Fio em Cenários de Cidades Inteligentes.** In: ESCOLA REGIONAL DE COMPUTAÇÃO DO CEARÁ, MARANHÃO E

- PIAUÍ (ERCEMAPI), 10. , 2022, São Luís. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022 . p. 109-118. (MORENO; OLIVEIRA, 2022a)
- MORENO, R. G. F.; OLIVEIRA, C. T. **Um Estudo de Desempenho do HWMP em Ambientes de Cidades Inteligentes**. *In*: TRABALHOS EM ANDAMENTO - SIMPÓSIO BRASILEIRO DE ENGENHARIA DE SISTEMAS COMPUTACIONAIS (SBESC), 12. , 2022, Fortaleza/CE. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022 . p. 80-85. ISSN 2763-9002. (MORENO; OLIVEIRA, 2022b)

1.3 Estrutura da Dissertação

O restante do documento está estruturado da seguinte forma: no Capítulo 2 são apresentados conceitos de Internet das Coisas (*Internet of Things* - IoT), de cidades inteligentes, do padrão IEEE 802.11s e, de forma mais detalhada, o protocolo HWMP, bem como seus modos de operação (reativo e proativo) e os mecanismos de controle de mensagens do HWMP; no Capítulo 3 são apresentados os trabalhos relacionados à temática da dissertação; no Capítulo 4 é detalhada a avaliação de desempenho; no Capítulo 5 são apresentados os resultados; no Capítulo 6 são apresentadas as considerações finais deste trabalho e encaminhamentos para trabalhos futuros.

2 Fundamentação Teórica

Este capítulo aborda o conceito de Internet das Coisas (*Internet of Things* - IoT) e de cidades inteligentes. Será visto os conceitos básicos do padrão IEEE 802.11s e, de forma mais detalhada, o protocolo de roteamento recomendado pelo padrão IEEE 802.11s, como o modo de operação reativo e proativo e o mecanismo de controle de mensagens do HWMP. Será visto, também, como as diferenciações de tamanhos de pacotes podem estar relacionadas à diferentes aplicações em cidades inteligentes. Outrossim, o software de simulação NS-3 e alguns de seus módulos de implementação são apresentados para entendimento da proposta e dos resultados deste trabalho.

2.1 Internet das Coisas

Os avanços da tecnologia e da ciência têm proporcionado uma melhora nas atividades cotidianas das pessoas. Geralmente, quando há alguma notícia sobre inovação, automaticamente, pensa-se em tecnologia. O fomento da inovação vem do desconforto de situações vividas que poderiam ser evitadas e, conseqüentemente, melhoradas (DUTTA; GHOSH; BAYOUMI, 2019).

Cada vez mais, a Internet se torna presente em objetos físicos. Um exemplo é o monitoramento das ruas através de câmeras e de sensores em cidades inteligentes. Os dispositivos IoT (como sensores e câmeras) coletam, por exemplo, a quantidade de veículos nas ruas, a velocidade, as condições temporais e o congestionamento em um determinado período. Esses dados são transmitidos para nuvens que realizam o processamento dessas informações e repassam para aplicações específicas (voltadas ao transporte público, ao monitoramento da segurança nas ruas, etc) (OTHMAN; EL-MOUSA, 2020).

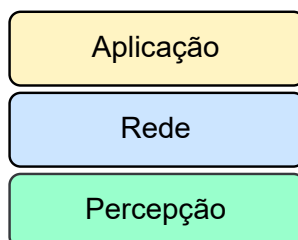
Nas últimas duas décadas, a definição do termo IoT sofreu e vem sofrendo mudanças haja vista a evolução dos dispositivos inteligentes e da visão que cada entidade pública ou privada à propõe. Segundo Rose, Eldridge e Chapin (2015), IoT pode ser definida, de forma generalista, como cenários onde objetos, sensores e todo tipo de item (físico ou virtual), não considerado como dispositivo de uso geral, estejam interligados à uma rede permitindo a troca, o consumo e a leitura de dados com o mínimo de intervenção humana. Os autores comentam que não há uma definição universal de IoT.

A presença de bilhões de dispositivos heterogêneos tornou o ambiente de IoT complexo de tal maneira que precisaram ser elaboradas arquiteturas na tentativa de entender melhor os papéis de cada parte desse ambiente. Na literatura, encontra-se arquiteturas IoT baseadas em 3 camadas, 5 camadas, ou integrada a outras camadas da

nuvem (*cloud*). Assim sendo, não há uma padronização de arquitetura IoT.

A arquitetura em 3 camadas é a mais simples e básica entre os tipos de arquitetura presentes na literatura. Essa arquitetura, como a proposta por Wu et al. (2010) (Figura 1), é composta pela camada de percepção, de rede e de aplicação. A camada de percepção (ou camada de dispositivos) realiza a coleta de dados através de sensores, atuadores, dispositivos de identificação por radio frequência (*Radio Frequency Identification Device* - RFID) e/ou de uma rede de sensores sem fio (*Wireless Sensor Network* - WSN). É nessa camada que os dados são detectados e identificados, como o movimento, a vibração, a umidade, a temperatura, entre outros. Esses dados são enviados para a próxima camada chamada de rede, na qual é feita a conexão entre os nós da rede e os variados dispositivos da camada de percepção. As tecnologias de transmissão presentes na camada de rede podem ser 4G, Wi-Fi, *bluetooth* (IEEE 802.15.1) ou Zigbee (IEEE 802.15.4), que transmitem os dados coletados da camada de percepção para a camada de aplicação. Na camada de aplicação, é realizado o gerenciamento, o processamento e o armazenamento dos dados obtidos das camadas inferiores. Essa camada tem como principal função auxiliar a tomada de decisão referente a análise dos dados para encaminhá-los aos serviços solicitados pelas aplicações de forma direcionada (KENAZA et al., 2022).

Figura 1 – Exemplo de Arquitetura IoT em 3 camadas.



Fonte: Adaptada de Wu et al. (2010)

Uma outra arquitetura existente na literatura está representada pela Figura 2, dessa vez dividida em 5 camadas. A camada de objetos (ou camada de detecção) representa os sensores e atuadores desempenhando diferentes funções. Nessa camada, pode-se fazer o uso da tecnologia *Universal Plug and Play*¹, que foi projetada para facilitar a conectividade e instalação entre dispositivos de fabricantes variados através de uma arquitetura de redes ponta a ponta. Essa tecnologia garante a auto-configuração dos objetos heterogêneos na camada de objetos. A camada seguinte, chamada de abstração de objetos, é responsável por transmitir os dados recebidos da camada anterior para a camada de gerenciamento através das tecnologias de rádio frequência (WiFi, 3G/4G/5G, Bluetooth, etc.). O processamento de dados, a tomada de decisão e a entrega por serviços essenciais estão presentes na camada de gerenciamento (ou *Middleware*). Além disso, ela facilita o desenvolvimento

¹ <https://www.iso.org/obp/ui/#iso:std:69286:en>

com objetos heterogêneos uma vez que não há a necessidade de adquirir uma plataforma específica. A camada de aplicação é responsável por entregar serviços inteligentes com qualidade conforme a demanda do cliente. Estão presentes nesta camada as aplicações de casa inteligente (*Smart Home*), construções inteligentes (*Smart Buildings*), saúde assistida inteligente (*Smart Healthcare*), automação industrial e entre outros. A camada de negócios inteligentes (ou *Business Intelligence*) é caracterizada pelo gerenciamento de todas as atividades e serviços do sistema de IoT. Ela é responsável pela construção de modelos de negócios em forma de gráficos, fluxogramas, painéis de controle e entre outros através da análise de dados recebidos da camada de aplicação. A camada de negócio não só projeta, analisa, monitora e implementa elementos relacionados ao sistema IoT como também auxilia na tomada de decisão com base na análise de distribuição massiva de dados (*Big Data*). Essa camada compara cada saída de cada camada com a saída esperada para melhorar os serviços e assegurar a privacidade dos usuários (AL-FUQAHA et al., 2015).

Figura 2 – Exemplo de Arquitetura IoT em 5 camadas.

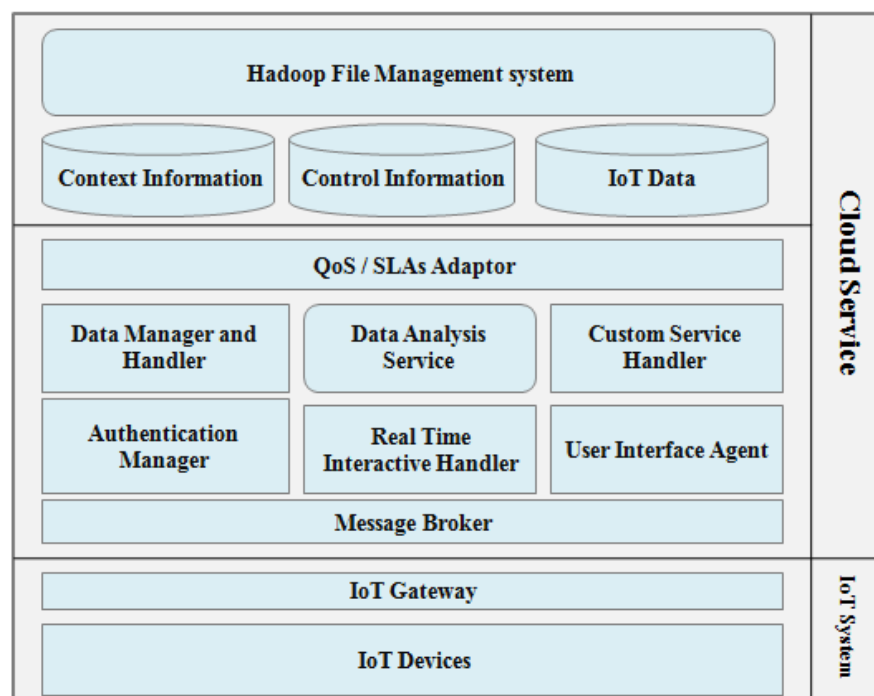


Fonte: Adaptada de Al-Fuqaha et al. (2015)

Uma outra abordagem de organização de sistema IoT foi proposta por Othman e El-Mousa (2020), chamada de Internet das Coisas orientada a Serviços (*Internet of Things as a Service* - IoTaaS). Nesse modelo, o serviço de nuvem atua como: um intermediário entre os dispositivos IoT e os usuários finais, um *middleware* para conectar os dispositivos IoT à Internet, um provedor de serviços para armazenamento e análise de dados (*Data Analysis*), um mediador para aplicações IoT e para serviços. Para suportar a interoperabilidade nesse modelo, unidades de serviços foram definidas para atuar de forma independente. Assim sendo, o sistema IoT se torna compatível para qualquer provedor de serviços de nuvem. Além disso, modelos de entrega de serviços foram elaborados em decorrência da necessidade de dar suporte às aplicações IoT para que sejam flexíveis e sob demanda baseados nas necessidades dos usuários. Esses modelos de entregas são chamados de monitoramento e análise como serviço (*Monitoring and Analysis as a Service* - MAaaS), de detecção e atuação como serviço (*Sensing and Actuating as a Service* -

SAaaS) e de publicação e subscrição para o recebimento de notificações (*Publish/Subscribe notifications*), que se comunicam com o sistema IoT através da interface de programação de aplicação (*Application Programming Interface* - API) chamada de transferência de estado representacional (*Representational State Transfer* - REST). O primeiro serviço (MAaaS) é responsável pela análise e monitoramento de uma grande quantidade de dados armazenados na nuvem, por exemplo: sistema de monitoramento do tempo meteorológico, vigilância da logística de transportes em uma cidade inteligente, etc. O serviço de SAaaS realiza a detecção e a atuação sobre as interações, em tempo real, entre os usuários e o sistema IoT, exercendo um papel de mediador para agir através do sistema IoT. Um exemplo para isso seria o controle e o monitoramento, remotamente, de termostatos em uma residência com o intuito de evitar incêndios. Outro serviço importante é o serviço de publicação e de subscrição para o recebimento de notificações através das atualizações do sistema IoT. Quando os usuários realizam a inscrição em um determinado serviço, eles recebem essas notificações, mantendo-se atualizados, por exemplo: o recebimento de notificações a respeito de um congestionamento de uma via causada por um acidente ou de um bloqueio de vias devido à situações temporais que impedem o trânsito de veículos.

Figura 3 – Arquitetura IoT orientada a serviços (IoTaaS).



Fonte: Othman e El-Mousa (2020)

A Figura 3 apresenta esta arquitetura orientada a serviços distribuída em serviços de nuvem e em sistema IoT. O sistema IoT é composto por *gateways* IoT e por dispositivos IoT (*IoT devices*). Juntos eles formam um *middleware*, que se conecta a um *message broker*. A função do *message broker* é interligar o sistema IoT aos serviços de nuvem, e vice-versa,

solicitados pelos usuários das aplicações. O gerenciador de autenticação (*Authentication Manager*) verifica as credenciais do sistema, trata do estabelecimento de sessões do sistema IoT e da troca de parâmetros. Ao estabelecer uma sessão, os parâmetros de conexões são passados para o *message broker* para permitir o tráfego de entrada de mensagens. O serviço de gerenciamento e manipulação de dados (*data manager and handler*) é responsável por tratar e atualizar os dados recebidos do sistema IoT ou das solicitações de serviços pelas aplicações. Esse serviço também controla as consultas de dados através da execução dos comandos de manipulação de dados: *select*, *update*, *delete*. Além disso, é nele que informações de contexto (*Context Information*) de sessões anteriores podem ser utilizadas para predições do comportamento dos usuários, melhorando, dessa forma, o tempo de resposta e contribuindo para a qualidade do serviço (*Quality of Services* - QoS). O serviço manipulador de interação em tempo real (*Real Time Interactive Handler*) é responsável pelo controle das interações entre os usuários e as aplicações em tempo real. Esse serviço é o mais inteligente e o mais complexo por realizar tarefas próprias da inteligência artificial, auxiliando na tomada de decisões dos modos de operação com base na qualidade dos serviços (QoS). Por meio dos serviços *web*, o proprietário do sistema IoT é controlado e gerenciado através da interface do usuário *User Interface Agent* remotamente. Outro modulo presente em IoTaaS é o manipulador de serviços customizados (*Custom Service Handler*), que fornece serviços estendidos para realizar operações de funções complexas para aplicações IoT. O adaptador de qualidade de serviços (QoS Adaptor), conforme o acordo a nível de serviços (SLAs), monitoram o desempenho dos serviços ofertados para realizar ajustes, quando necessário, nos recursos, garantindo a melhora do nível de serviços. Como mencionado anteriormente, a qualidade dos serviços (QoS) utiliza as informações de contexto das sessões anteriores para predições de recursos. O serviço de análise de dados consiste em ferramentas de análises que dão suporte a consultas e a funções de análises de *Big Data* sob demanda, fornecendo recursos para analisar, filtrar e visualizar registros de dados IoT. A camada de banco de dados (*Database Layer*) é constituída pelos módulos informações de contexto (*Context Information*), informações de controle (*Control Information*), e dados IoT (*IoT Data*). É nessa camada que é feito o armazenamento das mais variadas informações referentes ao tráfego de dados com base no contexto e no controle dessas informações. Por fim, o sistema de gerenciamento de arquivo do servidor Hadoop (*Hadoop File Management System*) é quem administra a grande quantidade de dados (*Big Data*) gerados pelas interações em tempo real das aplicações, podendo ser suportado por banco de dados não relacionais (NoSQL).

Todas essas arquiteturas têm como objetivo solucionar os desafios enfrentados pelos sistemas IoT, como segurança, desempenho, consumo de energia, entre outros.

2.2 Cidades Inteligentes

A civilização contemporânea está, pouco a pouco, iniciando em uma nova fase digital em virtude da alta velocidade e da alta qualidade de conexão que as redes de comunicações possibilitam. O termo cidade inteligente tem ganhado força nos últimos anos. Em razão desses avanços, a implementação de aplicações inteligentes vem crescendo, como *Smart Home*, *Smart Wearable*, *Smart Healthcare*, *Smart Energy*, etc. A urbanização da população mundial, bem como seus efeitos, resultam em problemas não só centralizados nas atividades humanas, como também nas necessidades sociais, econômicas e ambientais. Um estudo feito pelas Nações Unidas (*United Nations* - UN) prevê um percentual de urbanização, no mundo, em 66% até 2050. O crescimento de novas cidades emergentes e de novas megacidades ajudam a compreender a importância de transformá-las em cidades inteligentes (TONG et al., 2021).

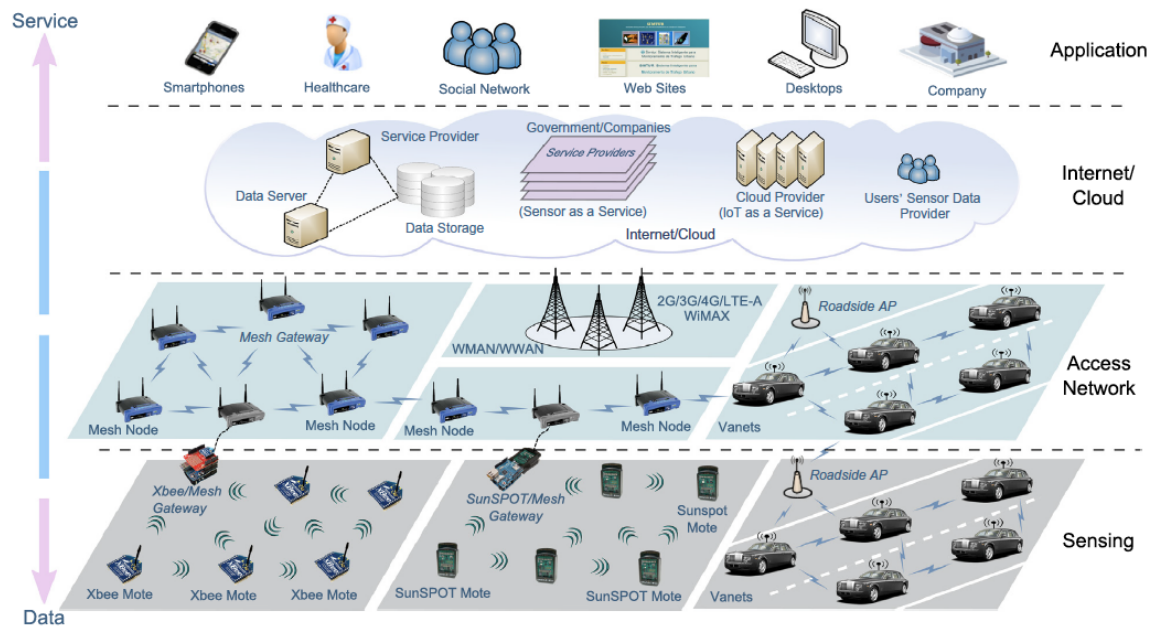
Na tentativa de redução dos efeitos causados pela urbanização, pesquisadores sugeriram o padrão de mudança social (ou, planejamento urbano sustentável) na década de 1990. O êxodo rural é o principal fator para o crescimento populacional descontrolado das grandes cidades. Por mais que indústrias se desloquem ou para pequenas cidades, ou para zonas rurais, as cidades competem entre si para atrair os melhores profissionais, ofertando as melhores condições financeiras (AVELAR et al., 2015). Os governantes das grandes cidades, cobrados a responder a esses problemas, adotaram uma abordagem de interconectar boa parte das regiões da cidade governada para transformá-las em cidades inteligentes.

Nesse contexto, apresentou-se uma arquitetura de redes baseada em 4 camadas para representar a heterogeneidade dos objetos em redes sem fio em cidades inteligentes (AVELAR et al., 2015).

Nessa arquitetura de 4 camadas (Figura 4), a primeira camada, denominada aplicação (*application*), é constituída por diferentes tipos de aplicações consumidoras de dados, coletados pela camada de sensoriamento (*sensing*). A segunda camada representada por Internet/nuvem (*Internet/Cloud*) é uma solução para disponibilização de dados através da Internet e para execução e distribuição massiva de dados (*Big Data*) através de servidores *web* e banco de dados. A terceira camada, acesso à redes (*access network*), pode ser formada por diversos tipos de roteadores compatíveis com a tecnologia IEEE 802.11 formando, por exemplo, uma rede *mesh*. Além disso, redes *ad hoc* veiculares (*Vehicular Ad hoc NETWORKS* - VANETs), e redes 2G/3G/4G podem fazer parte dessa camada para propagar os dados vindo dessas redes, permitindo maior interoperabilidade entre os dispositivos. Os modelos de redes tornam-se mais robustos com a auto-organização e a auto-configuração que os protocolos de roteamento realizam, evitando falhas que podem existir entre os nós durante a transmissão de dados. A última camada (sensoriamento) é definida como um conjunto de diversos sensores que proveem dados sensíveis para os servidores de nuvem, que realizam

gravações e decisões ao processarem esses dados.

Figura 4 – Arquitetura em camadas baseada na heterogeneidade das redes sem fio no cenário de cidades inteligentes.



Fonte: Avelar et al. (2015)

Uma abordagem futurística para cidades inteligentes foi proposta por Javed et al. (2022). Elas, também, podem ser chamadas de Sistema de Sistemas sobre um ângulo de 360°, ou melhor, englobando a maioria das interações entre os sistemas conhecidos, as cidades inteligentes. As aplicações que envolvem esse olhar de 360° se resumem em 4 principais pilares: Vida Inteligente (*Smart Life*), Sociedade Inteligente (*Smart Citizen*), Governança Inteligente (*Smart Governance*) e Ambiente Inteligente (*Smart Environment*). Segundo o autor, essas aplicações cobrem todo tipo de cenário que uma cidade inteligente se organiza para suprir as demandas de uma cidade.

Um estudo feito para analisar e relacionar os diferentes tamanhos de pacotes por cada aplicação de *smart city* foram observados, como mostra a Tabela 1. A primeira coluna (aplicação) categoriza os tipos de aplicações de *smart cities* presentes na literatura, são elas: Assistência Médica Inteligente (*Smart Healthcare*), Medição Inteligente (*Smart Metering*), Semáforos Inteligentes (*Smart Traffic Lights*), Transporte Público Inteligente (*Smart Public Transportation*), Estacionamento Inteligente (*Smart Parking*), Segurança Pública Inteligente (*Smart Public Safety*), Câmeras de Vigilância Inteligente (*Smart Cam - Surveillance*), Sensor de Qualidade do Ar (*Air Quality Sensor*), Iluminação Inteligente (*Smart Lightning*) e Medidor de Água Inteligente (*Smart Water Meter*). As informações

apresentadas na segunda coluna (tamanho de pacotes) representam uma média do tamanho de pacotes trafegados conforme a aplicação relacionada. Por último, a terceira coluna (referência) menciona os autores que fizeram a análise de cada aplicação. No próximo parágrafo, será tratado, brevemente, os trabalhos desses autores.

Tabela 1 – Aplicações em Cidades Inteligentes e suas características

Aplicação	Tamanho pacotes	Referência
<i>Smart Healthcare</i>	1024 bytes	Saxena e Raychoudhury (2019)
<i>Smart Metering</i>	150 bytes	Malandra et al. (2017)
<i>Smart Traffic Lights</i>	500 bytes	Malandra et al. (2017)
<i>Smart Public Transportation</i>	200 bytes	Malandra et al. (2017)
<i>Smart Parking</i>	200 bytes	Malandra et al. (2017)
<i>Smart Public Safety</i>	2000 bytes	Malandra et al. (2017)
<i>Smart Cam - Surveillance</i>	94 ~ 327 bytes	Sivanathan et al. (2017)
<i>Air Quality Sensor</i>	234 bytes	Sivanathan et al. (2017)
<i>Smart Healthcare</i>	144 ~ 234 bytes	Sivanathan et al. (2017)
<i>Smart Lightninging</i>	94 bytes	Sivanathan et al. (2017)
<i>Smart Water Meter</i>	12 ~ 1600 bytes	Oberascher, Rauch e Sitzenfrei (2022)

Fonte: Autor.

Segundo o trabalho de Saxena e Raychoudhury (2019), *Smart Healthcare* é caracterizado pela captura de sinais de saúde de um paciente e pelo envio de alertas aos familiares do paciente e aos profissionais da saúde para uma assistência remota personalizada. As aplicações presentes no trabalho de Malandra et al. (2017) foram identificadas com base na análise dos dados trafegados na região da ilha de Montreal, Canadá. Para determinar os tipos de aplicações (*Smart Cam - Surveillance*, *Air Quality Sensor*, *Smart Healthcare* e *Smart Lightninging*) e os tamanhos de pacotes de cada uma delas citadas pelos autores Sivanathan et al. (2017), foram utilizados cinco *clusters* cujos aplicam o algoritmo de *K-Means Clustering* para classificação dos dispositivos que são de IoT e que não são. Em Oberascher, Rauch e Sitzenfrei (2022), foi realizado um estudo sobre a medição da drenagem e da distribuição de água nas instalações centralizadas de tratamento de água no meio urbano. O volume de dados depende da variação das medições, ou melhor, depende da variação do volume de água aferidos em intervalos pequenos (10 s, 1 min, 5 min, 10 min) entre as medidas.

Neste trabalho, o parâmetro de tamanho de pacotes foi adotado como forma de representar estas aplicações. Nesta dissertação, três tamanhos de pacotes (32 bytes, 256 bytes e 1024 bytes) foram escolhidos baseados em pacotes pequenos, médios e grandes na análise feita da pesquisa de Silva et al. (2018) como forma de representar a heterogeneidade das aplicações IoT.

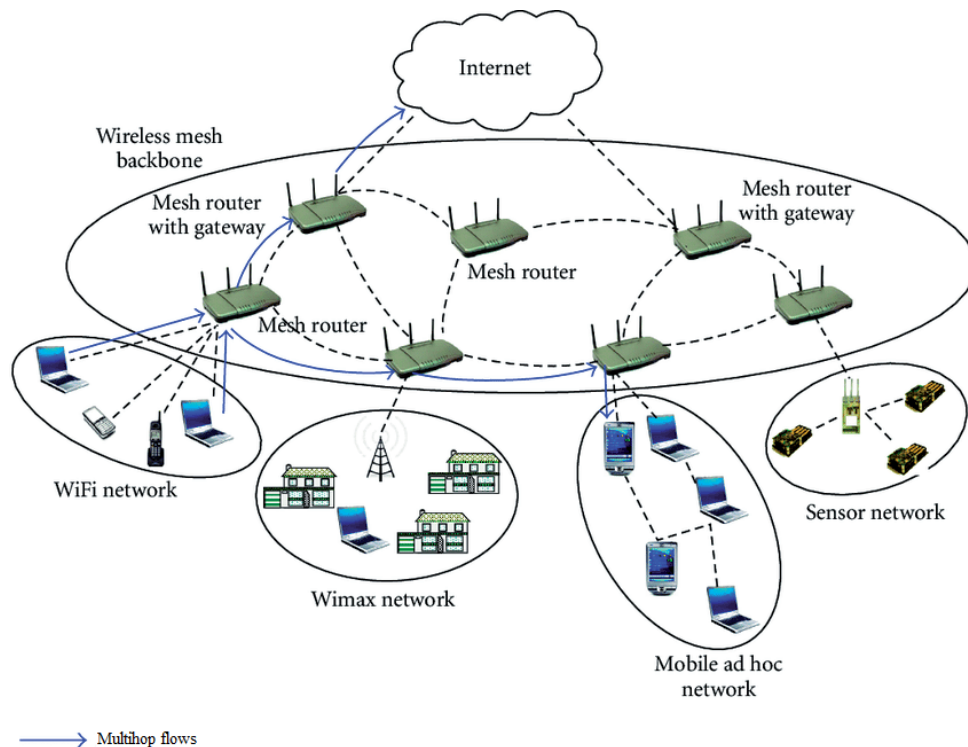
2.3 Redes em Malha Sem Fio

2.3.1 Conceitos Básicos do IEEE 802.11s

O padrão IEEE 802.11s objetiva, entre outros serviços das redes *mesh*, a seleção de caminhos, o encaminhamento com competências de roteamento na camada MAC, a interoperabilidade com redes externas e a segurança das informações (IEEE..., 2011).

De maneira geral, conforme ilustrado na WMN da Figura 5, pode-se dizer que as WMNs são compostas por um conjunto de roteadores sem fio chamados roteadores *mesh* (*mesh router* na figura), que são interligados entre si, formando um *backbone* (*wireless mesh backbone* na figura), para difusão *broadcast* e entrega de pacotes ao longo da topologia de múltiplos saltos.

Figura 5 – Exemplo de Rede em Malha sem Fio.



Fonte: Parvin (2019)

Uma das grandes particularidades dos roteadores *mesh* é que eles são dinamicamente auto-organizáveis, auto-configuráveis e auto-curáveis. Segundo Bayazeed, Khorzom e Aljnidi (2021), auto-organizáveis pode ser definida como uma automatização de redes que não necessitam de intervenção humana para o seu funcionamento. Isto é, realiza, de forma automática, a configuração, o diagnóstico de falhas e a reparação de eventuais dispositivos conectados ou não conectados. Auto-configuráveis é a capacidade de realizar conexões de forma *Plug-and-Play* para as estações bases. E, auto-curáveis é a capacidade de detectar e

diagnosticar falhas para recuperar o estado de funcionamento da rede. Em outras palavras, os roteadores não necessitam de nenhum controle centralizado na rede. Assim, essas propriedades permitem que uma WMN seja implementada incrementalmente, ou seja, um nó de cada vez. Inclusive, a WMN pode ser incrementada sob demanda.

Como ilustra a Figura 5, é possível observar a possibilidade de ampliar o acesso à rede com fio além do alcance de transmissão de um único ponto de acesso (*Access Point* - AP). Nota-se que os roteadores *mesh* são interligados através do meio sem fio com a finalidade de estabelecer e manter a conectividade da “malha” formada. Eles encaminham, através de múltiplos saltos, o tráfego recebido das estações, bem como o tráfego recebido de outros roteadores *mesh*. Roteadores *mesh* também podem exercer o papel de *gateway/bridge*. No mais, a WMN permite que os usuários de redes cabeadas e/ou sem fio (como as redes localizadas na parte inferior da Figura 5) tenham acesso a uma grande área de cobertura (a “malha”).

2.3.2 Roteamento em Redes *Mesh* IEEE 802.11s

O padrão IEEE 802.11s não restringe nenhum protocolo de roteamento para ser implementado por necessidades de aplicações especiais, mas recomenda o uso do protocolo *Hybrid Wireless Mesh Protocol* (HWMP) (IEEE. . . , 2011; ANDREEV; BOYKO, 2010). De acordo com Bahr (2006), a base do protocolo HWMP é uma melhoria do protocolo *Ad hoc On-demand Distance Vector* (AODV) chamado de *Radio Metric-AODV* (RM-AODV).

A métrica de tempo de transmissão (*Airtime Link Metric* - ALM) é utilizada pelo protocolo HWMP para medir a qualidade do caminho, podendo, posteriormente, selecionar o melhor caminho. A Equação 2.1 apresenta a fórmula da métrica ALM:

$$c_a = \left[O + \frac{B_t}{r} \right] \frac{1}{1 - e_f}, \quad (2.1)$$

tal que O e B_t são constantes de sobrecarga de acesso à canais e tamanho em bits do pacote de teste, respectivamente. A constante de sobrecarga de acesso à canais varia conforme a camada física definida, neste trabalho, as simulações foram feitas na camada física IEEE 802.11a cujo valor é 75 *us*. O valor padrão recomendado pelo padrão IEEE do tamanho de bits do pacote de teste é de 8192. A variável r é a atual taxa de bits (*bitrate*) do pacote e o parâmetro $1/(1 - e_f)$ define a fidedignidade da conexão. O resultado final da métrica (c_a) é expresso em unidades de tempo de 0,01, cujo valor é igual a 10,24 *us* (Paszkievicz, Andrzej; Zapala, Przemyslaw, 2018).

Há dois modos que o HWMP pode atuar: modo reativo, baseado em caminhos por demanda; e modo proativo, baseado em árvore. Para seleção de caminho entre roteadores *mesh*, ambos os modos trabalham com o envio de três tipos de mensagens: solicitação de caminho (*Path Request* - PREQ); resposta de caminho (*Path Reply* - PREP); e caminho com

erro (*Path Error* - PERR). Em cada uma dessas mensagens, com exceção das mensagens PERR, há campos importantes como número de sequência (*Destination Sequence Number* - DSN), que permite a comparação entre um caminho atual de um caminho obsoleto, e como tempo de vida (*Time-To-Live* - TTL). Ambos os campos visam evitar o problema de contagem infinita (ou *loop* infinito) (WANG; LIM, 2008). A sobrecarga dessas mensagens de controle, no desempenho do HWMP, é avaliada detalhadamente no presente trabalho para os dois modos de operação.

Em geral, a estratégia reativa é usada para encontrar caminhos entre os roteadores *mesh*, já a estratégia proativa é usada para descobrir rotas entre os roteadores *mesh* e o *gateway*. Em cidades inteligentes, a estratégia proativa é recomendada pela presença do *gateway*, que permite o acesso dos nós internos à Internet.

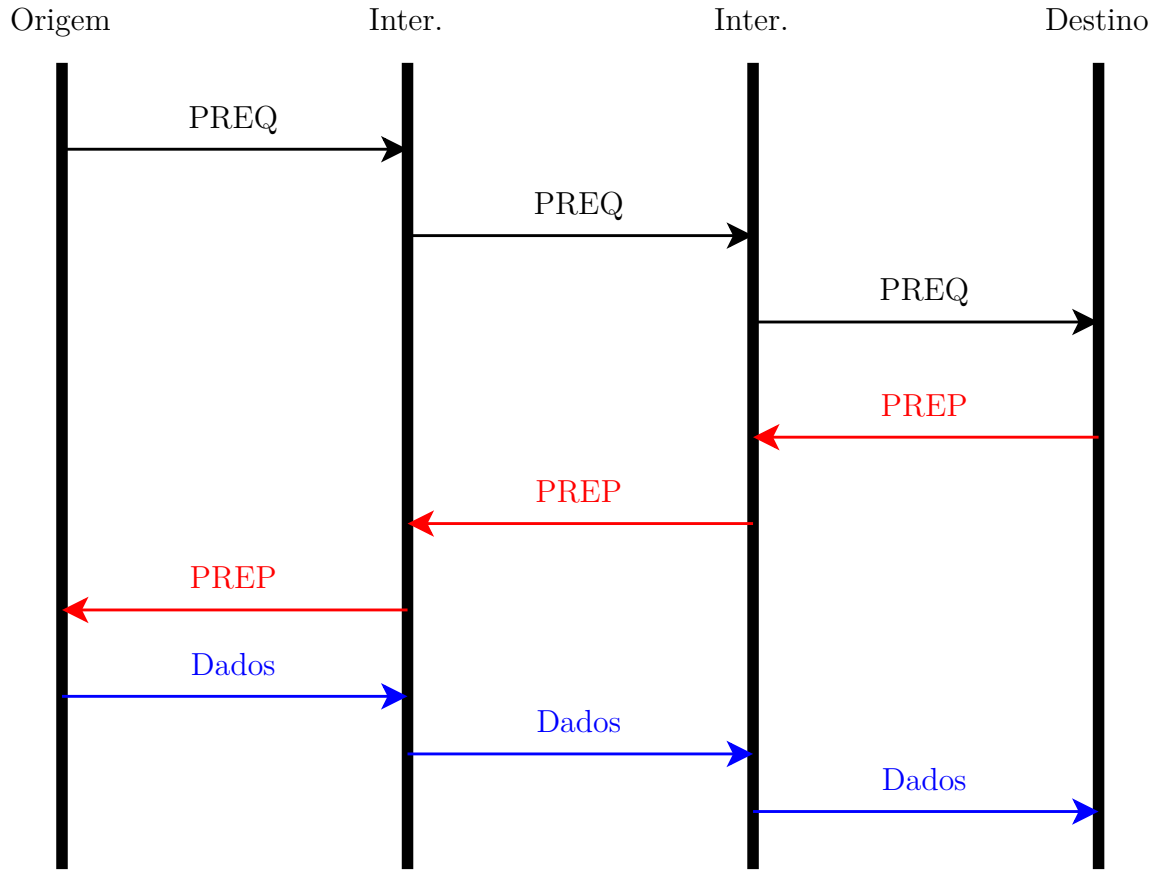
O modo sob demanda (reativo) concede aos roteadores *mesh* a comunicação entre si usando caminhos ponta a ponta. Desde já, é importante deixar claro que, independente da rede possuir ou não um *mesh Gateway* (nó raiz) configurado, o padrão IEEE 802.11s define que o modo reativo sempre estará disponível. Por isso, conforme será apresentado nos resultados do Capítulo 5, o modo reativo (denominado de R nos gráficos) é experimentado isoladamente, mas também combinado com o modo proativo (denominado de RP nos gráficos). Antes de enviar os pacotes de dados no modo reativo, o nó de origem envia uma mensagem *broadcast* PREQ com o intuito de descobrir caminhos até o nó destino. A cada salto, entre os nós intermediários, uma rota é criada/atualizada ou quando o número de sequência da mensagem PREQ seja menor comparada ao anterior, ou quando o número de sequência seja o mesmo, porém, com a métrica ALM melhor. Caso o nó intermediário não conheça um caminho até o nó destino, ele apenas encaminha a mensagem PREQ para os próximos nós. Senão, há três cenários diferentes que pode acontecer conforme as configurações de *flags* de somente destino (*Destination Only* - DO) e de resposta-encaminhamento (*Reply-Forward* - RF), que são detalhados abaixo. Considerando que o nó intermediário (ou coletor) não conheça o caminho, quando o nó coletor recebe a mensagem, ele envia de volta uma mensagem PREP para o nó origem. Ao receber a resposta PREP, o nó origem toma conhecimento de todos os caminhos existentes na malha e escolhe qual caminho será mais vantajoso (DUGAEV; SIEMENS, 2014).

Os três cenários que podem acontecer quando o nó intermediário conhece a rota até o nó destino, são:

1. DO sendo 1 (d1-r0), somente o destino pode responder a mensagem PREQ, ou seja, o nó intermediário só poderá encaminhar a mensagem PREQ para os próximos nós até chegar ao nó destino. Quando o nó destino recebe a mensagem PREQ, ele envia para o nó origem uma mensagem PREP *unicast* através dos nós intermediários, que atualizam suas rotas ao receberem a mensagem PREP. Nesse cenário, representado pela Figura 6, a configuração da *flag* RF não interfere neste cenário, pois apenas o

nó destino poderá responder com a mensagem PREP.

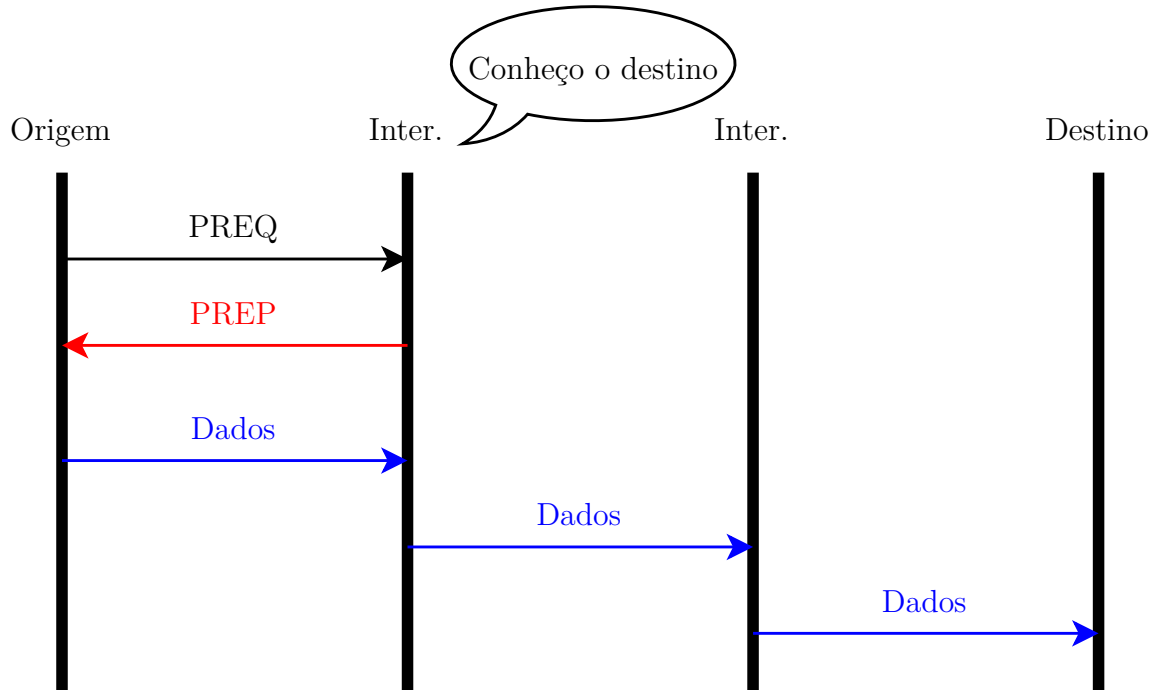
Figura 6 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d1-r0.



Fonte: Adaptada de Andreev e Boyko (2010)

2. DO sendo 0 e RF sendo 0 (d0-r0), representa um cenário em que os nós intermediários poderão responder ao nó origem, através dos caminhos existentes, com a mensagem PREP, mas não poderão reencaminhar as mensagens PREQ para os demais. Neste cenário, os caminhos não são atualizados, pois utilizam os caminhos conhecidos na tabela de roteamento. A Figura 7 ilustra o fluxo deste cenário. Neste exemplo, supõe-se que o primeiro nó intermediário conhece um caminho para o destino.
3. DO sendo 0 e RF sendo 1 (d0-r1), significa que, além do destino, os nós intermediários também poderão responder ao nó origem uma mensagem *unicast* PREP. Além disso, os nós intermediários podem realizar o encaminhamento de mensagens PREQ para os nós intermediários seguintes. Na Figura 6 abaixo, é mostrada o fluxo de mensagens de controle de roteamento desse cenário. Este cenário é definido como o padrão no simulador utilizado neste trabalho, o NS-3.

Figura 7 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d0-r0.



Fonte: Adaptada de Andreev e Boyko (2010)

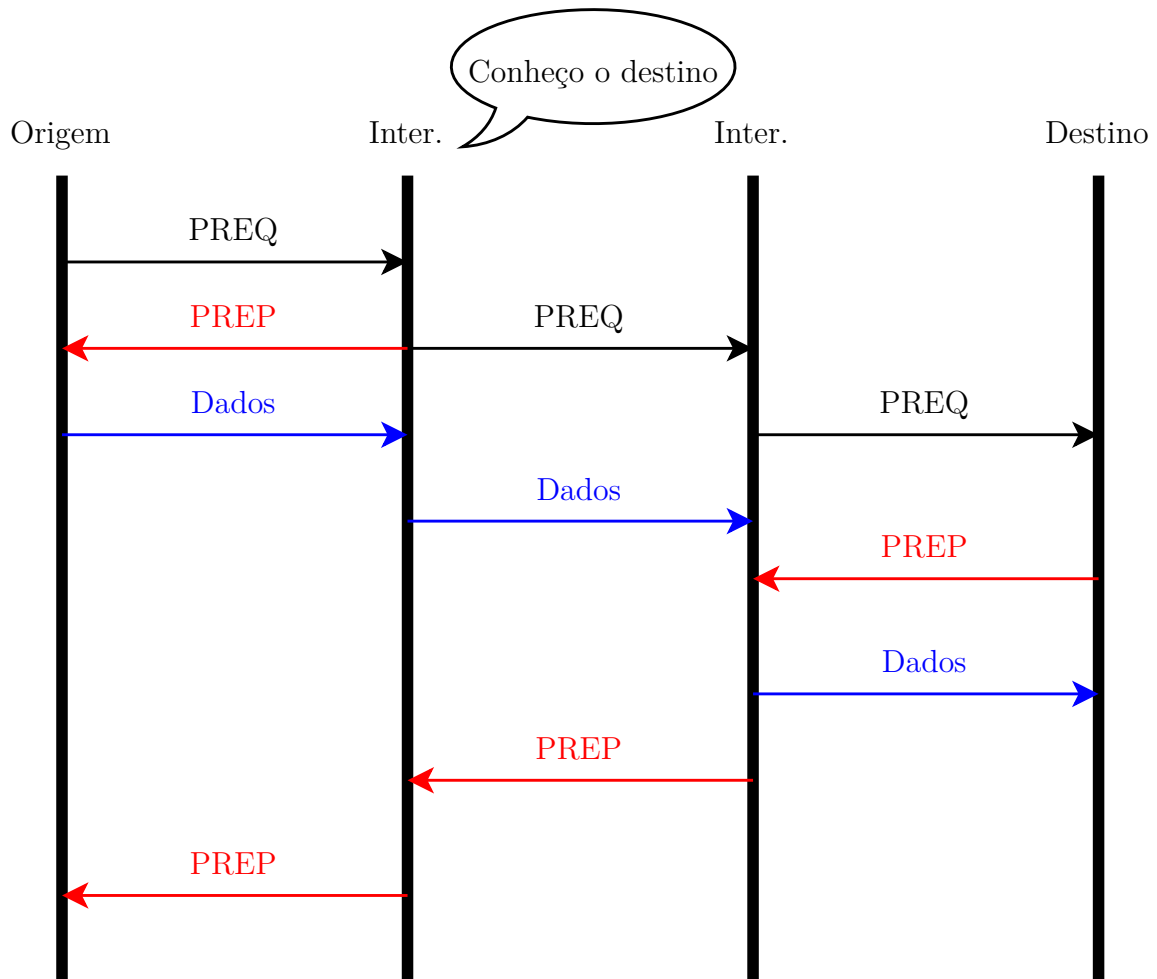
Uma diferença entre o cenário 2 e o cenário 3 é a possibilidade que um nó intermediário possui de transmitir mensagens PREP para o nó origem ao receber uma mensagem PREQ.

O modo proativo mantém um caminho em árvore a partir de um nó nomeado como raiz (*root* ou *mesh Gateway*). Determinado o nó raiz, o padrão define dois mecanismos para disseminar, proativamente, informações de seleção de caminho para alcançar a raiz da malha.

O primeiro mecanismo é denominado PREQ Proativo e cria caminhos através do *broadcast* de mensagens PREQ entre os roteadores mesh e a raiz (*root*) formando uma árvore de caminhos. Logo, todos os nós conhecem o caminho até a raiz e a raiz conhece os caminhos até eles. Esse mecanismo foi utilizado nos experimentos dessa dissertação.

O segundo mecanismo é o anúncio da raiz (*Root ANNouncement* - RANN), que o nó raiz, periodicamente, inunda a rede com mensagens RANN. Quando cada roteador da malha recebe uma mensagem RANN, ele cria/atualiza caminhos até a raiz, e envia mensagens *unicast* PREQ para a raiz. Ao receber a mensagem PREQ, a raiz responde com uma mensagem PREP para os roteadores. A mensagem *unicast* PREQ forma uma rota reversa da raiz ao nó origem enquanto a mensagem *unicast* PREP cria uma rota direta do nó origem ao nó raiz. Cada roteador da malha usa uma comunicação de reconhecimento (*acknowledged communication*) (WANG; LIM, 2008; YANG; MA; MIAO, 2009).

Figura 8 – Fluxo das mensagens de roteamento no mecanismo reativo do HWMP para o cenário d0-r1.



Fonte: Adaptada de Andreev e Boyko (2010)

Observa-se que os modos proativos sob demanda e baseados em árvore não são exclusivos. Eles podem ser usados simultaneamente, uma vez que o modo proativo baseado em árvore é uma extensão do modo sob demanda. As mensagens de PERR podem ser usadas por qualquer um dos modos HWMP disponíveis para anunciar um ou mais destinos inacessíveis. O anúncio é enviado a todas as fontes de tráfego que têm um caminho ativo para o(s) destino(s). O PERR alcançará as fontes correspondentes, que iniciarão um novo ciclo de descoberta de caminho para encontrar um caminho alternativo.

2.4 Simulador de redes NS-3

Neste trabalho, é utilizado o simulador de redes Network Simulator 3 (NS-3) para a realização dos experimentos da análise de desempenho, ou seja, para reproduzir cenários de cidades inteligentes com um grande volume de pacotes sendo trafegados pela rede de malha

sem fio. Assim, nesta seção, algumas informações básicas do simulador são apresentadas.

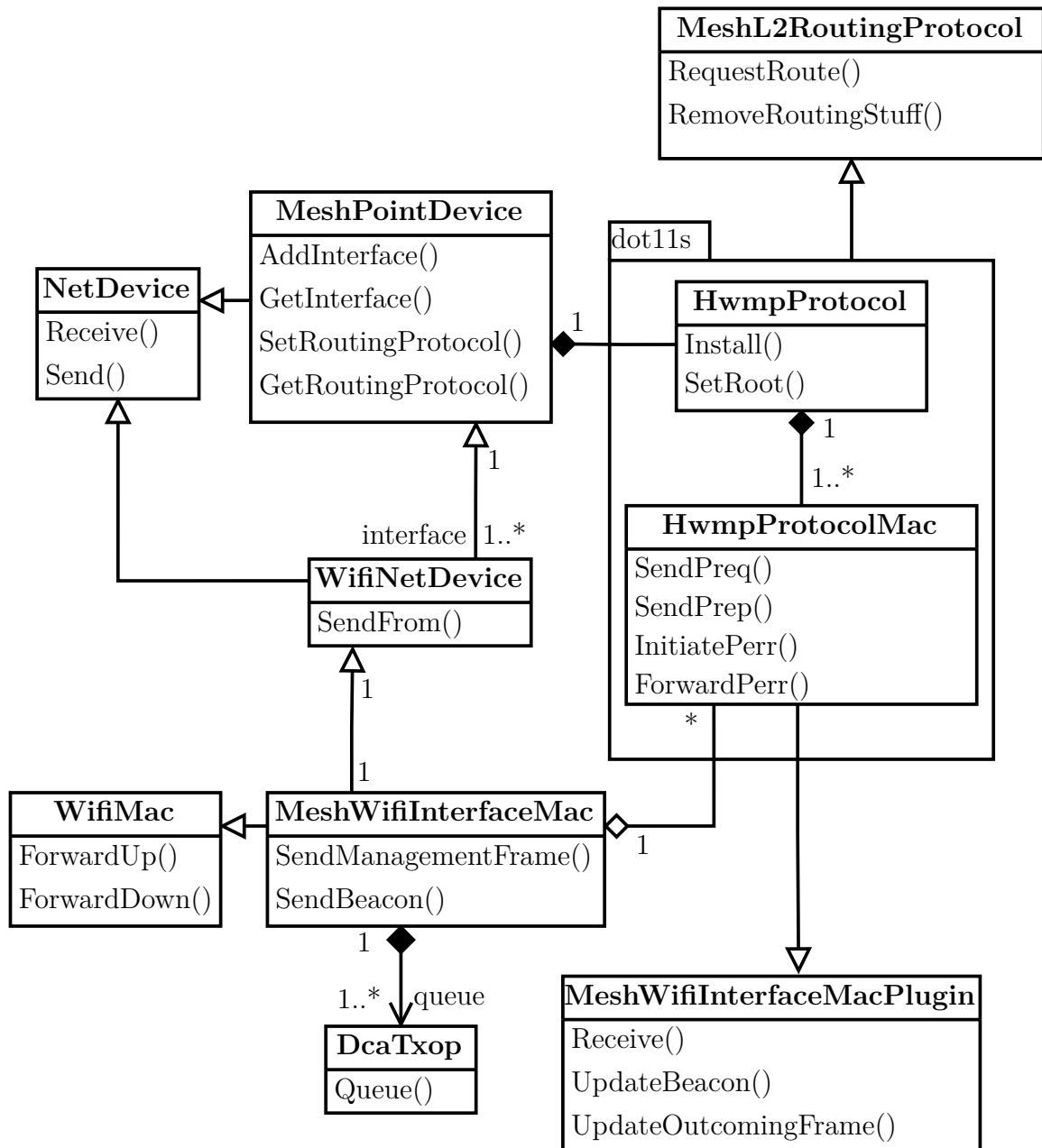
O NS-3 é um simulador de rede de eventos discretos que é direcionado, principalmente, para uso educacional e para pesquisa científica. É um software livre feito por e para pesquisadores. O simulador está disponível publicamente para pesquisa, desenvolvimento e uso desde 2006, e conta com as contribuições da comunidade para o desenvolvimento de novos modelos, assim como para corrigir erros ou manter códigos e compartilhar os resultados. Espera-se neste trabalho disponibilizar os códigos gerados para o projeto NS-3.

O NS-3 foi desenvolvido, puramente, na linguagem de programação C++. A arquitetura do NS-3 foi elaborada de forma semelhante à arquitetura Linux, especificamente em interface de rede, em *drivers* de dispositivos e em *sockets*. Todos os modelos de simulações e elementos de redes presentes no simulador compõe a rede de computadores, e.g., dispositivos de rede (roteador, *switch*, celular, etc.) representam dispositivos físicos que se conectam a um nó por meio de um canal de comunicação (ODA et al., 2016).

No mundo real da tecnologia WiFi, um AP se comunica com um roteador através de um canal de rádio frequência para transmissão de informações trocadas entre eles, e entre outras configurações de rede (endereçamento *Internet Protocol* - IP, MAC, etc.). No NS-3, esses dispositivos são chamados, grosseiramente, de *NetDevices*. Além disso, o NS-3 possui módulos *helpers* que ajudam na implementação, no caso deste trabalho, da topologia *mesh*. O módulo *mesh* (Figura 9, que segue o padrão IEEE 802.11s, automatiza toda a parte de configuração dos nós, que, sem ele, a configuração seria manual. Esse módulo é utilizado para as simulações deste trabalho. No entanto, algumas alterações foram necessárias para atender aos objetivos desta dissertação.

A Figura 9 representa o diagrama de classes do módulo *mesh* compreendido pelas classes *NetDevice*, *MeshPointDevice*, *WifiNetDevice*, *MeshL2RoutingProtocol*, *HwmpProtocol*, *HwmpProtocolMac*, *MeshWifiInterfaceMacPlugin*, *MeshWifiInterfaceMac*, *WifiMac*, *DcaTxop*. As classes *NetDevice* e *WifiNetDevice* representam o controle de envio e de recebimento de dados pelo nó *mesh*. Em *MeshPointDevice*, é solicitada a configuração do protocolo de roteamento realizada pelo pacote *dot11s* que, por sua vez, faz o gerenciamento das mensagens de controle de roteamento. As demais classes têm como objetivo verificar a presença de dispositivos *mesh* na rede, bem como encaminhar os quadros para outras interfaces do padrão IEEE 802.11.

Outro módulo que foi utilizado é denominado de *Flow Monitor*. Ele permite observar dados estatísticos através dos fluxos de pacotes entre os nós (CARNEIRO; FORTUNA; RICARDO, 2009). Os resultados dos dados estatísticos disponibilizados pelo *Flow Monitor* podem ser exportados para um arquivo no formato XML contendo diversas métricas. Dentre essas métricas, a vazão agregada (*Aggregate Throughput*), a taxa de entrega (*Delivery Rate*), o atraso médio (*Delay Mean*), soma de todas as mensagens de controle na rede, o total de mensagens na rede, foram utilizadas para os resultados apresentados neste

Figura 9 – Diagrama de classe do módulo *mesh* do NS-3.

Fonte: Adaptada de Andreev e Boyko (2010)

documento. Na Seção 4.4, as definições dessas métricas são detalhadas.

3 Trabalhos Relacionados

Este capítulo apresenta trabalhos que realizam avaliação de desempenho no contexto de redes em malha sem fio. Os trabalhos são apresentados de maneira cronológica.

Os autores de (IBRAHIM et al., 2013) compararam o protocolo HWMP com o protocolo de roteamento reativo *Ad hoc On-demand Distance Vector* (AODV) em uma rede em malha sem fio. Para efeitos de comparação entre os protocolos de roteamento, foi utilizado o simulador NS-3. Os parâmetros taxa de entrega de pacotes, *throughput* e o atraso fim-a-fim foram considerados como métricas para avaliar quais dos protocolos é melhor. Os resultados mostram que o HWMP é mais apropriado para WMN. Apesar da avaliação de desempenho considerar diferentes números de nós, a quantidade de nós utilizada é baixa (9, 16, 25 e 36 nós). Nesta dissertação, empregou-se 71 e 81 nós para análise do desempenho do HWMP com o objetivo de saturar a rede *mesh* para representar um cenário de cidade inteligente. Outras quantidades de nós (41, 61, 101, 121, 141) também foram testadas, mas sem êxito nos resultados para um mesmo tamanho fixo de raio 300 m. Mais especificamente, para uma rede *mesh* abaixo de 71 nós, verificou-se que nem todos os nós conseguem ter alcance entre os seus vizinhos, como resultado, formando uma rede desconexa. Já para uma rede *mesh* acima de 81 nós, observou-se que os pacotes são transmitidos, mas não são recebidos em virtude de possíveis colisões entre pacotes, interferências intra e inter fluxos, etc. Além da quantidade de nós na rede *mesh*, também, avaliou-se os diferentes tipos de mensagens de controle de roteamento, bem como os modos (reativo, reativo com proativo) do HWMP.

O artigo (MOZUMDER; ACHARJEE; ROY, 2014) cita que um dos requisitos de uma cidade inteligente é ter uma rede sem fio escalável que garanta a acessibilidade e a interoperabilidade entre diversas aplicações IoT. Para isso, o trabalho realiza uma avaliação de desempenho comparando o protocolo HWMP com o protocolo proativo *Better Approach To Mobile Ad-hoc Networking* (BATMAN) com o propósito de explorar o desempenho de ambos em uma rede em malha sem fio. O simulador NS-3 é utilizado nos experimentos, que considera cenários com 60 nós, com mobilidade bidimensional, em uma topologia em grade com uma área de 4 km². Além de avaliar a taxa de entrega, *throughput* e atraso, o *overhead*, causado pelas mensagens de roteamento, também é analisado. Nos experimentos, o protocolo HWMP apresentou uma perda de pacotes de roteamento maior que o BATMAN. Os autores citam que uma provável razão para tal comportamento é o fato do BATMAN realizar a troca de informações de roteamento somente entre vizinhos e o HWMP entre diversos nós da malha. O artigo aponta que estratégias para alocação de recursos de rede e roteamento de pacotes de controle e dados precisam ser desenvolvidas para fornecer alta qualidade de serviço com latência razoável para o maior número de

usuários simultaneamente. Nesta dissertação, diferencia-se do trabalho de Mozumder, Acharjee e Roy (2014) através da topologia e da avaliação por modos, mecanismos e tipos de mensagens em que o protocolo HWMP trabalha. Diferentemente da topologia em grade utilizada no trabalho de Mozumder, Acharjee e Roy (2014), a topologia em malha, com distribuição aleatória dos nós, foi utilizada para avaliação do protocolo HWMP nesta dissertação. Além disso, no trabalho de Mozumder, Acharjee e Roy (2014), não se deixa claro quais os modos do HWMP foram analisados e quais mensagens de controle de roteamento foram observadas, pois, apenas, apresenta uma visão genérica da transmissão de mensagens de roteamento.

No trabalho (NATARAJU; MAHESHAPPA; DEVKATTE, 2016), uma abordagem semelhante aos trabalhos anteriores, realiza experimentos utilizando o simulador NS-3, mas realiza a análise variando o tamanho de topologias em uma grade (com 9, 16, 25, 36 e 49 nós), o tamanho de pacotes (de 0,5k a 10k bytes) e o número de interfaces (1 a 4 interfaces), buscando encontrar o tamanho de pacote ideal para explorar melhores resultados das redes em malha sem fio. A estratégia de múltiplas interfaces de roteadores só se mostraram eficazes, ou melhor, foram melhores na taxa de transferência, no atraso fim-a-fim e na entrega de pacotes quando o tamanho de pacote é limitado a 2k bytes. O trabalho mostrou que, caso a área de cobertura (topologia) exigida por uma aplicação seja pequena, o aumento do número de interfaces, com o intuito de obter melhores resultados na taxa de transferência, torna-se irrelevante. Semelhantemente aos outros trabalhos, a pesquisa de Nataraju, Maheshappa e Devkatte (2016) não aborda em detalhes os modos e as mensagens de controle do HWMP. Conforme esta dissertação, a topologia em malha, a variação de fluxos, as mensagens de controle propagadas na rede, foram diferenciais em comparação à pesquisa de (NATARAJU; MAHESHAPPA; DEVKATTE, 2016).

O trabalho (NAYYAR, 2018) realiza uma avaliação comparativa dos protocolos HWMP, AODV, *Destination-Sequenced Distance Vector* (DSDV), *Dynamic Source Routing* (DSR) e *Optimized Link State Routing* (OLSR), em um cenário de *Flying Ad hoc Networks* (FANETs). Uma FANET é uma rede ad hoc caracterizada pela formação de Veículos Aéreos Não Tripulados (*Unmanned Aerial Vehicles* - UAV). Os parâmetros de simulação foram os seguintes: tempo de simulação de 1500 segundos, área de simulação 800 m x 800 m, 20 nós e tamanho de pacote de 512 bytes. Os resultados mostram que o HWMP supera em desempenho todos os outros protocolos em termos de taxa de entrega, atraso fim-a-fim e *throughput*. Em Nayyar (2018), realizou-se um cenário diferente do proposto por esta dissertação (cidades inteligentes), variando apenas a taxa de transmissão dos pacotes. Parâmetros como os tamanhos de pacotes, os modos de operação, as quantidades de mensagens de controle de roteamento e as diferentes combinações das *flags* de roteamento do HWMP foram avaliados, apenas, nesta dissertação.

O trabalho de (SILVA et al., 2018) avalia o desempenho do HWMP em um cenário

de cidades inteligentes. O NS-3 também é utilizado para realização dos experimentos. Estes são realizados com 81 nós, 8 diferentes tamanhos de pacotes (de 16 bytes a 1024 bytes) e variações na saturação da rede. Os resultados mostram, que em ambientes com maior saturação, a taxa de entrega do HWMP cai entre 20% e 30% à medida que a quantidade de fluxos aumenta, havendo em paralelo uma elevação do número de mensagens PREQs. O trabalho de Silva *et al.* é o mais similar a presente dissertação, mas as principais diferenças da presente proposta são: avaliar, comparativamente, os modos reativo e proativo do HWMP, os tipos de mensagem de controle do HWMP (PREQ, PREP e PERR) e as métricas de avaliação conforme a estratégia de roteamento definidas pelas *flags* de controle de roteamento somente destino (*Destination Only* - DO) e encaminhamento e resposta (*Reply and Forward* - RF).

No trabalho de (CHETHAN; BASAVARAJU, 2019), um melhoramento do HWMP é proposto, sendo chamado de *Scalable Local Route Repair*-HWMP (SLRR-HWMP). O simulador Qualnet é utilizado para comparar o HWMP com o SLRR-HWMP. Os experimentos são realizados com 5 diferentes quantidades de nós (de 16 a 32 nós) e um tamanho de pacote (512 bytes). Os resultados mostram que a técnica de reparo de rotas locais, proposta pelo SLRR-HWMP, considera apenas os nós vizinhos para reparar uma rota localmente e, conseqüentemente, minimiza a sobrecarga, o tempo e o custo de controle dos caminhos. O SLRR-HWMP é melhor avaliado que o HWMP em termos de sobrecarga, atraso e *throughput*. O trabalho de (CHETHAN; BASAVARAJU, 2019) fomentou esta dissertação para uma análise das mensagens de controle de roteamento, de acordo com as combinações das *flags* de roteamento DO e RF, dos modos do HWMP, com o intuito de documentar, de forma analítica, o impacto desses parâmetros no desempenho do protocolo HWMP.

Na pesquisa (GAUTAM; MAHAJAN; ZAFAR, 2020), um estudo da segurança do HWMP, AODV e *Secure AODV* (SAODV) é feito para verificar qual protocolo resiste melhor aos ataques *Distributed Deny of Services* (DDoS) em um ambiente *healthcare*. O simulador *Network Simulator-2* (NS-2) foi utilizado para realização de experimentos com 5 diferentes quantidades de nós (de 30 a 150 nós) e com um tamanho de pacote de 1024 bytes. Os resultados mostram que o HWMP é superior ao AODV e ao SAODV, sendo o preferido, dessa forma, em situações em que dados sensíveis precisam ser transmitidos com o mínimo possível de perda de pacotes, durante a transmissão, e em que o envio das informações, como os sintomas do paciente, precisa ser trafegado em regiões remotas. Apesar das análises referente à perda de pacotes, o trabalho de Gautam, Mahajan e Zafar (2020) não analisa a sobrecarga das mensagens de roteamento, o modo de operação e a combinação das *flags* de roteamento do HWMP.

A Tabela 2 apresenta um comparativo dos trabalhos relacionados. A tabela está dividida em 6 colunas cujos parâmetros trabalho, protocolo, área, quantidade de nós,

tamanho de pacotes e métricas foram analisadas. As sete linhas representam os trabalhos avaliados, inclusive esta dissertação, conforme os parâmetros. Percebe-se que, em nenhum dos trabalhos relacionados, houve a análise das mensagens de controle de roteamento e das diferentes combinações das *flags* de controle de roteamento (DO e RF). Portanto, este trabalho buscou explorar melhor as configurações do protocolo HWMP.

Tabela 2 – Comparativo de trabalhos

Trabalho	Protocolo	Área (m ²)	Quant. Nós	Tam. Pacote (bytes)	Métricas
IBRAHIM et al., 2013	AODV, HWMP	160 k, 360 k, 640 k, 1 M, 144 M	9, 16, 25, 36	N/A	vazão, taxa de entrega, atraso fim-a-fim
MOZUMDER et al., 2014	HWMP, BATMAN	4 k	5, 10, 20, 30, 40, 50, 60	128	taxa de entrega, vazão, atraso fim-a-fim
NATARAJU et al., 2016	HWMP	28,9 k	9, 16, 25, 36, 49	0.5 k, 1k, 1.5 k, 2k, ... 10 k	taxa de entrega, vazão, atraso fim-a-fim
NAYAR et al., 2018	AODV, DSDV, DSR, OLSR, AOMDV, HWMP	640 k	5, 10, 20, 30, 40, 50, 60	512	taxa de entrega, vazão, atraso fim-a-fim
SILVA et al., 2018	HWMP	282,7 k	81	16, 32, 64, 128, 256, 512, 750 1024	taxa de entrega, vazão, atraso fim-a-fim, <i>jitter</i> PREQ Iniciadas
CHEETHAN et al., 2018	HWMP, SLRR-HWMP	2,25 M	16, 20, 24 28, 32	512	sobrecarga, vazão, atraso fim-a-fim
GAUTAM et al., 2020	AODV, SAODV, HWMP	1,51 M	30, 60, 90 120, 150	1024	taxa de entrega, atraso fim-a-fim
Este trabalho	HWMP-R, HWMP-RP	282,7 k	81	32, 256, 1024	vazão agregada, taxa de entrega, atraso médio, $\sum$ msgs. controle, total msgs. PREQ, PREP, PERR na rede

Fonte: Autor.

4 Avaliação de Desempenho

Neste capítulo, a metodologia utilizada para elaboração da proposta é apresentada em 5 etapas. A Etapa 1 (Seção 4.1) é o entendimento do domínio de estudo. A Etapa 2 (Seção 4.2) consiste nos detalhes de preparação do *script* desenvolvido para automação da submissão das simulações no NS-3. A Etapa 3 (Seção 4.3) apresenta os detalhes das simulações propriamente ditas. A Etapa 4 (Seção 4.4) apresenta os *scripts* desenvolvidos para elaboração dos resultados (gráficos amigáveis) das simulações. Por último, a Etapa 5 (Seção 4.5) trata da análise dos resultados obtidos.

4.1 Etapa 1: Entendimento do domínio de estudo

A primeira etapa da pesquisa consistiu no entendimento do domínio de estudo, i.e., do problema e da definição dos objetivos a serem alcançados neste trabalho. Pode-se equiparar essa primeira etapa aos estudos realizados para a escrita da Fundamentação Teórica (Capítulo 2) e dos Trabalhos Relacionados (Capítulo 3).

Entre os principais tópicos de estudo realizados nesta etapa, destacam-se: redes em malha sem fio, padrão IEEE 802.11s (com foco no roteamento usando o protocolo HWMP), cidades inteligentes, IoT, simulador NS-3, entre outros.

4.2 Etapa 2: Preparação do *script* de entrada para as simulações no NS-3

Antes de realizar as simulações propriamente ditas, foi pensado em desenvolver um *script* para automatizar o processo de submissão das simulações no NS-3.

Para exemplificar as inúmeras vantagens do uso do *script* desenvolvido antes da etapa de simulação, tem-se a Tabela 3 que mostra os parâmetros utilizados como entrada para as simulações já realizadas neste trabalho. Percebe-se que alguns desses parâmetros foram fixados em todas as análises, são eles: raio, número de rodadas de simulação, número de *gateways*, etc. Já outros parâmetros podem assumir valores distintos em cada rodada de simulação, como é o caso do: número de roteadores, número de fluxos de pacotes, tamanho de pacotes e nó raiz.

Assim, um dos aspectos que o *script* desenvolvido, nesta etapa, otimiza é a submissão automática da combinação de todos os valores da Tabela 3. Por exemplo, ao executar o *script* em questão, diversas entradas são dadas ao simulador NS-3, cada uma sendo uma

combinação diferente de parâmetros. Essa característica torna o processo de simulação mais rápido e reduz a chance de uma combinação da tabela de parâmetro não ser avaliada.

Além disso, as três combinações das *flags* de somente destino (*Destination Only* - DO) e de resposta e encaminhamento (*Reply and Forward* - RF) foram avaliadas, refletindo, em suma, os três cenários vistos: apenas o nó destino responde as mensagens PREQ (d1-r0), o nó vizinho ao nó origem responde a mensagem PREQ baseado nas rotas existentes em sua tabela de roteamento (d0-r0) e, o último caso, qualquer nó responde as mensagens PREQ (d0-r1), conforme visto na Seção 2.3.2. Apenas a combinação $DO = 1$ e $RF = 1$ foi ignorada, pois a *flag* RF não tem nenhuma ação quando a *flag* DO está habilitada, i.e., só quem poderá responder a solicitação de caminho será o nó destino. A avaliação dessas *flags* ajuda a compreender em quais situações há uma sobrecarga das mensagens de controle sendo propagadas na rede.

Tabela 3 – Parâmetros de Entrada para as Simulações.

Parâmetro	Descrição /Valor(es)
Interface PHY	IEEE 802.11a
Área para formação da topologia	Circular
Raio da área	300 metros
Tempo total de simulação	30 segundos
Número de rodadas de simulações	80
Número de roteadores <i>mesh</i> (nós)	71 e 81
Número de <i>gateways</i> na topologia	1
Número de fluxos	1, 5, 10, 15, 20, 25, 30, 40, 50, 60
Intervalo entre pacotes para cada fluxo	0,01 s
Tamanho do pacote	32, 256, 1024 bytes
Nó raiz configurado	não (sem proativo)/sim (com proativo)
<i>Flags</i> DO e RF	DO:0 e RF:0 DO:0 e RF:1 DO:1 e RF:0

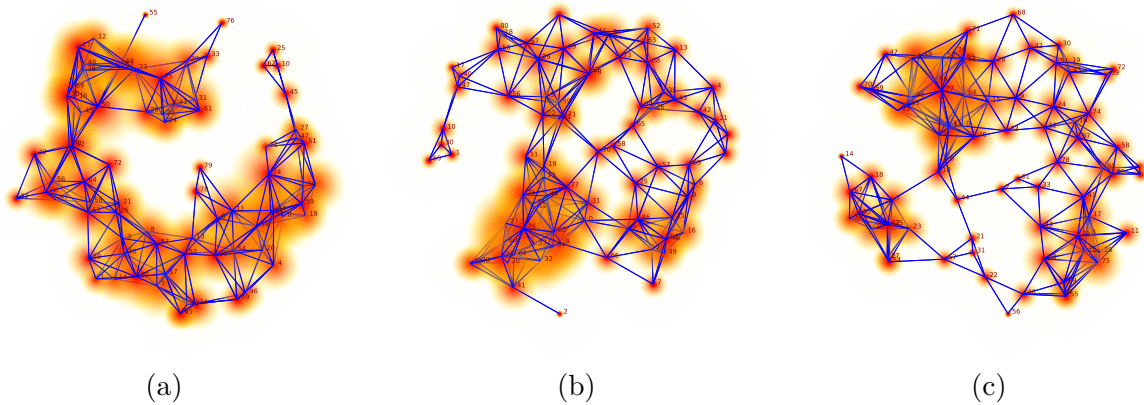
Fonte: Autor.

Um segundo aspecto relevante do *script* criado é em relação às topologias utilizadas para as simulações. Toda topologia usada na avaliação é gerada de forma aleatória pelo simulador de acordo com: a área estabelecida e o número de roteadores *mesh* que devem ser dispostos de maneira uniforme na área em questão.

A Figura 10 mostra exemplos de topologias *mesh* geradas pelo NS-3 a partir desses parâmetros de entrada. Percebe-se que as três topologias (Figuras 10a, 10b e 10c) foram geradas de forma distinta pelo *script*, porém com características em comum: topologia em malha. Cada ponto possui um número, que são os roteadores *mesh*, e as arestas azuis, que

representam um enlace entre dois nós. No caso das topologias da Figura 10, observa-se também que existe pelo menos um caminho entre quaisquer dois nós da rede, ou seja, não há nenhum nó isolado. Assim, ao realizar a simulação, pode-se garantir que a taxa de entrega de pacotes e o atraso de entrega obtidos com a simulação não são afetados pela não existência de um caminho.

Figura 10 – Exemplos de Topologias *mesh* com 81 nós em um círculo de raio de 300m.



Fonte: Autor

Seguindo esse raciocínio, antes do uso de uma determinada topologia no processo de simulação, o *script* verifica se a topologia está conectada, ou seja, se existe um caminho entre quaisquer dois nós da rede. Para tanto, é utilizada a biblioteca Python NetworkX¹.

Se pelo menos um nó estiver desconectado, a topologia é descartada e uma nova é gerada para ser avaliada pelo *script*. No mais, caso o *script* não encontre topologias conectadas por três vezes consecutivas, então, o *script* aborta a execução enviando um alerta. Esse número de tentativas pode ser facilmente alterado.

4.3 Etapa 3: Simulações no NS-3

Neste trabalho, foi utilizado o NS-3 versão 29 para a realização dos experimentos².

Para realizar as inúmeras simulações da avaliação de desempenho (combinação dos parâmetros da Tabela 3), fez-se necessária a utilização de uma máquina virtual Linux (*Standard DS3 v2*) em nuvem na plataforma *Azure*. Cada experimento foi repetido 80 vezes com um intervalo de confiança de 95%. Todos os pré-requisitos de configuração exigidos pelo simulador NS-3 foram atendidos. Além disso, utilizou-se a ferramenta *gitHub*³ para o controle de versionamento dos scripts, códigos C++, códigos Python, logs, etc.

¹ <https://networkx.org/>

² <https://www.nsnam.org/releases/ns-3-29/>

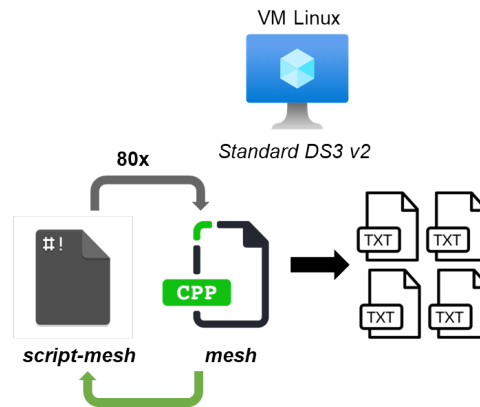
³ <https://github.com/ruigmoreno/ngn-ns3.29>

O algoritmo escrito em C++ faz o uso de funções *helpers*, presentes nas bibliotecas padrões do simulador NS-3, para modelar e simular uma topologia de rede mesh sem fio com a utilização do protocolo HWMP. De maneira resumida, os passos que o algoritmo realiza são:

1. Configuração dos parâmetros de entrada (como os da Tabela 3);
2. Formação do modelo de topologia e instalação da aplicação (UDP servidor e UDP cliente);
3. Inicialização de um monitor dos fluxos (*Flow Monitor*);
4. Execução da simulação;
5. Estatística dos parâmetros a serem analisados gerados no formato texto.

A Figura 11 ilustra um fluxograma representado pela execução da simulação baseada nos parâmetros de entrada. Percebe-se que o *script* roda 80 vezes sobre o arquivo *mesh* que, por sua vez, gera os arquivos estatísticos.

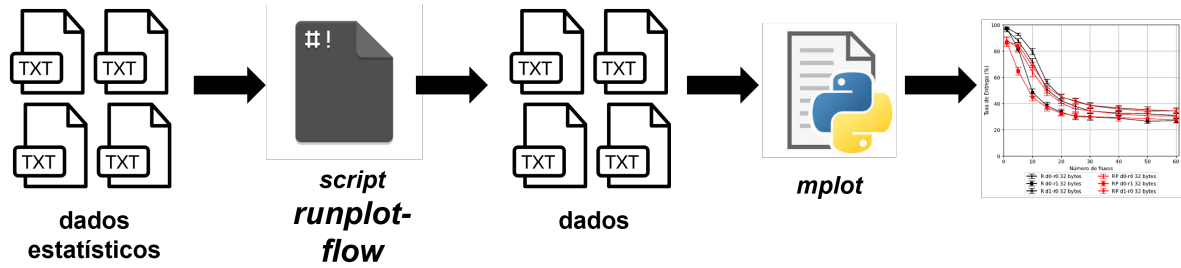
Figura 11 – Fluxograma do *script mesh*.



Fonte: Autor

A Figura 12, apresenta, de maneira resumida, os passos seguintes após a extração dos dados estatísticos. Primeiramente, esses dados são importados no *script runplot-flow*, do qual são extraídos dados no formato em que a biblioteca *matplotlib lib* consiga ler e, consequentemente, exibir os gráficos afim de ser possível a análise analítica.

Figura 12 – Fluxograma do script runplot-flows



Fonte: Autor

4.4 Etapa 4: Preparação de um *script* para exibição dos resultados das simulações em gráficos amigáveis

Para a preparação dos resultados, foi estruturado um *script*⁴ com base nos parâmetros de entrada da Tabela 3.

Para a exibição de gráficos, é necessário fazer o uso de bibliotecas que possuem o recurso de plotagem. Uma delas é a conhecida biblioteca python chamada matplotlib, que, segundo Hunter (2007), é amplamente utilizada pela comunidade científica para plotagens de gráficos 2D e 3D, de imagens e de expressões matemáticas. Ela faz o uso, também, de outras bibliotecas como a NumPy. Nela, há uma poderosa *Application Programming Interface* (API) nas mais diversas aplicações científicas para manipulações de vetores e de matrizes de N-dimensões, armazenando em memória todo tipo de manipulação vetorial de N-dimensões em uma única dimensão sem perder a originalidade da dimensão (HARRIS et al., 2020).

As métricas de avaliação utilizadas neste trabalho foram:

- A taxa de entrega (*Delivery Rate*) é a quantidade total de pacotes que saindo do nó fonte chegam ao nó destino com sucesso, medida em porcentagem (MOULEMA et al., 2014).
- Atraso Médio (*Delay Mean*) é a média de tempo que pacotes levam ao serem transmitidos de um nó origem a um nó destino. Os atrasos mais importantes a serem mensurados são o atraso de processamento nodal (atraso que o roteador leva para ler o cabeçalho do pacote para transferir para o destinatário), atraso da fila, atraso da transmissão e atraso da propagação. Acumulando esses atrasos tem-se o atraso total (MOULEMA et al., 2014).
- Vazão Agregada (*Aggregate Throughput*) é definida como o somatório de todos os *throughputs*, que, por sua vez, é a quantidade de bits recebidos dividido pela

⁴ <https://github.com/ruigmoreno/ngn-ns3.29/blob/main/runplot-flows.sh>

diferença entre o tempo de chegada do primeiro pacote e o tempo de chegada do último pacote, unidade de medida sendo megabits por segundo (Mbits/s ou Mbps) (IBRAHIM et al., 2013).

- Somatório de mensagens de controle na rede é a soma do total de cada uma das mensagens de controle na rede: PREQ, PREP e PERR.
- Total de mensagens de controle de roteamento na rede PREQ.
- Total de mensagens de controle de roteamento na rede PREP.
- Total de mensagens de controle de roteamento na rede PERR.

4.5 Etapa 5: Análise dos resultados das simulações

Essa etapa final consiste na análise dos gráficos obtidos na Etapa 4. No próximo capítulo são apresentados resultados das simulações realizadas.

5 Resultados

Este capítulo apresenta os resultados obtidos das simulações realizadas conforme a combinação de todos os parâmetros definidos na Tabela 3 da Etapa 2 (Seção 4.2 da proposta). As seções do capítulo estão divididas de acordo com as métricas de avaliação de desempenho de uma rede (conforme citado na Seção 4.4 da proposta).

Em cada seção, são apresentados gráficos com cada um exibindo três tamanhos de pacotes diferentes (32 KB, 256 KB e 1024 KB) para 71 e 81 nós. Em todos eles, o eixo das abscissas representa o número de fluxos (de 1 a 60 fluxos) e o eixo das ordenadas representa uma das métricas supracitadas. Além disso, o intervalo de tempo entre pacotes é de 0,01 segundos. As cores em vermelho e preta estão relacionadas ao modo reativo (R) e reativo com proativo (RP), respectivamente. As curvas nos gráficos correspondem aos três tipos de cenários vistos na Seção 2.3.2, são eles: apenas nó destino responde às mensagens PREQ (d1-r0), nó vizinho a origem responde a mensagem PREQ baseado nas rotas existentes em sua tabela de roteamento (d0-r0) e, o último caso, qualquer nó responde às mensagens PREQ (d0-r1). Os cenários (ou combinações) d0-r0, d0-r1 e d1-r0 foram representados, respectivamente, por marcadores: ponto, quadrado e x.

5.1 Taxa de Entrega

Em todos os gráficos desta seção (Figuras 13 e 14), pode-se perceber que há uma degradação na taxa de entrega de pacotes quando há o aumento do número de fluxos (eixo x dos gráficos), tanto para o cenário com 71 nós, quanto para 81 nós.

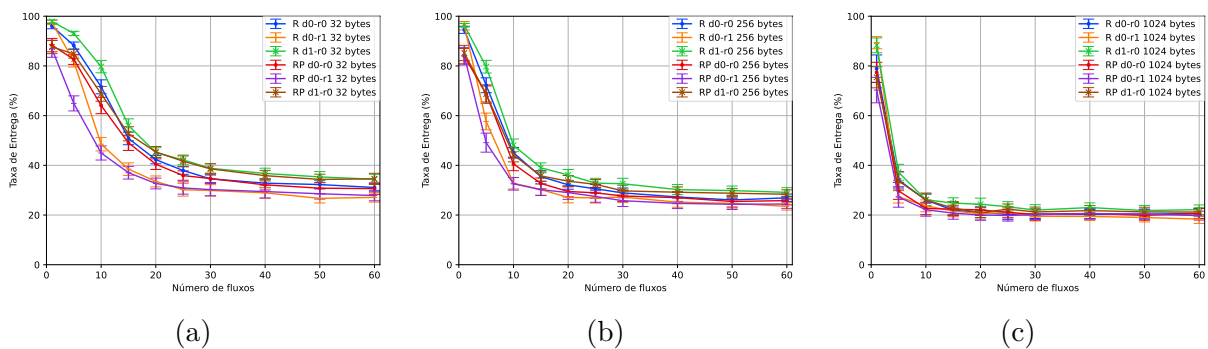


Figura 13 – Resultado da métrica taxa de entrega para 71 nós.

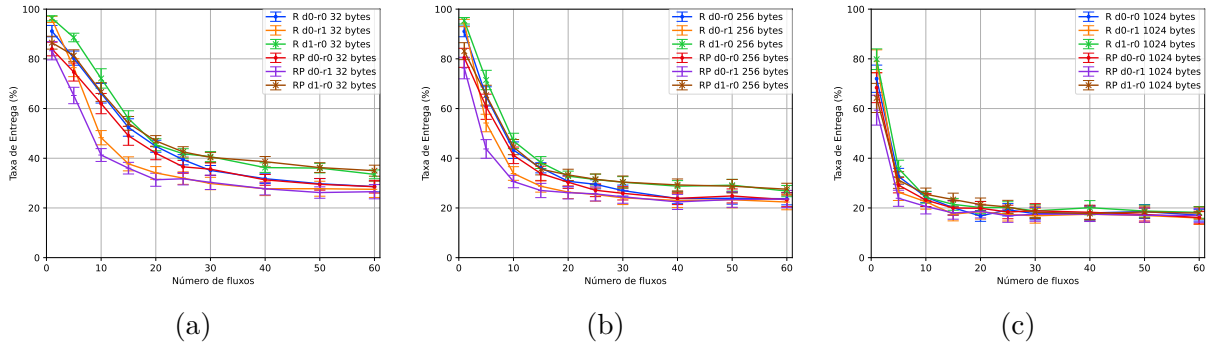


Figura 14 – Resultado da métrica taxa de entrega para 81 nós.

Além disso, é evidente que pacotes maiores (1024B) sofrem mais drasticamente e rapidamente uma queda da taxa de entrega que pacotes menores (32B) com esse aumento do número de fluxos. Pode-se observar também que, em cenários com 81 nós, a taxa de entrega sofre uma degradação maior que com 71 nós.

Para facilitar a compreensão dos gráficos dessa seção e das próximas, é importante lembrar o conceito do tempo de transmissão de um quadro na rede. Em poucas palavras, pode-se considerar que o tempo para transmitir um quadro na rede é influenciado pela soma de dois fatores: o tempo para um nó obter uma oportunidade de transmissão sem fio e o tempo de transmissão do pacote em si. Por isso, quanto mais intensa é a competição pelo acesso ao meio (por exemplo: uma rede mais densa e com mais fluxos sendo injetados na rede) e maior o tamanho do pacote a ser transmitido, maior será esse tempo para transmitir um quadro na rede, conforme será mostrado em maiores detalhes na Seção 5.2, que trata do atraso. Além disso, também é importante lembrar que a probabilidade de ocorrer um erro de transmissão aumenta quando pacotes maiores são transmitidos. Em resumo, quando se tratam de cenários de cidades inteligentes com múltiplos fluxos de dados e com tamanhos de pacotes maiores, é esperado que esses cenários cheguem mais rapidamente a um estado de saturação da rede que cenários com poucos fluxos e pacotes de tamanho menores, conforme refletem os resultados das Figuras 13 e 14.

Um outro fator importante a ser observado no comportamento dos gráficos está relacionado ao número de mensagens de controle de roteamento (PREQ, PREP e PERR) gerado pelos seis casos relacionados aos modos de operação do HWMP e às combinações de flags experimentados nesta dissertação. Sabe-se que mensagens de controle de um protocolo de roteamento podem aumentar com o número de fluxos injetados na rede, no entanto, não é desejável que essas mensagens utilizem demasiadas oportunidades de transmissão porque isso significa que essas oportunidades deixarão de ser utilizadas para o envio dos dados das aplicações em si. Os detalhes do impacto dessas mensagens de controle para cada modo de operação e tipo de mensagem de controle serão apresentados e discutidos nas Seções 5.4 e 5.5.

Um outro fator importante a ser observado, também, é a comparação entre os modos R e RP. O modo R possui uma melhor taxa de entrega que o modo RP quando poucos fluxos são considerados. Como se pode ver na Figura 13a, o modo R, para 1 fluxo, encontra-se próximo de 100% da taxa de entrega e o modo RP próximo de 90%. Ainda na mesma figura, a medida em que o número de fluxos aumenta, as diferenças entre os modos vão diminuindo até se equipararem. A partir de 30 fluxos, percebe-se bem este comportamento, com exceção de 50 fluxos que apresenta uma leve variação no modo R d0-r1.

Nas Figuras 13 e 14, a combinação das *flags* d1-r0 mostra uma maior taxa de entrega, principalmente, para 30 fluxos em diante se comparada com as outras combinações. A combinação de pior desempenho está representada pela combinação de *flags* d0-r1.

5.2 Atraso Médio

As Figuras 15 e 16 representam o atraso médio para 71 nós e para 81 nós, respectivamente. Verifica-se que, conforme o aumento do tamanho de pacotes e do número de fluxos, maior é o atraso médio. Além disso, comparando os dois cenários (71 nós e 81 nós), percebe-se um atraso maior nos gráficos da Figura 16, onde a rede está mais densa (81 nós). Por exemplo, a Figura 16a mostra que, para 10 fluxos, a combinação RP d0-r1 está acima de 0,4 s de atraso médio. Já na Figura 15a, essa mesma combinação se encontra abaixo de 0,4 s. Nota-se, também, que a função do atraso médio cresce mais rapidamente, ou melhor, atinge o seu pico em poucos fluxos para pacotes maiores (1024B) quando comparados a pacotes menores (32B). Após a função atingir o seu valor de pico, o seu valor tende a ficar constante (ou com poucas variações).

Nas Figuras 15a e 16a para 32B, o modo reativo, para fluxos abaixo de 15, apresenta um atraso médio inferior em relação ao modo reativo com proativo. Após 15 fluxos, percebe-se um aumento gradativo do atraso para o modo R, chegando a ser semelhante em atraso com o modo RP em fluxos maiores que 30. Comportamentos similares são vistos nas Figuras 15c e 16c.

Para as comparações entre as *flags* DO e RF, as Figuras 15 e 16 mostram que a combinação d1-r0 é a menor em atraso quando poucos fluxos são considerados, principalmente para pacotes pequenos. Na medida em que o número de fluxos aumenta, as combinações d1-r0 e d0-r1 passam a inverterem de posições no gráfico (Figuras 15b e 16b), isto é, a combinação d1-r0 se torna a função com o maior atraso e a combinação d0-r1 se torna a função com o menor atraso. Ao visualizar o gráfico da Figura 16c, esse comportamento não é o mesmo, pois, as três combinações passam a oscilar entre elas, isto é, qualquer uma das combinações passam a ter um atraso maior.

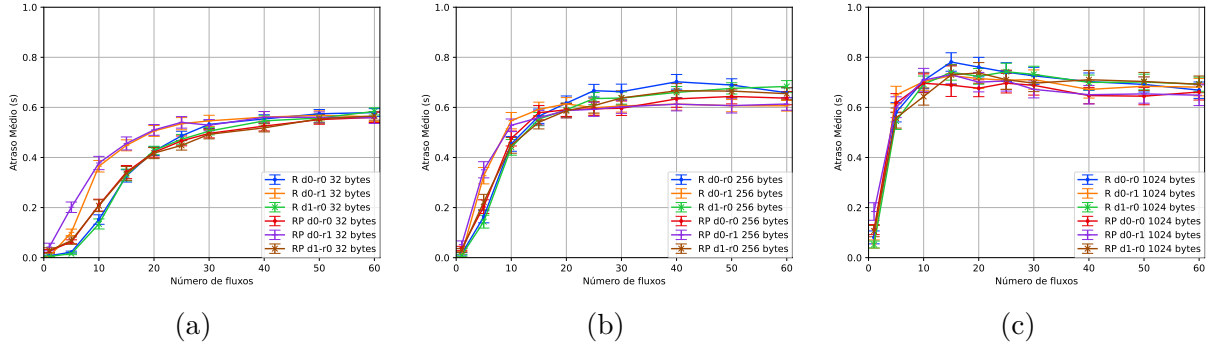


Figura 15 – Resultado da métrica atraso médio para 71 nós.

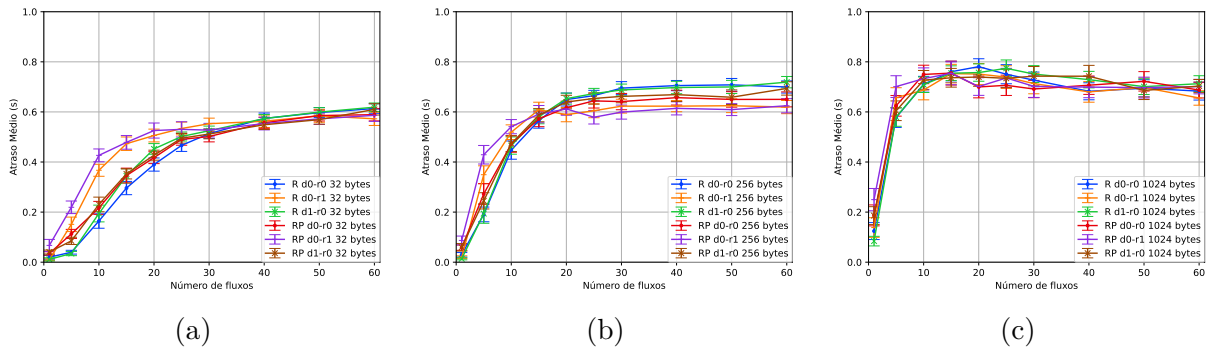


Figura 16 – Resultado da métrica atraso médio para 81 nós.

5.3 Vazão agregada

As Figuras 17 e 18 apresentam a vazão agregada (*Aggregate Throughput*) em função do número de fluxos para 71 e 81 nós, respectivamente.

Inicialmente, analisando e comparando as Figuras 17 e 18, pode-se perceber que a vazão agregada aumenta na medida em que o tamanho do pacote e o número de fluxos aumentam. Assim, tanto para 71 quanto para 81 nós, percebe-se uma maior variação entre as funções para pacotes maiores (1024B) quando comparados aos pacotes menores (32B).

Um comparativo entre as Figuras 17 e 18 também mostra que, apesar da pequena diferença, quanto maior a densidade da rede, menor é a vazão agregada. Este comportamento se reflete para todos os tamanhos de pacotes, mas com diferenças mais significativas para pacotes maiores. Por exemplo, ao analisar as Figuras 17c e 18c para 1 fluxo, a vazão agregada das seis situações (modos de operação do HWMP e combinações de *flags*) está abaixo de 2 Mbit/s para 81 nós, enquanto para 71 nós a vazão dos modos está mais próxima de 2 Mbit/s. Já ao analisar os mesmos gráficos para 60 fluxos, a vazão agregada dos seis modos está perto ou abaixo de 5 Mbit/s para 81 nós, enquanto para 71 nós a vazão agregada da maioria dos modos está perto ou acima de 6 Mbit/s.

Já quanto aos modos de operação do HWMP (R e RP), verifica-se que, de forma geral, a vazão agregada do modo de operação reativo é superior quando comparada ao seu correspondente no modo reativo com proativo, principalmente para pacotes maiores. Para que fique mais claro esse ponto, considerando como exemplo a Figura 17b, percebe-se que a vazão agregada do modo R d1-r0 é notadamente superior ao modo RP d1-r0 já a partir de 5 fluxos e se mantém assim até 60 fluxos. O mesmo ocorre quando se compara os modos R e RP para d0-r0, assim como para os modos R e RP para d0-r1.

Ao se comparar os modos de operação em função das *flags*, a combinação das *flags* d1-r0 resultou em uma maior vazão quando comparada às duas outras combinações. Já para a combinação d0-r1, observa-se a menor vazão entre os resultados, independente do tamanho do pacote e da densidade.

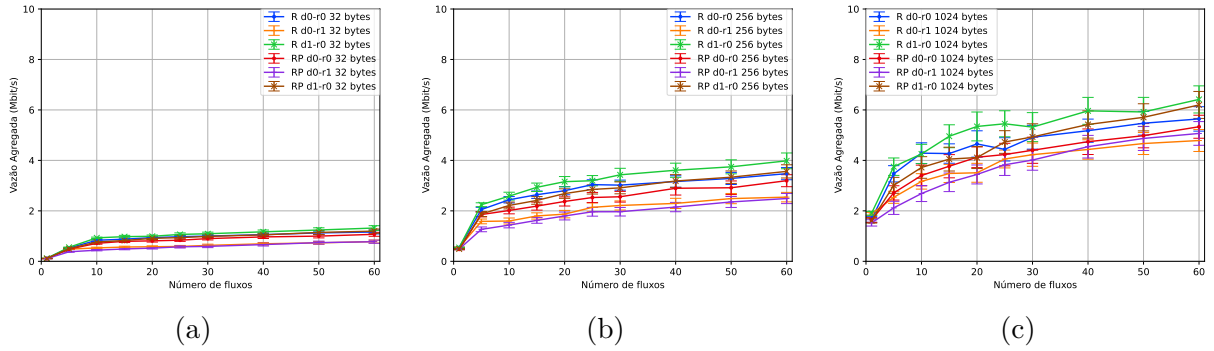


Figura 17 – Resultado da métrica vazão agregada para 71 nós.

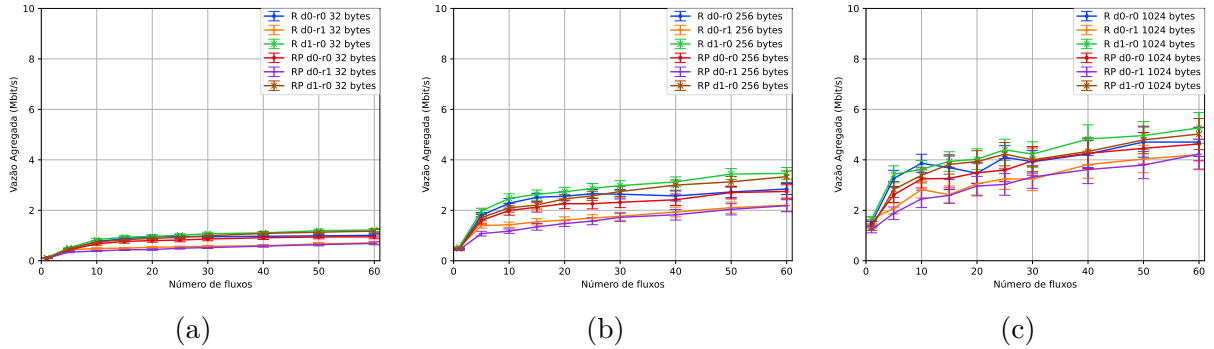


Figura 18 – Resultado da métrica vazão agregada para 81 nós.

5.4 Soma das mensagens de controle na rede

A soma das mensagens de controle na rede corresponde a soma das mensagens PREQ, PREP e PERR do HWMP.

Ao analisar as Figuras 19 e 20, percebe-se que, inicialmente, há uma tendência de aumento do número de mensagens de controle quando aumenta o número de fluxos, porém,

para tamanhos do pacote maiores, o ponto de saturação é alcançado mais rapidamente (com poucos fluxos). Por exemplo, na Figura 19a para 32B, a quantidade de mensagens de controle do modo R d0-r1 permanece praticamente a mesma a partir de 40 fluxos, já na Figura 19c para 1024B, essa quantidade já fica praticamente constante com apenas 20 fluxos.

Conforme os conceitos vistos na Seção 2.3.2, a combinação d0-r1 representa o pior cenário em relação a quantidade de mensagens presentes na rede. Isso porque qualquer um dos nós pode responder e propagar às mensagens de solicitação de caminho, inundando a rede com mensagens PREP e PREQ, respectivamente. A Figura 20a ilustra essa situação, na qual as mensagens do modo R d0-r1 chegam a atingir valores aproximados de 260 mil mensagens a partir de 15 fluxos. Contrariamente a este cenário, a combinação d0-r0 denota a menor quantidade de mensagens de controle, uma vez que não há mensagens sendo propagadas na rede para atualização de rotas, apenas rotas existentes na tabela de roteamento dos nós serão utilizadas para o envio de dados. Ainda na mesma figura, observa-se que a quantidade de mensagens de controle para o modo R d0-r0 chega a atingir, aproximadamente, 60 mil mensagens, uma redução de quase 77% em comparação com a curva representada por R d0-r1. Para o cenário R d1-r0, tem-se um cenário onde as mensagens de solicitação só poderão ser respondidas pelo nó destino. Os nós intermediários apenas encaminharão as mensagens PREQ para os nós subsequentes. Assim sendo, este cenário tem um comportamento intermediário em comparação às combinações d0-r1 e d1-r0.

Em uma outra análise, constatou-se, através das Figuras 20a, 20b, 20c, que as mensagens de controle no modo RP são superiores em relação ao seu semelhante, modo R. Por exemplo, na Figura 20b, o modo RP d0-r0 para 20 fluxos apresenta uma quantidade de, aproximadamente, 200 mil mensagens, enquanto a quantidade de mensagens para o modo R está em torno de 225 mil mensagens. Da mesma forma, esse comportamento se assemelha tanto para a combinação d0-r1 quanto para d1-r0. Isso pode ser explicado pelo fato do modo reativo com proativo trabalhar de forma híbrida no controle de mensagens de roteamento.

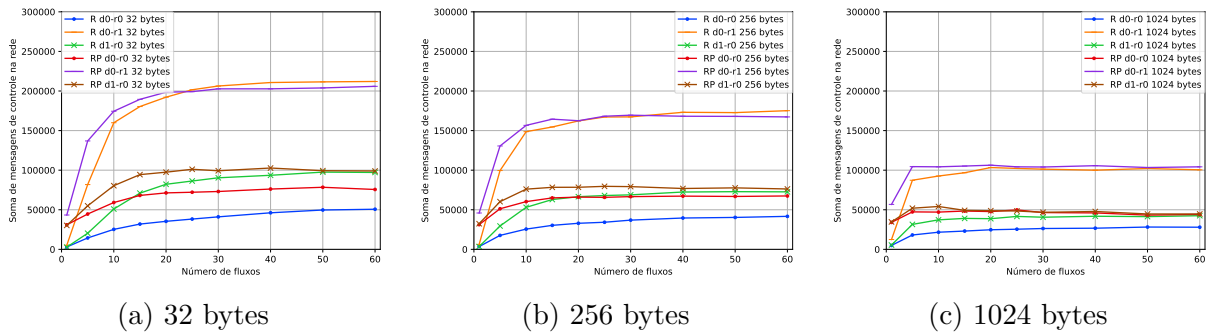


Figura 19 – Resultado da métrica soma de mensagens de controle na rede para 71 nós.

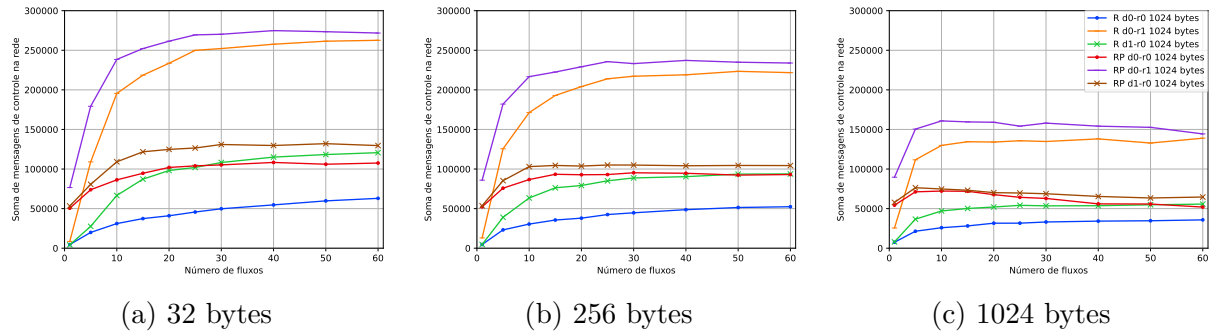


Figura 20 – Resultado da métrica soma de mensagens de controle na rede para 81 nós.

5.5 Total das mensagens de controle na rede

Nesta seção, é abordada de maneira individualizada a quantidade de mensagens PREQ, PREP e PERR que são propagadas na rede.

5.5.1 PREQ

As Figuras 21 e 22 apresentam as mensagens de solicitação de rota (PREQ) na rede com 71 nós e 81 nós, respectivamente, e com a escala do eixo y variando entre 0 e 120 mil mensagens. Nelas, é fácil perceber um crescimento na quantidade total de mensagens PREQ na rede na medida em que o número de fluxos aumenta, o que é esperado, já que mais fluxos resultam em mais demandas de rotas pelo HWMP. Ademais, verifica-se que quanto mais cresce a densidade (número de nós) na rede, maior é a quantidade de mensagens PREQ na rede, já que há uma maior probabilidade de rotas com maior número de saltos na rede com 81 nós.

Comparando as Figuras 21a e 22a, nesta ordem, nota-se que o número de mensagens PREQ para 71 nós no modo R ou RP d1-r0 em 15 fluxos, por exemplo, é próximo de 60 mil mensagens, enquanto para 81 nós estas mensagens ficam a cerca de 70 mil mensagens. Logo, percebe-se um aumento de 10 mil mensagens PREQ nessa análise. Nas outras combinações, principalmente na combinação d0-r0 (modo que manda menos PREQ dentre as três combinações de *flags*), o aumento de mensagens PREQ também ocorre, mas é menor em relação a cenários com poucos fluxos.

Ao olhar isoladamente a Figura 22a, o número de mensagens PREQ para o modo R d1-r0 em 10 fluxos é de, aproximadamente, 60 mil mensagens. Dobrando o número de fluxos, a quantidade de mensagens vai para 80 mil, um aumento de quase 33% em relação à quantidade de mensagens no fluxo anterior.

Comparando as Figuras 21b e 21c, verifica-se que o total de mensagens PREQ na rede, para 60 fluxos, 256B, R ou RP d1-r0, é de cerca de 60 mil mensagens, enquanto o total

de mensagens PREQ na rede, para 60 fluxos, 1024B, R ou RP d1-r0, cai consideravelmente, ficando por volta de 35 mil mensagens. Logo, compreende-se que quanto maior é o tamanho do pacote, menor é a quantidade de mensagens PREQ. O mesmo comportamento se repete para 81 nós. Nesse ponto, é importante apontar para os resultados da taxa de entrega na Seção 5.1, que mostram que a taxa de entrega para pacotes maiores como 1024B cai drasticamente já com cerca de 10 fluxos. Além disso, que o atraso dos pacotes grandes também cresce rapidamente com o aumento de 1 para apenas 5 fluxos, conforme mostrou a Seção 5.2.

Ao analisar o total de mensagens PREQ na rede (na Figura 21 e 22) em relação aos modos R e RP, com exceção da combinação d0-r0, verifica-se que o modo reativo e o modo reativo com proativo possuem comportamentos parecidos, quando comparados em modo e em mesma combinação de *flags*. Para a combinação d0-r0, em pacotes menores (32B), o total de mensagens PREQ no modo RP é um pouco maior que no modo R para fluxos abaixo de 20. Nas Figuras 21a e 22a, consegue-se ver, em 10 fluxos, que o modo R d0-r0 está próximo de 10 mil mensagens, enquanto o modo RP d0-r0 está um pouco acima de 10 mil mensagens. Uma vez que o número de fluxos cresce, as diferenças entre os modos R e RP vão diminuindo para redes contendo 71 nós. Em redes com 81 nós, percebe-se que o modo R supera em número de mensagens o modo RP. Nas Figuras 21 e 22, é possível notar isso principalmente para 60 fluxos.

Em relação às combinações das *flags*, o total de mensagens PREQ para a combinação R d0-r0 (ou RP d0-r0) denota a menor quantidade de mensagens totais PREQ se comparada às outras duas combinações R d0-r1 (ou RP d0-r1) e R d1-r0 (ou RP d1-r0). Por exemplo, comparando a combinação RP d0-r1 com a combinação RP d1-r0 da Figura 22a, percebe-se que, em 60 fluxos, há uma diferença aproximada de 40 mil mensagens. Na sequência, comparando a combinação RP d0-r1 com a combinação RP d0-r0, observa-se que, em 60 fluxos, a diferença é, também, em torno de 40 mil mensagens. Ou seja, para pacotes de 32B, percebe-se uma simetria entre as combinações das *flags* para fluxos altos.

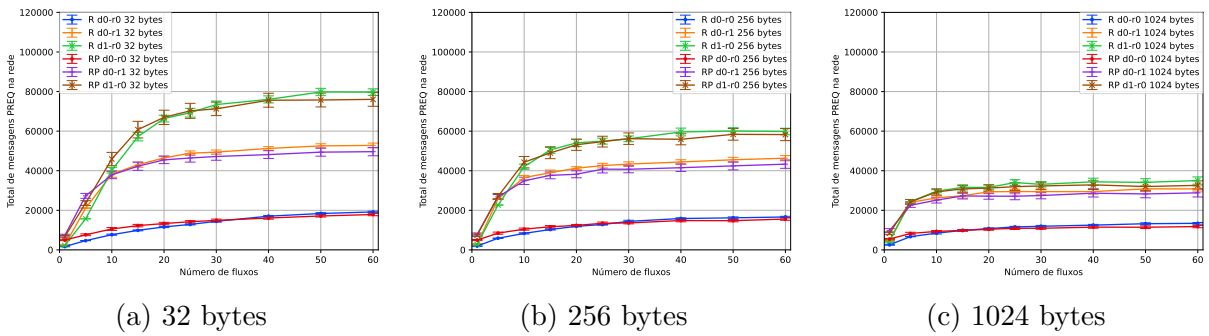


Figura 21 – Resultado da métrica total de mensagens PREQ na rede para 71 nós.

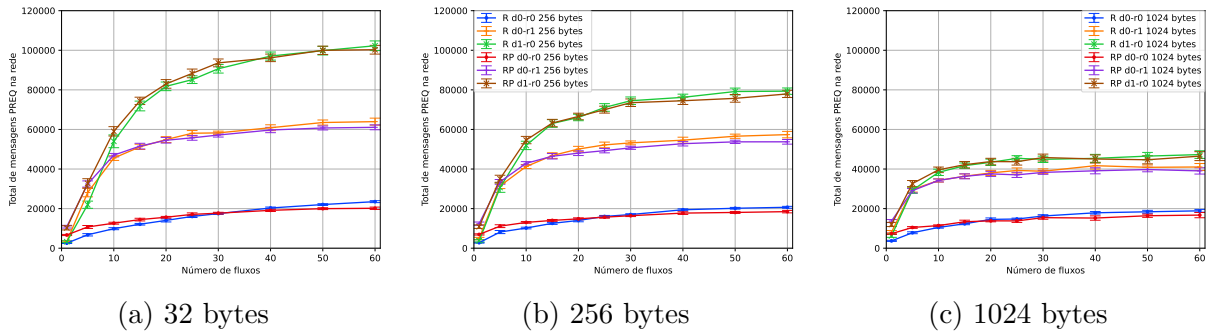


Figura 22 – Resultado da métrica total de mensagens PREQ na rede para 81 nós.

5.5.2 PREP

As Figuras 23 e 24 representam o total de mensagens de resposta (PREP) na rede com 71 nós e 81 nós, nesta ordem, e com a escala do eixo y variando entre 0 e 225 mil mensagens. Fazendo uma comparação entre o valor máximo dessa escala (225 mil) e o valor máximo da escala presente nos gráficos das mensagens PREQ (Figuras 21 e 22, com o valor máximo da escala de 120 mil), nota-se uma diferença de 105 mil mensagens.

De modo geral, percebe-se que, para poucos fluxos (abaixo de 10 fluxos), o número total de mensagens PREP cresce proporcionalmente ao número de fluxos. Além disso, quanto maior o tamanho de pacotes, menor é a quantidade de mensagens PREP transmitidas. Nas Figuras 23a e 23b, observa-se que o total de mensagens PREP para 40 fluxos e para a combinação R d0-r1 é cerca de 150 mil e 125 mil mensagens PREP, respectivamente. Uma diferença de 25 mil mensagens PREP entre os tamanhos de 32B e 256B.

Ao visualizar as Figuras 23 e 24, percebe-se que o modo RP supera, em termos de quantidade de mensagens PREP, o modo R, quando comparados entre as mesmas combinações de *flags* e modos de operação. Na Figura 24b, pode ser demonstrado isso, pois, para 30 fluxos, o número de mensagens PREP no modo R d0-r0 é em torno de 25 mil mensagens, enquanto no modo RP d0-r0 é em torno de 75 mil mensagens. Assim, a quantidade de mensagens PREP para o modo RP chega a ser três vezes mais em relação às mensagens PREP para o modo R.

Comparando o total de mensagens PREP em relação as diferentes combinações de *flags*, tem-se que o número total de mensagens PREP da combinação d0-r1 é superior à combinação d0-r0, que, por sua vez, é superior à combinação d1-r0 quando comparados entre um mesmo modo (R d1-r0 com R d0-r0, R d0-r0 com d1-r0).

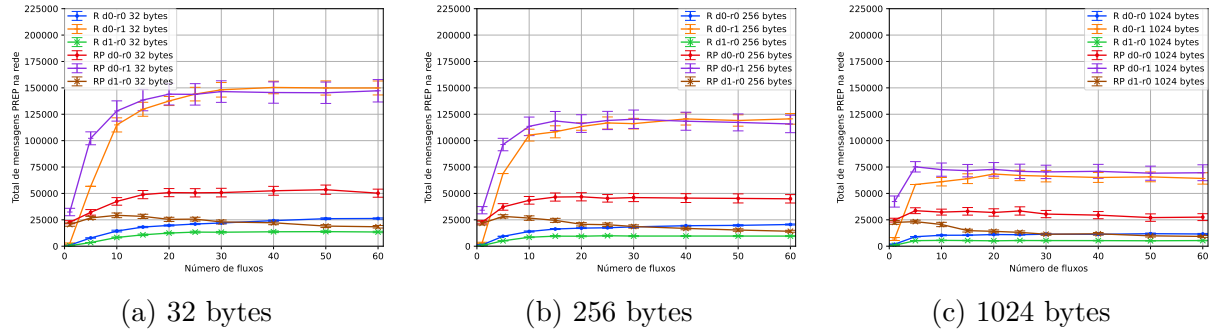


Figura 23 – Resultado da métrica total de mensagens PREP na rede para 71 nós.

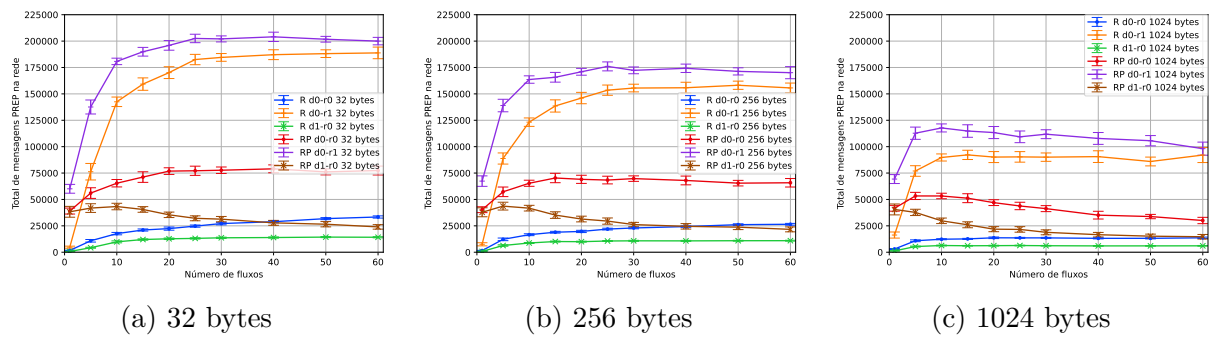


Figura 24 – Resultado da métrica total de mensagens PREP na rede para 81 nós.

5.5.3 PERR

As Figuras 25 e 26 apresentam o total de mensagens de erro (PERR) na rede com 71 nós e 81 nós, na devida ordem, e com a escala do eixo y variando entre 0 e 12 mil mensagens de erro. Uma diferença de 100 mil mensagens em relação à escala dos gráficos das mensagens PREQ (Figuras 21 e 22). Nessas figuras das mensagens de erro, observa-se que, para pacotes de 32B e de 1024B, o número total de mensagens de erro diminuem ao aumentar o tamanho de pacotes.

Ao analisar os modos R (d0-r0 e d0-r1) e RP (d0-r0 e d0-r1) nas Figuras 25a e 26a, percebe-se que, inicialmente, há uma tendência de aumento do número de mensagens PERR quando aumenta o número de fluxos, porém, para tamanhos de pacote maiores, o ponto de saturação é alcançado mais rapidamente (com poucos fluxos). Por exemplo, na Figura 25a para 32B, a quantidade de mensagens PERR do modo R d0-r1 permanece praticamente a mesma a partir de 40 fluxos, já na Figura 25c para 1024B, essa quantidade já fica praticamente constante com apenas 20 fluxos.

No total de mensagens PERR na rede, na maior parte dos casos, o modo RP possui um maior número de mensagens em relação ao modo R se comparados entre as mesmas combinações de *flags*, os mesmos modos e os mesmos tamanhos de pacotes, ou seja, entre R d0-r0 de 32B e RP d0-r0 de 32B, R d0-r1 de 32B e RP d0-r1 de 32B e assim

sucessivamente. Esse mesmo comportamento é tal qual o total de mensagens PREP na rede.

Para as combinações d0-r1 (Figura 26), há uma maior probabilidade de ocorrer erros nas mensagens de roteamento quando as mensagens PERR na rede são comparadas com as duas outras combinações (d0-r0 e d1-r0). Esse aumento de falhas na rede é causado pela sobrecarga de mensagens respondidas em virtude da combinação das *flags*, de acordo com os exemplos da Seção 5.4.

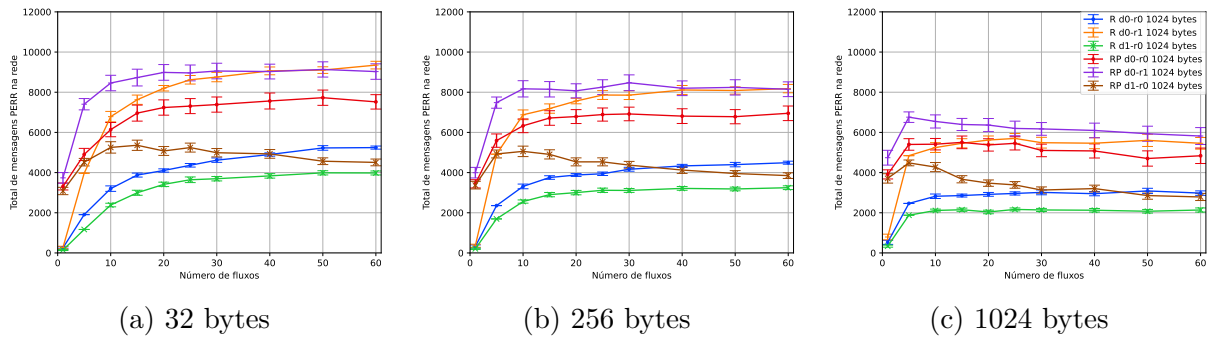


Figura 25 – Resultado da métrica total de mensagens PERR na rede para 71 nós.

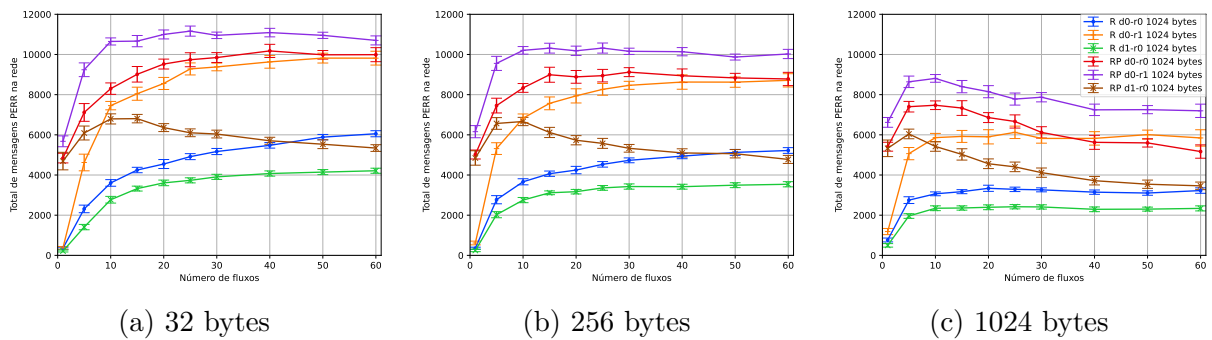


Figura 26 – Resultado da métrica total de mensagens PERR na rede para 81 nós.

6 Conclusão e Trabalhos Futuros

Este trabalho apresentou uma avaliação de desempenho de redes em malha sem fio em cenários de cidades inteligentes. O simulador de software livre NS-3 foi utilizado para realização de experimentos da análise de desempenho das redes *mesh*, e, conseqüentemente, do protocolo de roteamento HWMP. O NS-3 permite realizar simulações do mundo real através dos mais atuais tipos de protocolos de redes.

Diferentemente da maioria dos trabalhos existentes que utilizam o padrão IEEE 802.11s, este trabalho realizou uma interpretação analítica tanto sobre os mecanismos do HWMP quanto sobre as diferentes configurações presentes através de *flags* do HWMP. Nos gráficos avaliados, ficou evidenciado que o aumento da quantidade de fluxos, quantidade de nós e o tamanho de pacotes interferem negativamente no desempenho das redes *mesh*.

Nas análises dos mecanismos de operação do HWMP, observou-se que a quantidade de mensagens de controle de roteamento no modo reativo com o proativo (RP) veio a superar as mensagens de controle no modo reativo. A literatura sugere o uso do modo proativo para cenários onde *gateways* fazem o acesso a redes externas para um controle centralizado das mensagens de roteamento. Porém, conforme os resultados deste trabalho, identificou-se que o desempenho do HWMP no modo reativo com o proativo, em sua maioria, é impactado pela sobrecarga das mensagens de controle de roteamento (PREQ, PREP e PERR), que são manipuladas através das *flags* somente destino (DO) e resposta e encaminhamento (RF).

Ainda mais, três tipos de combinações das *flags* DO e RF foram avaliados. A combinação que menos impactou o desempenho do HWMP foi com a habilitação (1) da flag DO e desabilitação (0) da *flag* RF, ou seja, apenas o nó destino poderá responder às solicitações de caminhos feitas pelo nó origem. A pior configuração dessas *flags* está representada pela desabilitação da *flag* DO e habilitação da *flag* RF, ou seja, qualquer nó poderá responder às solicitações de caminho gerando um alto volume de mensagens de controle de roteamento na rede.

Para trabalhos futuros, a análise do mecanismo RANN e do controle de mensagens de anúncio de raiz ficaram como propostas ao estudo das redes *mesh* no cenário de cidades inteligentes.

Referências

AL-FUQAHA, A. et al. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, v. 17, n. 4, p. 2347–2376, 2015. Citado na página 22.

ALRIKABI, H. T. S.; JASIM, N. A. Design and implementation of smart city applications based on the internet of things. *International Journal of Interactive Mobile Technologies (iJIM)*, v. 15, n. 13, p. pp. 4–15, Jul. 2021. Disponível em: <<https://online-journals.org/index.php/i-jim/article/view/22331>>. Citado na página 16.

ANDREEV, K.; BOYKO, P. Ieee 802.11 s mesh networking ns-3 model. In: *3rd International ICST Conference on Simulation Tools and Techniques*. [s.n.], 2010. v. 43. Disponível em: <<https://www.nsnam.org/workshops/wns3-2010/dot11s.pdf>>. Citado 5 vezes nas páginas 29, 31, 32, 33 e 35.

AVELAR, E. et al. Interoperability issues on heterogeneous wireless communication for smart cities. *Computer Communications*, v. 58, p. 4–15, 2015. ISSN 0140-3664. Special Issue on Networking and Communications for Smart Cities. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0140366414002485>>. Citado 2 vezes nas páginas 25 e 26.

BAHR, M. Proposed routing for ieee 802.11s wlan mesh networks. In: *Proceedings of the 2nd Annual International Workshop on Wireless Internet*. New York, NY, USA: Association for Computing Machinery, 2006. (WICON '06), p. 5–es. ISBN 159593510X. Disponível em: <<https://doi.org/10.1145/1234161.1234166>>. Citado na página 29.

BAYAZEED, A.; KHORZOM, K.; ALJNIDI, M. A survey of self-coordination in self-organizing network. *Computer Networks*, v. 196, p. 108222, 2021. ISSN 1389-1286. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1389128621002681>>. Citado na página 28.

CARNEIRO, G.; FORTUNA, P.; RICARDO, M. Flowmonitor: A network monitoring framework for the network simulator 3 (ns-3). In: *Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009. (VALUETOOLS '09). Disponível em: <<https://doi.org/10.4108/ICST.VALUETOOLS2009.7493>>. Citado na página 34.

CHETHAN, K.; BASAVARAJU, T. Scalable local route repair-hybrid wireless mesh protocol (slrr-hwmp) for ieee 802.11 s. *International Journal of Computer Science Engineering (IJCSE)*, IEEE Press, v. 8, p. 173–184, Sep-Oct 2019. ISSN 2319-7323. Disponível em: <<http://www.ijcse.net/docs/IJCSE19-08-05-012.pdf>>. Citado na página 38.

DENG, D.-J.; CHEN, K.-C.; CHENG, R.-S. Ieee 802.11ax: Next generation wireless local area networks. In: *10th International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. [s.n.], 2014. p. 77–82. Disponível em: <<https://doi.org/10.1109/QSHINE.2014.6928663>>. Citado na página 16.

DUGAEV, D.; SIEMENS, E. A wireless mesh network ns-3 simulation model: implementation and performance comparison with a real test-bed. In: BIBLIOTHEK, HOCHSCHULE ANHALT. *Proceedings of the 2nd International Conference on Applied Innovations in IT, Volume Nr. 2*. Anhalt University of Applied Sciences, 2014. p. 1–5. Disponível em: <https://icaiit.org/issue.php?paper=2nd_ICAIIT/>. Citado na página 30.

DUTTA, I. K.; GHOSH, B.; BAYOUMI, M. Lightweight cryptography for internet of insecure things: A survey. In: *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*. [S.l.: s.n.], 2019. p. 0475–0481. Citado na página 20.

FELDMANN, A. et al. A year in lockdown: How the waves of covid-19 impact internet traffic. *Commun. ACM*, Association for Computing Machinery, New York, NY, USA, v. 64, n. 7, p. 101–108, jun 2021. Disponível em: <<https://doi.org/10.1145/3465212>>. Citado na página 16.

GAUTAM, A.; MAHAJAN, R.; ZAFAR, S. Performance analysis of hybrid protocol under distributed denial of service attack in remote healthcare systems. *Recent Advances in Computer Science and Communications*, v. 15, n. 3, 2020. Citado na página 38.

HAKIRI, A.; GOKHALE, A.; BERTHOU, P. Software-defined wireless mesh networking for reliable and real-time smart city cyber physical applications. In: *Proceedings of the 27th International Conference on Real-Time Networks and Systems*. New York, NY, USA: Association for Computing Machinery, 2019. (RTNS '19), p. 165–175. Disponível em: <<https://doi.org/10.1145/3356401.3356406>>. Citado 2 vezes nas páginas 16 e 17.

HARRIS, C. R. et al. Array programming with NumPy. *Nature*, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, set. 2020. Disponível em: <<https://doi.org/10.1038/s41586-020-2649-2>>. Citado na página 45.

HUNTER, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007. Citado na página 45.

IBRAHIM, I. et al. Performance comparison of aodv and hwmp routing protocols in wireless mesh networks. In: *2013 IEEE International RF and Microwave Conference (RFM)*. [S.l.: s.n.], 2013. p. 116–120. Citado 2 vezes nas páginas 36 e 46.

IEEE Standard for Information Technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 10: Mesh Networking. *IEEE Std 802.11s-2011 (Amendment to IEEE Std 802.11-2007 as amended by IEEE 802.11k-2008, IEEE 802.11r-2008, IEEE 802.11y-2008, IEEE 802.11w-2009, IEEE 802.11n-2009, IEEE 802.11p-2010, IEEE 802.11z-2010, IEEE 802.11v-2011, and IEEE 802.11u-2011)*, p. 1–372, 2011. Citado 3 vezes nas páginas 16, 28 e 29.

JAVED, A. R. et al. Future smart cities: requirements, emerging technologies, applications, challenges, and future aspects. *Cities*, v. 129, p. 103794, 2022. ISSN 0264-2751. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0264275122002335>>. Citado na página 26.

- KASHEF, M.; VISVIZI, A.; TROISI, O. Smart city as a smart service system: Human-computer interaction and smart city surveillance systems. *Computers in Human Behavior*, v. 124, p. 106923, 2021. ISSN 0747-5632. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0747563221002466>>. Citado na página 17.
- KENAZA, R. et al. Internet of things (iot): Architecture, applications, and security challenges. In: *2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*. [S.l.: s.n.], 2022. p. 1–5. Citado na página 21.
- MALANDRA, F. et al. A case study for m2m traffic characterization in a smart city environment. In: *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*. New York, NY, USA: Association for Computing Machinery, 2017. (IML '17). ISBN 9781450352437. Disponível em: <<https://doi.org/10.1145/3109761.3109809>>. Citado na página 27.
- MORENO, R.; OLIVEIRA, C. Avaliação de desempenho de redes em malha sem fio em cenários de cidades inteligentes. In: *Anais da X Escola Regional de Computação do Ceará, Maranhão e Piauí*. Porto Alegre, RS, Brasil: SBC, 2022. p. 109–118. ISSN 0000-0000. Disponível em: <<https://sol.sbc.org.br/index.php/ercemapi/article/view/21965>>. Citado na página 19.
- MORENO, R.; OLIVEIRA, C. de. Um estudo de desempenho do hwmp em ambientes de cidades inteligentes. In: *Anais Estendidos do XII Simpósio Brasileiro de Engenharia de Sistemas Computacionais*. Porto Alegre, RS, Brasil: SBC, 2022. p. 80–85. ISSN 2763-9002. Disponível em: <https://sol.sbc.org.br/index.php/sbesc_estendido/article/view/22846>. Citado na página 19.
- MOULEMA, P. et al. On effectiveness of mesh-based protocols for smart grid communication networks. *SIGAPP Appl. Comput. Rev.*, Association for Computing Machinery, New York, NY, USA, v. 14, n. 2, p. 59–70, jun 2014. Disponível em: <<https://doi.org/10.1145/2656864.2656869>>. Citado na página 45.
- MOZUMDER, A. H.; ACHARJEE, T.; ROY, S. Scalability performance analysis of batman and hwmp protocols in wireless mesh networks using ns-3. In: *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*. [S.l.: s.n.], 2014. p. 1–5. Citado 2 vezes nas páginas 36 e 37.
- NATARAJU, A.; MAHESHAPPA, H. D.; DEVKATTE, A. Performance analysis of hwmp protocol for wireless mesh networks using ns3. In: *2016 IEEE Region 10 Conference (TENCON)*. [S.l.: s.n.], 2016. p. 1593–1598. ISSN 2159-3450. Citado na página 37.
- NAYYAR, A. Flying adhoc network (fanets): Simulation based performance comparison of routing protocols: Aodv, dsdv, dsr, olsr, aomdv and hwmp. In: *2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD)*. [S.l.: s.n.], 2018. p. 1–9. Citado na página 37.
- OBERASCHER, M.; RAUCH, W.; SITZENFREI, R. Towards a smart water city: A comprehensive review of applications, data requirements, and communication technologies for integrated management. *Sustainable Cities and Society*, v. 76, p. 103442, 2022. ISSN 2210-6707. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2210670721007150>>. Citado na página 27.

ODA, T. et al. A genetic algorithm-based system for wireless mesh networks: analysis of system data considering different routing protocols and architectures. *Soft Computing*, v. 20, n. 7, p. 2627–2640, Jul 2016. Disponível em: <<https://doi.org/10.1007/s00500-015-1663-z>>. Citado na página 34.

OTHMAN, M. M.; EL-MOUSA, A. Internet of things & cloud computing internet of things as a service approach. In: *2020 11th International Conference on Information and Communication Systems (ICICS)*. [S.l.: s.n.], 2020. p. 318–323. Citado 3 vezes nas páginas 20, 22 e 23.

PARVIN, J. R. An overview of wireless mesh networks. In: KHATIB, M.; ALSADI, S. (Ed.). *Wireless Mesh Networks*. Rijeka: IntechOpen, 2019. cap. 1. Disponível em: <<https://doi.org/10.5772/intechopen.83414>>. Citado 2 vezes nas páginas 17 e 28.

Paszkievicz, Andrzej; Zapala, Przemyslaw. The modified metric for self-organization wireless mesh networks. *ITM Web Conf.*, v. 21, p. 00010, 2018. Disponível em: <<https://doi.org/10.1051/itmconf/20182100010>>. Citado na página 29.

RETHFELDT, M. et al. Adaptive multi-channel clustering in iee 802.11s wireless mesh networks. *Sensors*, v. 21, n. 21, 2021. Disponível em: <<https://www.mdpi.com/1424-8220/21/21/7215>>. Citado na página 17.

ROSE, K.; ELDRIDGE, S.; CHAPIN, L. The internet of things: An overview. *The internet society (ISOC)*, Reston, VA, v. 80, p. 1–50, 2015. Citado na página 20.

SAXENA, D.; RAYCHOUDHURY, V. Design and verification of an ndn-based safety-critical application: A case study with smart healthcare. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, v. 49, n. 5, p. 991–1005, 2019. Citado na página 27.

SILVA, C. et al. Performance evaluation of wireless mesh networks in smart cities scenarios. In: *Proceedings of the Euro American Conference on Telematics and Information Systems*. New York, NY, USA: Association for Computing Machinery, 2018. (EATIS '18). Disponível em: <<https://doi.org/10.1145/3293614.3293615>>. Citado 2 vezes nas páginas 27 e 37.

SIVANATHAN, A. et al. Characterizing and classifying iot traffic in smart cities and campuses. In: *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. [S.l.: s.n.], 2017. p. 559–564. Citado na página 27.

TONG, Z. et al. A survey on algorithms for intelligent computing and smart city applications. *Big Data Mining and Analytics*, v. 4, n. 3, p. 155–172, 2021. Citado 2 vezes nas páginas 17 e 25.

Transforma Insights. *Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030*. 2020. Acesso em: 21 fev. 2022. Disponível em: <<https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>>. Citado na página 17.

WANG, W. et al. Coexistent routing and flooding using wifi packets in heterogeneous iot network. *IEEE/ACM Transactions on Networking*, v. 29, n. 6, p. 2807–2819, 2021. Citado na página 17.

WANG, X.; LIM, A. O. Ieee 802.11s wireless mesh networks: Framework and challenges. *Ad Hoc Networks*, v. 6, n. 6, p. 970–984, 2008. ISSN 1570-8705. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1570870507001370>>. Citado 2 vezes nas páginas 30 e 32.

WASHBURN, D. et al. Helping cities understand “smart city” initiatives. *Growth*, v. 17, n. 2, p. 1–17, 2009. Citado na página 17.

WU, M. et al. Research on the architecture of internet of things. In: *2010 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE)*. [S.l.: s.n.], 2010. v. 5, p. V5–484–V5–487. Citado na página 21.

YANG, K.; MA, J.-f.; MIAO, Z.-h. Hybrid routing protocol for wireless mesh network. In: *2009 International Conference on Computational Intelligence and Security*. [S.l.: s.n.], 2009. v. 1, p. 547–551. Citado na página 32.