



Pós-Graduação em Ciência da Computação

Alexsanderson Vieira Santos

Um *framework* SDN para distribuição de *streaming* de vídeo para veículos conectados



Universidade Federal de Pernambuco
posgraduacao@cin.ufpe.br
<http://cin.ufpe.br/~posgraduacao>

Recife
2019

Alexsanderson Vieira Santos

Um *framework* SDN para distribuição de *streaming* de vídeo para veículos conectados

Este trabalho foi apresentado à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Área de Concentração: Redes de Computadores

Orientador: Kelvin Lopes Dias

Recife
2019

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

S237f Santos, Alexsanderson Vieira
Um *framework* SDN para distribuição de *streaming* de vídeo para veículos conectados / Alexsanderson Vieira Santos. – 2019.
81 f.: il., fig., tab.

Orientador: Kelvin Lopes Dias.
Dissertação (Mestrado) – Universidade Federal de Pernambuco. CIn, Ciência da Computação, Recife, 2019.
Inclui referências e apêndice.

1. Redes de computadores. 2. Priorização de dados. 3. SDN. I. Dias, Kelvin Lopes (orientador). II. Título.

004.6 CDD (23. ed.) UFPE- MEI 2019-105

Alexsanderson Vieira Santos

**“Um Framework SDN para Distribuição de Streaming de Vídeo
para Veículos Conectados”**

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

Aprovado em: 14/05/2019.

BANCA EXAMINADORA

Prof. Dr. José Augusto Suruagy Monteiro
Centro de Informática/UFPE

Profª. Dra. Atslands Rego da Rocha
Departamento de Engenharia de Teleinformática / UFC

Prof. Dr. Kelvin Lopes Dias
Centro de Informática / UFPE
(Orientador)

Decido este trabalho a minha família e minha namorada que foram porto seguro perante as dificuldades durante este percurso.

AGRADECIMENTOS

Agradeço primeiramente a Deus, pois sem ele nada disso seria possível, pois sua ajuda foi essencial para que esse caminho tivesse significado.

Ao meu orientador, professor Kelvin Lopes Dias pela oportunidade, ajuda e apoio que possibilitou a realização de meu sonho de concluir esse mestrado.

Agradeço especialmente aos meus amigos de laboratório, que por muitas vezes debatemos sobre idéias relacionadas a essa dissertação e a temas diversos, levando sempre àquele lugar alegria e diversão.

Sou grato por minha família, minha mãe Francisca Santos, minha irmã Alexsandra Santos, ao meu pai José Maria dos Santos, e ao meu irmão, Alexsandro Santos, que me apoiaram desde o início para seguir caminho da educação.

Um obrigado especial à minha namorada, Hayaly Soares, que me apoiou nesse caminho, e sempre me alegrou quando eu estava triste e me ajudou quando eu precisei.

Muito obrigado pela ajuda de todos, pois sem esse apoio não seria capaz de chegar onde cheguei.

“O único limite às nossas realizações de amanhã serão as nossas dúvidas hoje”.
(MAXWELL, 2012, p. 222)

RESUMO

Estima-se que até 2022 exista uma grande rede de veículos interconectados, com mais de 300 milhões de carros conectados, gerando mais de 400 PetaBytes de dados. Estes carros formam as chamadas redes *Vehicular Ad-hoc Networks* (VANETs) que são redes de veículos conectados trafegando dados. Esses dados podem ser informações de segurança, mensagens de controle, aplicações, etc. No cenário de redes VANETs temos redes *Internet of Things* (IoT) e *Cloud Computing* que fornecem suporte a diversas aplicações e serviços para os carros. Algumas dessas aplicações para VANETs são as de conforto e informação e entretenimento que visam fornecer aos viajantes o suporte de informações e entretenimento necessários para tornar a viagem mais agradável. Por exemplo, as aplicações baseadas em *streaming* de dados, que requerem recursos de Qualidade de Serviço - *Quality of Service* (QoS) específicos para uma maior Qualidade de Experiência - *Quality of Experience* (QoE). Espera-se que essas aplicações criem novas oportunidades comerciais, aumentando o mercado da tecnologia e tornando-a mais rentável. O desafio de prover QoS é ainda maior em VANETs devido à mobilidade, constantes desconexões, topologias instáveis, etc. Neste sentido, o paradigma *Software Defined Networking* (SDN) tem sido empregado para permitir melhor gerenciamento, flexibilidade e programabilidade às VANETs por meio das chamadas *Software Defined Vehicular Networking* (SDVN). Contudo, as arquiteturas SDVN, em geral, têm controladores centralizados e distantes do plano de dados, o que pode gerar problemas de atrasos na transmissão de aplicações de *streaming* de vídeo. Com base nessas premissas, propomos uma nova arquitetura baseada no agrupamento adaptativo e hierárquico dos veículos da rede VANETs. Utilizando controladores locais nas *Road Side Units* (RSUs) e mensagens de controle *OpenFlow* (OF), regras de fluxo são utilizadas para auxiliar na alocação de parâmetros de QoS nos veículos para priorizar tráfego de dados na rede. Tais regras de fluxo são calculadas com fundamento na agregação hierárquica nos controladores locais e sua configuração adaptativa baseado na visão da rede. As avaliações realizadas foram feitas a partir de um simulador desenvolvido nesta dissertação denominado *Software Defined Vehicular Network Simulator* (SDVN-Sim). Para tanto, utilizamos algumas métricas para avaliar a solução apresentada, elas são o atraso de transmissão, para avaliar a comunicação dos carros com os controladores mais próximos, o *overhead* de sinalização, para avaliar a quantidade de mensagens de controle OF em diferentes localizações de controladores e qualidade de experiência QoE, utilizando de informações de *Peak Signal to Noise Ratio* (PSNR) para verificar a qualidade dos dados de vídeo transmitidos. Como resultados obtidos temos ganhos médios acima de 20% na qualidade QoE do vídeo transmitido utilizando a solução proposta por esta dissertação.

Palavras-chaves: SDN. VANETs. SDVN. Agregação. Priorização de Dados. *OpenFlow*.

ABSTRACT

It is estimated that by 2022 there will be a large network of interconnected vehicles, with more than 300 million cars connected, generating more than 400 PetaBytes of data. These cars form the so-called Vehicular Ad-hoc Networks (VANETs) networks that are connected vehicle networks traveling data. This data can be security information, control messages, applications, etc. In the VANETs network scenario, we have Internet of Things (IoT) and Cloud Computing networks that provide support for various applications and services for cars. Some of these applications for VANETs are comfort and information and entertainment that aim to provide travelers with the information and entertainment they need to make the trip more enjoyable. For example, applications based on data streaming, which require specific Quality of Service (QoS) capabilities for a higher Quality of Experience (QoE). These applications are expected to create new business opportunities, increasing the technology market and making it more profitable. The challenge of providing QoS is even greater in VANETs due to mobility, constant disconnections, unstable topologies, etc. In this sense, the Software Defined Networking (SDN) paradigm has been employed to allow better management, flexibility and programmability for VANETs through the so-called Software Defined Vehicle Networking (SDVN). However, SDVN architectures typically have centralized and remote data plan controllers, which can lead to delays issues in video streaming applications. Based on these premises, we propose a new architecture based on the adaptive and hierarchical grouping of VANETs vehicles. Using local controllers in Road Side Units (RSUs) and OpenFlow Protocol (OpenFlow) control messages, flow rules are used to aid in the allocation of QoS parameters in vehicles to prioritize data traffic on the network. Such flow rules are calculated based on the hierarchical aggregation in the local controllers and their adaptive configuration based on the network view. The evaluations were made using a simulator developed in this dissertation called Software Defined Vehicular Network Simulator (SDVNSim). To do so, we used some metrics to evaluate the solution presented, they are the transmission delay, to evaluate the communication of the cars with the closest controllers, the signaling overhead, to evaluate the amount of OpenFlow control messages in different control locations and quality of QoE experience using Peak Signal to Noise Ratio (PSNR) information to verify the quality of the transmitted video data. As obtained results we have average gains above 20% in the QoE quality of the video transmitted using the solution proposed by this dissertation.

Keywords: SDN. VANETs. SDVN. Aggregation. Prioritization of Data. OpenFlow.

LISTA DE FIGURAS

Figura 1 – Representação de um cenário de rede VANET	20
Figura 2 – Aplicações de streaming de vídeo para passageiros em um veículo inteligente	22
Figura 3 – Mobilidade dos veículos entre as coberturas de sinal das RSUs	23
Figura 4 – Arquitetura SDN	25
Figura 5 – Arquitetura SDN em Camadas	26
Figura 6 – Quadro Ethernet 802.1p	28
Figura 7 – Localização do controle SDN em diferentes níveis	33
Figura 8 – Arquitetura de <i>Handover</i> em uma rede VANETs	35
Figura 9 – Arquitetura em camadas	44
Figura 10 – Diagrama de sequência de funcionamento do Handover	45
Figura 11 – Fluxograma de funcionamento algoritmo desenvolvido	46
Figura 12 – Esquema de <i>Handover</i> proposto nessa dissertação	47
Figura 13 – Representação do algoritmo de agregação e priorização	48
Figura 14 – Diagrama de Sequência Geral	50
Figura 15 – Módulo OFcar	51
Figura 16 – Módulo RSUC	52
Figura 17 – Módulo Servidor de Aplicações	52
Figura 18 – Módulo Servidor de Vídeo	53
Figura 19 – Representação das potências dos sinais utilizados	58
Figura 20 – Mapas representativos do bairro da Boa Vista na cidade de Recife-PE	59
Figura 21 – Representação gráfica da topologia utilizada nas simulações de Overhead	60
Figura 22 – Quantidade média de pacotes OpenFlow recebidos. (CL - Controlador Local, CN - Controlador na Nuvem)	60
Figura 23 – Representação gráfica da topologia utilizada nas simulações de QoS	63
Figura 24 – Comparação dos resultados de PSNR obtidos. (CA - Com Algoritmo de Prioridades, SA - Sem Algoritmo de Prioridades)	64
Figura 25 – Análise estatística para as avaliações de QoE. (CA - Com Algoritmo de Prioridades, SA - Sem Algoritmo de Prioridades)	65
Figura 26 – Comparação de uma cena do vídeo simulado.	66
Figura 27 – Representação gráfica da topologia utilizada nas simulações de Handover	66
Figura 28 – Tempo de reconexão médio observado. (LCsem - Controlador Local sem algoritmo, LCcom - Controlador Local com algoritmo)	67

LISTA DE TABELAS

Tabela 1 – Aplicações de redes VANETs e características, adaptado de (PAPADIMITRATOS et al., 2009) e (JABALLAH; CONTI; LAL, 2019)	21
Tabela 2 – Comparativo de simuladores de redes veiculares, adaptado de (MUSSA et al., 2015)	29
Tabela 3 – Interseções dos planos de localização do controle SDN	34
Tabela 4 – Comparação dos trabalhos relacionados	39
Tabela 5 – Qualidade estimada de vídeo a partir do PSNR	56
Tabela 6 – Fatores e Níveis especificados para as avaliações	58
Tabela 7 – Porcentagem de redução de mensagens de controle. Comparação Individual com cada CL em relação ao central.	61
Tabela 8 – Porcentagem de redução de mensagens de controle. Comparação agrupando os valores dos controladores locais.	62
Tabela 9 – Comparação individual, observando as velocidades máximas, dos ganhos de PSNR	65

LISTA DE ABREVIATURAS E SIGLAS

BGP	<i>Border Gateway Protocol</i>
DiffServ	<i>Differentiated Services</i>
DSRC	<i>Dedicated Short Range Communications</i>
FTP	<i>File Transfer Protocol</i>
GPRS	<i>General Packet Radio Services</i>
GPS	<i>Global Positioning System</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IoT	<i>Internet of Things</i>
IP	<i>Internet Protocol</i>
ITS	<i>Intelligent Transportation System</i>
IVC	<i>Inter-Vehicle Communications</i>
LCD	<i>Liquid-crystal display</i>
LDP	<i>Label Distribution Protocol</i>
LTE	<i>Long-Term Evolution</i>
MANET	<i>Mobile ad hoc network</i>
MPLS	<i>Multiprotocol Label Switching</i>
OBU	<i>On Board Unit</i>
OF	<i>OpenFlow</i>
ONF	<i>Open Networking Foundation</i>
OSPF	<i>Open Shortest Path First</i>
PPP	<i>Point-to-Point Protocol</i>
PSNR	<i>Peak Signal to Noise Ratio</i>
QoE	<i>Quality of Experience</i>
QoS	<i>Quality of Service</i>
RSUC	<i>Road Side Unit Controller</i>
RSUs	<i>Road Side Units</i>
RSVP-TE	<i>Resource Reservation Protocol - Traffic Engineering</i>
RTP	<i>Real-time Transport Protocol</i>

SDN	<i>Software Defined Networking</i>
SDVN	<i>Software Defined Vehicular Networking</i>
SDVNSim	<i>Software Defined Vehicular Network Simulator</i>
SUMO	<i>Simulation of Urban MObility</i>
TCP	<i>Transmission Control Protocol</i>
UDP	<i>User Datagram Protocol</i>
UMTS	<i>Universal Mobile Telecommunication System</i>
V2I	<i>Vehicle-to-Infrastructure Communications</i>
V2V	<i>Vehicle-to-Vehicle Communications</i>
V2X	<i>Vehicle-to-Everything</i>
VANETs	<i>Vehicular Ad-hoc Networks</i>
WEB	<i>World Wide Web</i>
Wi-Fi	<i>Wireless</i>
WiMax	<i>Worldwide Interoperability for Microwave Access</i>

SUMÁRIO

1	INTRODUÇÃO	15
1.1	MOTIVAÇÃO	15
1.2	OBJETIVOS	17
1.3	OBJETIVOS ESPECÍFICOS	17
1.4	ESTRUTURA DA DISSERTAÇÃO	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	VANETS	19
2.1.1	Fundamentos	19
2.1.2	Aplicações	21
2.1.3	Handover em redes VANETs	22
2.2	REDES DEFINIDAS POR <i>SOFTWARE</i>	24
2.2.1	Fundamentos	24
2.2.2	Protocolo <i>OpenFlow</i>	26
2.3	QUALIDADE DE SERVIÇO	27
2.3.1	Fundamentos	27
2.3.2	Padrão 802.1p	28
2.4	SIMULADORES DE REDES VEÍCULARES	29
2.5	CONSIDERAÇÕES FINAIS	31
3	TRABALHOS RELACIONADOS	32
3.1	ESQUEMAS DE LOCALIZAÇÃO DO CONTROLE SDN	32
3.2	<i>HANDOVER</i> EM REDES VANETS	34
3.3	CRITÉRIOS DE AGREGAÇÃO DE VEÍCULOS	36
3.4	ESTRATÉGIAS DE PRIORIZAÇÃO DE TRÁFEGO E QOS	37
3.5	CONSIDERAÇÕES FINAIS	39
4	ARQUITETURA PROPOSTA	41
4.1	CAMADAS ARQUITETURAIS	41
4.1.1	Plano de Dados	41
4.1.2	Plano de Controle Local	42
4.1.3	Plano de Controle Global	43
4.1.4	Camada de Aplicação e Internet	43
4.2	<i>HANDOVER</i> DE CONTROLADORES LOCAIS SDN	44
4.3	ALGORITMO DE PRIORIZAÇÃO	48

4.4	SDVNSIM: <i>FRAMEWORK</i> DE SIMULAÇÃO PARA REDES VEICULARES DEFINIDAS POR <i>SOFTWARE</i>	50
4.4.1	Componentes	50
4.5	CONSIDERAÇÕES FINAIS	53
5	AVALIAÇÃO EXPERIMENTAL	55
5.1	RECURSOS UTILIZADOS	55
5.2	MÉTRICAS	56
5.3	PARÂMETROS	57
5.3.1	Fatores e Níveis	58
5.4	CENÁRIO DE EXECUÇÃO DOS EXPERIMENTOS	59
5.5	EXPERIMENTOS	59
5.5.1	Experimento 1 - <i>Overhead</i> de sinalização em arquiteturas de loca- lização de controle SDN	59
5.5.2	Experimento 2 - QoE em aplicação de vídeo streaming	63
5.5.3	Experimento 3 - Atraso de reconexão Veículo-RSUC após <i>handover</i> geográfico, em ambiente de grande <i>overhead</i> de sinalização	66
5.6	CONSIDERAÇÕES FINAIS	68
6	CONCLUSÃO	69
6.1	CONCLUSÕES	69
6.2	CONTRIBUIÇÕES	70
6.3	LIMITAÇÕES E DIFICULDADES ENCONTRADAS	70
6.4	TRABALHOS FUTUROS	71
	REFERÊNCIAS	73
	APÊNDICE A – CÓDIGOS SHELL SCRIPT	79

1 INTRODUÇÃO

Este Capítulo descreve a motivação da dissertação, introduzindo conceitos na área de VANETs SDN, e conceitos sobre QoS. Em seguida, são apresentados o objetivo geral e os objetivos específicos, bem como a solução proposta e, por fim, a estrutura da pesquisa desenvolvida nesta dissertação.

1.1 MOTIVAÇÃO

Atualmente verifica-se um crescente número de dispositivos móveis conectados, o que está revolucionando muitos aspectos em nossas vidas. Projeções apontam que essa quantidade até 2021 será de 11,6 bilhões, superando a previsão da população mundial deste período (7,8 bilhões) (CISCO, 2016). Muitos desses dispositivos fazem parte da IoT, a qual considera que uma gama de objetos que estejam conectados à internet transfiram, processem, coletem dados, etc. Recentemente diversos veículos estão saindo de fábrica equipados com sensores, câmeras, radares, computadores de bordo. Deste modo, redes IoT estão sendo estendidas e, ao mesmo tempo, especializadas com a chamada redes VANETs.

As comunicações em redes VANETs estão recebendo uma grande atenção dos pesquisadores. Isto devido à demanda no uso de aplicações multimídia durante viagens. Um dos problemas que aparecem em redes VANETs é devido à grande mobilidade dos veículos, deixando a conectividade intermitente e deteriorando a qualidade no nível de aplicação. Um exemplo a ser citado, são serviços de multimídia que exigem largura de banda adequadas para usuários em redes veiculares, o que necessita de soluções de gerenciamento eficientes (AKYILDIZ et al., 2016). Além disso, como cada tipo de aplicação possui diferentes requisitos de QoS, estes cenários influenciam de forma mais ou menos intensa o seu funcionamento (REHMAN et al., 2017).

As redes VANETs foram criadas usando os princípios de redes móveis ad hoc, ou seja, é a criação espontânea de uma rede sem fio para troca de dados no cenário veicular. É o termo utilizado para identificar redes veiculares, que tem como principal objetivo possibilitar a comunicação de usuários móveis e oferecer as condições necessárias para que aplicações com diferentes requisitos sejam atendidas satisfatoriamente. Tais aplicações compõem um Sistema Inteligente de Transporte (*Intelligent Transportation System - ITS*). Exemplos dessas aplicações incluem a monitoração cooperativa do tráfego, a prevenção de colisões ou o auxílio a cruzamentos sem sinalização. Essas redes podem fazer o uso de algumas tecnologias sem fio como base. Por exemplo, é muito comum que um veículo possa ter várias interfaces de rede a bordo, essas interfaces podem ser *Wireless (Wi-Fi)*, *Dedicated Short Range Communications (DSRC)*, *Universal Mobile Telecommunication System (UMTS)*, *Worldwide Interoperability for Microwave Access (WiMax)*,

Bluetooth e etc (HE; ZHANG; LIANG, 2015). Por fim, ainda há desafios no desenvolvimento de aplicações VANETs, alguns como fluxo de tráfego entre topologias multi caminhos, e utilização ineficiente da rede. Assim sendo, arquiteturas de VANETs abertas e flexíveis são requisitos para melhorar o gerenciamento de recursos de rede, aplicações e usuários (KU et al., 2014).

O paradigma SDN é um novo conceito emergente em que modifica e provê soluções para as atuais infraestruturas de rede. Primeiramente, SDN separa os planos de controle e o encaminhamento de tráfego, plano de dados, dos dispositivos da rede. Segundo, com a separação entre os planos, esses *switches* tornam-se simples dispositivos de encaminhamento e a lógica de controle é implementada em um controlador centralizado, ou um sistema operacional de rede. Além disso, SDN é outro paradigma que apresenta estratégias para a melhoria das redes em geral. Ele foi originalmente definido para o mundo dos datacenters e redes cabeadas e torna a rede programável através da separação entre planos de dados e controle proporcionando um maior gerenciamento. A partir dessas características, SDN é capaz de prover flexibilidade, oferecendo para a rede um modelo mais simplificado de gerenciamento, possibilitando a criação de novas abstrações e facilitando a evolução e inovação da rede (KREUTZ et al., 2015).

Atualmente o *software* de gerenciamento dos dispositivos de rede como *switches* e roteadores são de código fechado. Essa característica ocasiona em pouca programabilidade e flexibilidade da rede. O paradigma SDN define a separação do *software* de controle do dispositivo, inserindo-o em um plano centralizado. O protocolo *OpenFlow* - OF foi proposto para padronizar a comunicação entre estes dispositivos de rede e plano de controle que irá gerenciá-los. Este gerenciamento é realizado através da implementação de tabelas de fluxo nos dispositivos da rede. Estas tabelas são usadas pelo controlador centralizado para inserção de regras de fluxo, enviadas através de mensagens do protocolo OF, para decidir como os dados trafegarão dentro da rede (LARA; KOLASANI; RAMAMURTHY, 2014).

Muitas aplicações executam em meio às redes veiculares, algumas delas são para segurança outras para entretenimento. Atualmente, com o crescimento da popularidade da Internet, muitas aplicações têm ganhado interesse, uma delas é o de *streaming* de vídeo. Propostas que utilizam esta aplicação buscam desenvolver soluções eficientes para a transferência de seus dados. Porém, uma das características de redes veiculares é a alta mobilidade dos carros e ambiente heterogêneo, que requer mecanismos de QoS eficazes para a execução dessas aplicações.

Em vista disso, e adaptando o modelo classes de serviços do protocolo 802.1p, a nossa proposta visa prover durante a mobilidade dos carros, o rápido acesso aos parâmetros de QoS que determinada aplicação necessite para transmitir seus frames na rede. Em nossa arquitetura, as RSUs possuem controladores SDN que, através de mensagens do OF, gerenciam os veículos daquela determinada área de atuação do sinal da RSUs. Nos controladores locais é implementado um algoritmo que processa requisições para a deter-

minação de agregação do veículo a um grupo virtual. Para isso parâmetros de QoS das aplicações são informados ao algoritmo, além do número de veículos que pertencerão ao determinado grupo virtual, no nosso caso estamos definindo maior prioridade para dados de *streaming* de vídeo. Um dos aspectos de nossa proposta é o da utilização do controlador mais próximo dos veículos, que proporciona menor tempo de comunicação das mensagens OF, bem como menor *overhead* de sinalização no controlador, já que estamos trabalhando com a visão da rede local. Resumindo, nosso trabalho visa reorganizar alocação dos valores de classes de serviços definidas pelo padrão 802.1p, através de uma solução de agregação virtual de dispositivos gerenciados por SDN, e assim priorizar aplicações de *streaming* de vídeo em redes veiculares.

1.2 OBJETIVOS

O principal objetivo deste trabalho é propor e desenvolver uma arquitetura SDN hierárquica para redes veiculares, com suporte à priorização de tráfego, QoS e *handover* no contexto de transmissões de *streaming* de vídeo fornecidas por provedores de conteúdo da Internet para usuários de veículos conectados.

1.3 OBJETIVOS ESPECÍFICOS

- Conceber e implementar uma arquitetura hierárquica de controladores SDN.
- Desenvolver um sistema adaptativo baseado em informações geográficas, priorização de tráfego e estratégia de agregação de veículos para o fornecimento de QoS.
- Propor e implementar proposta de sinalização para o *handover* entre RSUs, considerando a arquitetura hierárquica de controladores SDN e informações de localização dos veículos.
- Definir uma estratégia de conectividade baseada em mensagens OpenFlow.
- Propor e avaliar um modelo de gerenciamento mais próximo do plano de dados.

1.4 ESTRUTURA DA DISSERTAÇÃO

Esta dissertação contém seis Capítulos organizados da seguinte forma: O Capítulo 2 apresenta os conceitos relacionados às redes VANETs, os fundamentos, aplicações e acerca do *handover* nessas redes. Também são descritos conceitos sobre o paradigma de gerenciamento de redes SDN, e o protocolo OF. Elencamos também os conceitos relacionados do protocolo de QoS 802.1p. Por fim, falamos a respeito dos simuladores mais populares e utilizados em redes VANETs.

O Capítulo 3 apresenta e discute os trabalhos relacionados que inspiraram o tema proposto por esta dissertação. Para tanto, os trabalhos elencados foram divididos em subtemas, primeiro falamos das propostas relacionadas aos esquemas de localização de controle SDN, depois discutimos sobre os trabalhos relacionados de *handover* em redes VANETs, após discorreremos acerca dos trabalhos com relação a critérios de agregação de veículos, por fim descrevemos as propostas das estratégias de priorização de dados e QoS.

O Capítulo 4 descreve a arquitetura e a proposta desta dissertação. São discutidas características sobre o *framework* desenvolvido nesta dissertação denominado SDVNSim. Discorreremos acerca das informações do algoritmo de *handover* entre controladores locais da rede SDVN. Por fim, é feita a descrição das características e funcionalidades do algoritmo de priorização de dados desenvolvido.

O Capítulo 5 descreve as avaliações experimentais realizadas, especificando os recursos utilizados e as métricas e parâmetros escolhidos. Também detalhamos o ambiente de teste e cenários definidos em cada experimento e os resultados obtidos com a utilização dos sistemas propostos.

Por fim, o Capítulo 6 contém as conclusões finais desta dissertação, descrevendo as considerações finais, as contribuições que este trabalho está realizando, algumas limitações e dificuldades encontradas e possíveis propostas para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, são apresentadas as tecnologias relacionadas ao desenvolvimento da proposta desta dissertação. Está dividido em duas subseções. A Seção 2.1 aborda sobre as redes VANETs e suas características primordiais. A Seção 2.2 aborda o paradigma SDN que é utilizado neste trabalho no desenvolvimento de parte do sistema de controle da rede. A seção 2.3 explica os conceitos acerca de QoS e as principais características do protocolo 802.1p. A Seção 2.4 discorre sobre os simuladores de redes veiculares, porque são utilizados quais seus benefícios e alguns exemplos deles. Por fim, a Seção 2.5 elenca algumas considerações finais sobre os temas centrais descritos nesse capítulo.

2.1 VANETS

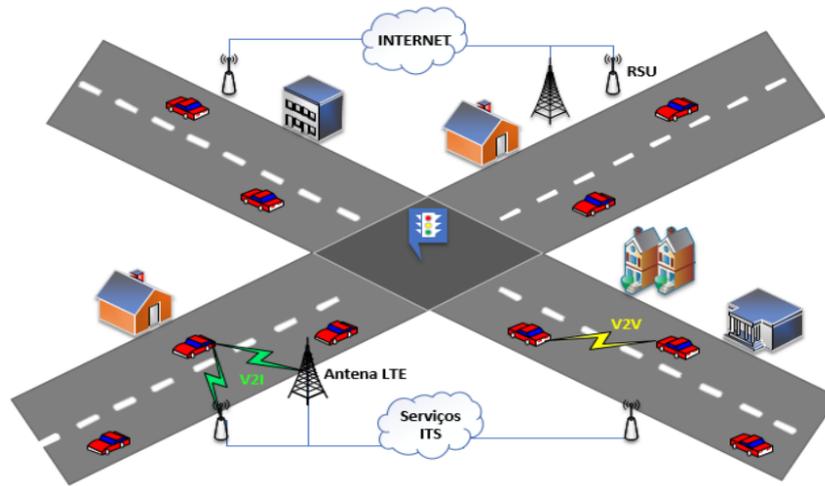
Estudos relacionados com VANETs têm atraído bastante atenção por sua grande quantidade de aplicações suportadas. Algumas destas aplicações são para áreas de segurança (*safety*) e emergência, como alerta de colisões, prevenção de acidentes, avisos de engarrafamentos, re-roteamento de veículos e etc. Além dessas, podemos ter aplicações de entretenimento utilizadas na Internet como jogos e *streaming* de vídeo para o conforto do passageiro. Com as VANETs, os veículos e infraestruturas ao redor trocam informações de uma forma coordenada. De uma maneira geral, as VANETs são componentes importantes na implantação de Sistemas Inteligentes de Transporte (*Intelligent Transportation System* (ITS)) e das futuras cidades inteligentes (*Smart Cities*).

2.1.1 Fundamentos

As redes veiculares VANETs têm se destacado como uma área de pesquisa promissora, bem como, considerada uma infraestrutura catalisadora de soluções para grandes desafios da mobilidade urbana nas Cidades Inteligentes (KRANENBURG, 2008) (XU; HE; LI, 2014). Podemos observar através da Figura 1 uma representação de redes veiculares.

O conceito de VANETs, vem sendo bastante usado pela academia para se referir ao modelo de comunicação definido pela ITS. Esse ramo de estudos é bastante mencionado em pesquisas relacionadas à Internet das Coisas - IoT (ALIYU et al., 2018) (TOMOVIC et al., 2017) (AL-TURJMAN, 2018). É prevista uma grande quantidade de dados que serão gerados e trafegados na rede, o que demandará soluções de gerência desses dados no tráfego para não degradar a rede. A comunicação em redes VANETs ocorre de duas formas, *Vehicle-to-Vehicle Communications* (V2V), *Vehicle-to-Infrastructure Communications* (V2I). A primeira é a comunicação inter-veicular e a segunda a comunicação dos veículos com a infraestrutura adjacente (KU et al., 2014). Abaixo na Figura 1 podemos observar um exemplo de rede VANETs e seus tipos de comunicações.

Figura 1 – Representação de um cenário de rede VANET



Fonte: Elaborada pelo Autor

Os carros de nova geração, que serão pertencentes às redes VANETs, estão se tornando elementos com alta capacidade de processamento e de armazenamento, equipados com câmeras, sensores, radares e dispositivos de posicionamento (KAIWARTYA et al., 2016) (SHEET et al., 2016). Será comum nestes veículos várias interfaces de rede. Estas interfaces poderão ser Wi-Fi, DSRC, UMTS, WiMax, *Bluetooth* e etc, podendo o carro vir a utilizar múltiplas interfaces de rede simultaneamente. Por exemplo, o veículo pode usar a interface Wi-Fi para transmitir e UMTS para receber os dados, o que pode ocasionar no aumento da largura de banda da rede (HE; ZHANG; LIANG, 2015) (FONTES et al., 2017).

A grande demanda de VANETs inclui todos os desafios que uma rede sem fio móvel deve superar, somando a isso características específicas da comunicação. Estas redes são altamente dinâmicas devido à alta mobilidade dos nós. Outra característica presente nestas redes é que mudanças constantes na topologia podem ocorrer, o que ocasionará em grandes taxas de perdas de pacotes.

A popularidade das redes veiculares na pesquisa científica em geral tem uma justificativa, o impacto das aplicações no cotidiano das pessoas, principalmente dos condutores de veículos e passageiros. Este impacto impõe diversas exigências sobre as tecnologias de redes veiculares, o que leva a uma série de desafios de pesquisa. Os principais desafios estão relacionados com temas sobre endereçamento de rede e endereçamento geográfico, gerenciamento e análise de risco, anonimato, privacidade, localização segura, algoritmos de encaminhamento, restrições de atraso, controle de congestionamento, controle da rede, encaminhamento de grande volume de dados, entre outros. Por isso, tanto a indústria quanto a academia estão extremamente ativos nessa área de pesquisa e se espera avanços significativos nos próximos anos.

Em redes VANETs existem três modelos de comunicação, estas descritas a seguir:

- **Vehicle-to-Vehicle (V2V)**: Nessa arquitetura os veículos se comunicam entre si sem a necessidade de nenhuma estrutura externa ou elemento centralizador. Nesse caso há uma comunicação inter-veicular, denominado *Inter-Vehicle Communications* (IVC) que envolve exclusivamente dois ou mais veículos. Para tanto, os veículos funcionam como roteadores e encaminham tráfego através de múltiplos saltos. Cada veículo é equipado com um *On Board Unit* (OBU), responsável por realizar esse tipo de comunicação.
- **Vehicle-to-Infrastructure (V2I)**: Nessa arquitetura existem RSUs distribuídos ao longo das rodovias. Esses nós estáticos funcionam como pontos de acesso da rede IEEE 802.11 também em modo infraestruturado. Eles centralizam todo o tráfego da rede, servindo como nós intermediários das comunicações. A grande vantagem desse modo é a possibilidade de comunicação com outras redes, a exemplo da Internet;
- **Vehicle-to-Everything (V2X)**: Essa arquitetura é uma solução intermediária entre comunicação V2V e V2I. Sendo assim há a possibilidade de utilizar uma infraestrutura estática para melhorar a conectividade da rede e prover serviços bem como uma comunicação entre veículos através de múltiplos saltos;

2.1.2 Aplicações

São três as principais categorias de aplicações em sistemas de comunicação veicular: segurança do transporte, eficiência do transporte e entretenimento. As duas primeiras são as principais motivações para o desenvolvimento dessas redes. A terceira categoria é mais recente e muito pesquisada atualmente. A Tabela 1 apresenta vários tipos de aplicações e suas características principais.

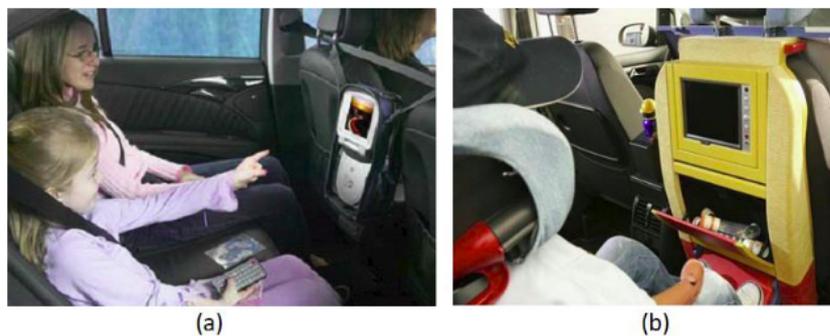
Tabela 1 – Aplicações de redes VANETs e características, adaptado de (PAPADIMITRATOS et al., 2009) e (JABALLAH; CONTI; LAL, 2019)

Aplicação	Mensagem	Comunicação	Requisitos
Aviso de colisão	periódica, permanente	V2V, V2I	localização precisa, alta prioridade
Aviso de pista perigosa	eventual, limitada por um período de tempo	V2V, V2I	alta prioridade
Aviso de veículo lento	periódica, permanente	V2V	alta prioridade
Detecção de potencial batida	periódica	V2V	alcance: 50 ms, média/alta prioridade
Download de Mídias	sob demanda	VI2	acesso à internet
Download/Atualização de mapas	sob demanda	V2V, V2I	acesso à internet, disponibilidade de serviço
Assistência de condução ecológica	sob demanda	V2V, V2I	acesso à internet, disponibilidade de serviço

Aplicações para VANETs são muitas das vezes orientadas a questões de segurança, por exemplo, serviços de tráfego e mensagens de avisos, com o objetivo de melhorar a qualidade do transporte através de aplicações críticas de segurança e gerenciamento de tráfego. Porém, ao mesmo tempo, aplicações de entretenimento também estão aumentando, por exemplo, *streaming* de vídeo e jogos *online*, navegação na web, etc.

Espera-se que essas aplicações de entretenimento criem novas oportunidades comerciais, aumentando o mercado da tecnologia e tornando-a mais rentável. Além disso, as aplicações de conforto e informação e entretenimento visam fornecer aos viajantes o suporte de informações e entretenimento necessários para tornar a viagem mais agradável. Por exemplo, *streaming* de mídia, voz sobre *Internet Protocol* (IP), navegação na Web, etc. Até aplicativos exclusivos do ambiente veicular, por exemplo, anúncios de ponto de interesse, *download* de mapas, pagamentos de estacionamento, serviços de pedágio, etc (VEGNI; BIAGI; CUSANI, 2013). A Figura 2 (a) e (b) mostra o uso de dispositivos de *Liquid-crystal display* (LCD) para aplicativos de entretenimento.

Figura 2 – Aplicações de streaming de vídeo para passageiros em um veículo inteligente



Fonte: SLIPPERYBRICK (2019)

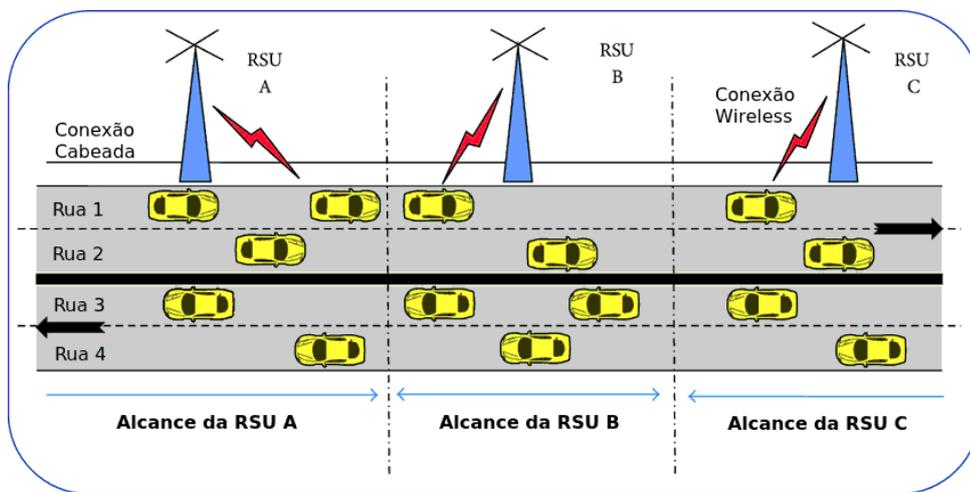
Além dessas aplicações, imagina-se que futuramente os veículos terão acesso à internet em qualquer lugar e a qualquer instante. Muitos pesquisadores acreditam que os benefícios oferecidos pelas redes veiculares para a segurança no trânsito e para as aplicações comerciais justificam o custo da rede, pois o valor da montagem de uma rede veicular ainda é elevado.

2.1.3 Handover em redes VANETs

As redes veiculares são compostas por veículos que estão equipados com sensores e uma OBU que serve para comunicação entre os próprios veículos e a infraestrutura adjacente. Essa infraestrutura é composta por dispositivos denominados RSUs, que podem ser baseados em tecnologias como DSRC 802.11p, *General Packet Radio Services* (GPRS) e Wi-Fi (ARANITI et al., 2013).

A crescente demanda de conectividade aumenta a preocupação com a QoS e a confiabilidade da entrega. Muitos pesquisadores estão tentando desenvolver novos modelos de *handover* para redes VANETs, porém a alta mobilidade dos carros pode representar um desafio. Podemos observar na Figura 3 uma representação da mobilidade dos carros entre diferentes alcances de antenas da rede veicular. A alta mobilidade irá criar frequentes transferências entre antenas adjacentes, que causa atrasos e perdas de pacotes, e o ambiente denso dos veículos, bem como a rede veicular heterogênea, criam novos desafios. Os veículos entram e saem das áreas de cobertura das redes, acarretando frequentes mudanças de topologia (SALIHIN et al., 2018).

Figura 3 – Mobilidade dos veículos entre as coberturas de sinal das RSUs



Fonte: Adaptada de (GANESHKUMAR; GOKULAKRISHNAN, 2015)

O termo *handover* refere-se ao processo de transferência de uma sessão de dados em andamento de um dispositivo de uma estação base para outra. Originalmente a transferência era apenas para mudar uma chamada telefônica em curso entre duas estações de telefone celular, no entanto a definição tornou-se mais ampla hoje em dia, surgindo variações de subtipos de *handover*.

Atualmente, diversos trabalhos e pesquisas vêm sendo realizados pela indústria e academia e com isso algumas características são adotadas como os tipos mais comuns de *handover*. Abaixo discorremos sobre os tipos de *handover*.

- **Handover Horizontal:** é caracterizado como uma transferência quando há uma mudança intra-domínio. Exemplificando, é quando o equipamento de usuário conectado muda entre duas células que usam a mesma tecnologia.
- **Handover Vertical:** é caracterizado por um processo envolvendo a transferência quando há uma mudança inter-célula, entre domínios. Exemplificando, o equipamento de usuário conectado muda seu ponto de acesso para outro que usa uma

tecnologia diferente, por exemplo: *handover* de uma estação base *Long-Term Evolution* (LTE) para Wi-Fi.

- **Handover Diagonal:** É a combinação dos dois tipos explicados anteriormente, *handover* horizontal e vertical. O equipamento do usuário altera sua rota e a tecnologia subjacente conectada simultaneamente.

2.2 REDES DEFINIDAS POR SOFTWARE

Podemos observar com o passar do tempo e com a evolução das redes de computadores, as fabricantes de dispositivos de rede tornaram o *software* destes totalmente privado. A partir disso, o paradigma SDN tem o objetivo principal de prover programabilidade a esses dispositivos de rede, separando o *software* de controle do *hardware* para encaminhamento de pacotes.

Este paradigma SDN propõe superar certas limitações existentes nas redes tradicionais, pelo fato dos administradores estarem restritos a funcionalidades já definidas dos equipamentos de infraestrutura de rede.

2.2.1 Fundamentos

A arquitetura distribuída foi essencial para o crescimento do protocolo IP. No entanto, com o passar do tempo, novas soluções como aplicativos e serviços provocaram mudanças da arquitetura predominantemente cliente-servidor. Por exemplo, soluções como, aplicativos para celulares, virtualização e provedores de nuvem, mudaram o padrão de tráfego ocasionando necessidade da rede ser mais dinâmica (FOUNDATION, 2012). Os obstáculos para atender as novas exigências fizeram com que as redes tradicionais oferecessem resistência à inovação.

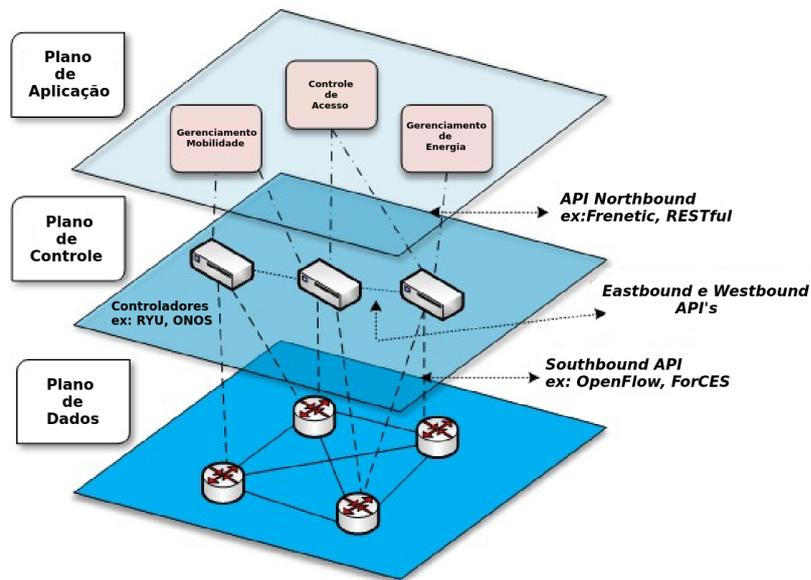
As redes SDN constituem um novo paradigma para o desenvolvimento de pesquisas em redes de computadores que vem ganhando a atenção de grande parte da comunidade acadêmica e da indústria na área (GUEDES et al., 2012). O princípio por trás das redes SDN é possuir a capacidade de controlar o plano de dados através de uma interface bem definida. O plano de controle tem o objetivo de realizar tomadas de decisões que resultam na confecção das tabelas de roteamento, e o plano de dados cuida da comutação e repasse dos pacotes na rede (COSTA, 2013).

A arquitetura atual de roteadores de rede é formada basicamente por duas camadas distintas, que é o *software* de controle e o *hardware* dedicado ao encaminhamento de pacotes. Em contrapartida a arquitetura das redes SDN subdivide essas camadas de forma que seja possível programar essa rede, permitindo que o plano de controle possa ser movido para um servidor dedicado (COSTA, 2013).

O paradigma SDN tem como objetivo favorecer uma maior flexibilidade e programabilidade em redes. Diferente do modelo atualmente utilizado para dispositivos de enca-

minhamento de dados em que o *software* de controle é fechado e definido para o próprio *hardware* do dispositivo, SDN define uma separação dos planos de dados e controle, esse último pode ser centralizado em um dispositivo denominado controlador de rede, que tem a função de ser toda a inteligência da rede, tornando o roteador apenas um dispositivo de encaminhamento de acordo com as tabelas de decisão atualizadas pelo controlador (KREUTZ et al., 2015). Podemos observar na Figura 4 uma representação das camadas do paradigma SDN e alguns componentes.

Figura 4 – Arquitetura SDN



Fonte: Elaborada pelo Autor

Ou seja, o paradigma SDN é caracterizado pela existência de um sistema de controle, ou seja *software*, que pode controlar o mecanismo de encaminhamento dos elementos de comutação da rede por uma interface de programação bem definida (GUEDES et al., 2012).

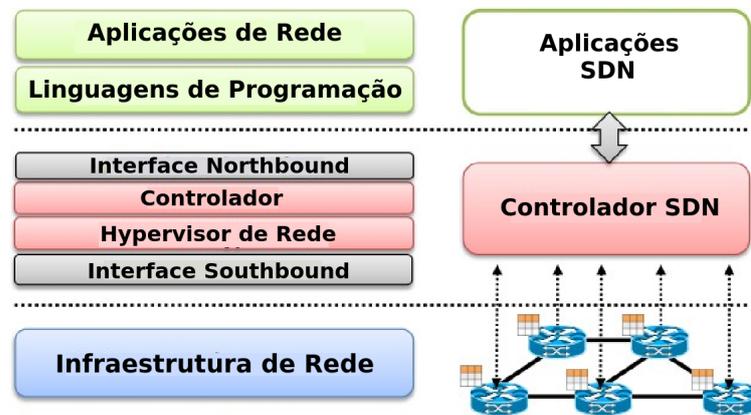
De acordo com (RAMOS; KREUTZ; VERISSIMO, 2015) as vantagens da utilização de SDN são:

- Torna-se mais fácil a programação de aplicações para essas redes pois uma vez que as abstrações providas pela plataforma de controle e as linguagens de programação de rede podem ser compartilhadas.
- Todas as aplicações podem tirar proveito das mesmas informações de rede (no ponto de vista de rede global), levando a mais decisões políticas coerentes e eficazes.
- As aplicações podem tomar medidas a partir de qualquer parte da rede, assim não há a necessidade de elaborar uma estratégia precisa sobre a localização da nova funcionalidade.

- A integração de diferentes aplicações torna-se mais simples. Por exemplo, o balanceamento de carga e aplicações de roteamento podem ser combinados sequencialmente, tendo precedência sobre políticas de roteamento.

Uma arquitetura SDN pode ser descrita como uma composição de diferentes camadas, cada camada tem suas próprias funções específicas. Algumas interfaces fazem o intermédio de comunicação entre estas camadas, por exemplo a interface *Southbound* que faz a comunicação do plano de controle com o de dados e a interface *Northbound* que intercambia a comunicação entre as aplicações de rede e o controlador (RAMOS; KREUTZ; VERISSIMO, 2015). Na Figura 5 podemos visualizar os pontos principais da arquitetura SDN.

Figura 5 – Arquitetura SDN em Camadas



Fonte: Adaptada de (RAMOS; KREUTZ; VERISSIMO, 2015)

2.2.2 Protocolo *OpenFlow*

O OpenFlow (OF) foi o primeiro protocolo aberto e popular de comunicação entre os planos de controle e de dados da rede SDN. Os seus preceitos ilustram a praticidade da implementação do paradigma SDN. Diante da inovação que esse protocolo poderia ocasionar, um grupo de empresas organizaram-se em um consórcio em prol do OF e fundaram a *Open Networking Foundation* (ONF).

O paradigma de SDN e o protocolo OF foram desenvolvidos com o propósito de viabilizar o desenvolvimento e testes de novas propostas de redes, sem afetar a rede real ou rede de produção (GUEDES et al., 2012). Tal protocolo é um dos padrões, senão o mais popular, utilizado na comunicação de dispositivos da camada de dados com o plano de gerenciamento. O OF foi desenvolvido por pesquisadores da universidade de *Stanford* (MCKEOWN et al., 2008) e após isso foi adotado e padronizado pela ONF¹.

O funcionamento básico de uma rede OF consiste em que todos os switches-roteadores estão, a princípio, com as tabelas sem nenhum fluxo instalados. Desse modo, quando um

¹ ONF: <<https://www.opennetworking.org/>>

pacote chega ao *switch* OF, ele envia uma mensagem ao controlador para este decidir o que fazer. O controlador devolve ao *switch* uma mensagem contendo uma instrução que pode consistir de ações variadas, tais como encaminhar o pacote para todas as portas, encaminhar para a próxima tabela do pipeline, criar um fluxo ou, ainda, descartar o pacote.

A importância do entendimento das propriedades do protocolo OF, bem como as características de cada versão, é dividido em dois aspectos: (1) cada controlador SDN possui linguagem de programação e versão do OF específicas; (2) a checagem da versão escolhida que deve ser compatível com os equipamentos existentes na infraestrutura de rede.

2.3 QUALIDADE DE SERVIÇO

O termo QoS refere-se à capacidade de uma rede para proporcionar o melhor serviço de tráfego de rede, e para isso é preciso satisfazer diferentes requisitos. Deste modo, uma rede que implementa QoS deve ser capaz de atender a diferentes requisitos a partir da demanda explícita das aplicações.

2.3.1 Fundamentos

Uma rede com suporte ao QoS é uma condição ideal para sistemas multimídias. Os benefícios seriam vantajosos, como, por exemplo, a priorização do vídeo sensível ao atraso em relação aos outros tipos de dados ou outras aplicações de vídeo (APOSTOLOPOULOS; TAN; WEE, 2002).

Os principais sistemas de *streaming* de dados utilizam a Internet ou a rede tradicional IP como rede de entrega de conteúdo. Esses sistemas sofrem com a variação da largura de banda disponível, o que provoca perdas de pacotes e atrasos. Isso implica diretamente na baixa QoE dos usuários. As redes tradicionais IP, de maneira geral, não garante QoS a nenhuma aplicação.

As aplicações que fornecem serviços de *streaming* de dados devem ter perdas de pacotes e atraso toleráveis e não somente largura de banda disponível. Perdas de pacotes elevadas, inviabilizam a execução dessas aplicações. Como a convergência é uma realidade, praticamente tudo roda sobre o protocolo IP, a diminuição e o controle do atraso, da variação do atraso e da perda de pacotes será primordial (APOSTOLOPOULOS; TAN; WEE, 2002).

Um dos mecanismos de QoS utilizados é o de controle de tráfego por agregação. O tráfego é agrupado em um conjunto de dados de múltiplas aplicações e classificado para o mesmo fluxo e controlado de forma agregada. Um bom exemplo de controle de tráfego por agregação é o modelo 802.1p e serviços diferenciados.

2.3.2 Padrão 802.1p

A popularidade e aumento do uso das aplicações de multimídias, imagens, vídeos e voz necessita a elevação da capacidade de processamento e armazenamento dos dispositivos pessoais, bem como conexões de redes mais rápidas e estáveis. O tráfego gerado por essas aplicações multimídia possuem requisitos específicos de QoS (TANENBAUM; WETHERALL, 1996) (UFRN-GTA, 2002).

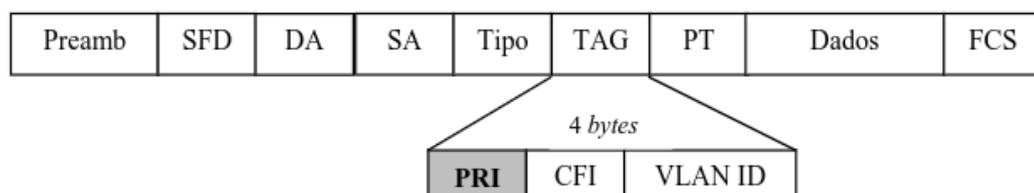
Atualmente a arquitetura de rede Internet é projetada para o envio de informações no melhor esforço, ou seja, sem garantias de QoS e observando os recursos disponíveis no momento. Dessa forma, para o suporte de aplicações multimídia em *streaming* em tempo real na Internet são necessários mecanismos para que as aplicações indiquem seus requisitos de QoS à rede, e mecanismos de gerenciamento de tráfego (TANENBAUM; WETHERALL, 1996) (UFRN-GTA, 2002).

O padrão 802.1p é um padrão definido pelo *Institute of Electrical and Electronics Engineers* (IEEE) para priorização de fluxos com tempo crítico, priorizando esses dados na camada 2 de redes. O cabeçalho do protocolo inclui três bits para priorização permitindo a configuração de oito níveis de prioridade (EK, 1999). O 802.1p é um padrão que está bem consolidado em redes de computadores, definindo a prioridade em redes Ethernet.

Para se fazer uso dessa prioridade do 802.1p, é necessário que o *switch* em questão tenha algum mecanismo para controlar a QoS bem como algoritmos de filtragem em caso de congestionamento. Ou seja, tem que ser implementado com filas separadas, com políticas de encaminhamento específicas para quadros com prioridades diferentes, e consequentemente com necessidades de QoS também diferentes (EK, 1999) (BANAVALIKAR, 2015).

Quando uma aplicação requer uma determinada priorização dentro da rede local, podemos exemplificar o caso de telefones IP, o telefone faz a marcação da precedência e o *frame* Ethernet segue para o *switch* de acesso que naturalmente precisa suportar o protocolo 802.1p. A priorização é explicitamente definida na marcação do campo, esta priorização não deriva do endereço MAC de origem nem de destino. Os quadros Ethernet podem ser marcados de acordo com a sua importância através dos bits de prioridade 802.1p assim como na Figura 6.

Figura 6 – Quadro Ethernet 802.1p



Fonte: Elaborada pelo Autor

Uma classe com uma prioridade alta terá precedência na estação local e possivelmente em outras pertencentes na mesma rede.

2.4 SIMULADORES DE REDES VEÍCULARES

O número de veículos trafegando nas estradas principalmente nas grandes cidades metropolitanas aumentou consideravelmente nos últimos anos. Devido a esse aumento tem ocorrido a necessidade de melhoramento em relação à segurança do tráfego, bem como a comunicação e eficiência das aplicações nas redes implantadas. Para isso, o ITS tem sido empregado para fornecer suporte a várias aplicações de segurança e até de eficiência de tráfego (MUSSA et al., 2015).

Cabe destacar também, a quantidade de aplicações está aumentando rapidamente, e com isso a crescente necessidade de se criar uma plataforma padronizada para redes VANETs (MUSSA et al., 2015). Diante da necessidade de avaliar alternativas de projetos para redes veiculares e das dificuldades envolvidas na configuração de ambientes experimentais reais, a utilização de simuladores de redes e suas extensões têm sido cada vez mais empregada no avanço das pesquisas em VANETs. A Tabela 2 apresenta vários tipos de simuladores utilizados para redes VANETs caracterizados em critérios específicos.

Tabela 2 – Comparativo de simuladores de redes veiculares, adaptado de (MUSSA et al., 2015)

Critério/ Simulador	GrooveNet	TraNS	NCTUns	VEINS
Portabilidade de Software	Não	Não	Não	Sim
Freeware	Sim	Sim	Não	Sim
OpenSource	Sim	Sim	Não	Sim
Linguagem de Programação	C++	Java, C++	C++	C++
Documentação	Não	Sim	Sim	Sim
Exemplos Disponíveis	Não	Sim	Não	Sim
Desenvolvimento Continuado	Não	Não	Sim	Sim
Facilidade de Uso	Moderado	Moderado	Moderado	Fácil
Escalabilidade	Sim	Sim	Sim	Sim

As simulações de redes VANETs são aplicadas em vários cenários de grande escala considerando características especiais do ambiente veicular. Então, testar e implementar redes veiculares na realidade requer alta quantidade de dinheiro e trabalho intenso. Além

disso, a implementação real de redes VANETs com ambientes complexos pode demandar de grandes recursos financeiros. Alternativamente, as ferramentas de simulações podem ser usadas para construir cenários complexos com resultados precisos e equivalentes em comparação com a plataforma real. Com isso, verifica-se que os modelos de mobilidade podem afetar a precisão dos resultados da simulação e vale ressaltar que, o modelo simulado deve estar o mais próximo possível do real (CAVIN; SASSON; SCHIPER, 2002) (MUSSA et al., 2015).

Os principais simuladores utilizados em redes VANETs elencados por (MUSSA et al., 2015), e relacionados na Tabela 2 são descritos a seguir:

- **GrooveNet:** A universidade *Carnegie Mellon* desenvolveu o GrooveNet² este é um simulador híbrido que permite a comunicação entre os veículos da simulação. Ele tem a capacidade de incorporar entidades relacionadas ao tráfego do mundo real nos estudos da simulação. Esse simulador fornece uma interface gráfica agradável e mapas reais extraídos do banco de dados denominado TIGER. A arquitetura em módulos desse simulador consolida modelos de transmissão, mobilidade e transmissão de mensagens em diferentes tipos de comunicação física e de camada de enlace.
- **TraNS:** este é um simulador de código aberto que oferece duas operações de uso, centrado na rede e na aplicação. O primeiro modo é usado para aplicações que não afetam, em tempo real, a mobilidade dos nós, como troca de dados e *download* de músicas, etc. O segundo é dedicado a aplicativos que influenciam o comportamento dos veículos durante o tempo da simulação, como alertas de emergência e aplicativos de prevenção de colisão. A implementação atual do TraNS³ é uma combinação do simulador de rede ns-2 e do simulador de tráfego SUMO, vinculado como uma ferramenta exclusiva para criar uma simulação realística.
- **NCTUns:** é um simulador e emulador híbrido que permite a simulação integrada de tráfego da rede. O NCTUns⁴ suporta processadores multicore e programação paralela, bem como pode simular vários protocolos IP com e sem fio, tais como Ethernet, LANs sem fio IEEE 802.11b/e, WIMAX 802.16d/e, redes de satélites DVBRCS, redes móveis multi-interface e redes veiculares sem fio 802.11p/1609WAVE.
- **VEINS:** é um simulador bastante popular entre os pesquisadores na área de redes veiculares, os autores deste simulador o desenvolveram de forma robusta e altamente escalável, além de ser código aberto. O simulador VEINS⁵ é composto por mais dois outros simuladores, o OMNeT++, que é um simulador de rede, e o SUMO que define os modelos de tráfego rodoviário.

² GrooveNet: <<https://github.com/mlab-upenn/GrooveNet>>

³ TraNS: <<http://lca.epfl.ch/projects/trans/>>

⁴ NCTUns: <<http://nsl.cs.nctu.edu.tw/NSL/nctuns.html>>

⁵ VEINS: <<https://veins.car2x.org/>>

O *framework* OMNeT++⁶ é uma biblioteca e estrutura de simulação C++ extensível, modular e baseada em componentes, principalmente para construir simuladores de rede. A partir deste, foi possível a implementação de diversas soluções para redes veiculares.

Outro *framework* que cabe ser mencionado é o INET⁷ que é uma biblioteca de modelo de código aberto para o ambiente de simulação do OMNeT++. Fornece protocolos, agentes e outros modelos para pesquisadores e estudantes que trabalham com redes de comunicação. Ele contém modelos para a pilha de protocolos da Internet *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP), IPv4, IPv6, *Open Shortest Path First* (OSPF), *Border Gateway Protocol* (BGP) e diversos modelos de aplicativos.

Por fim, o *Simulation of Urban MObility* (SUMO)⁸ é um pacote de simulação de tráfego rodoviário projetado para lidar com grandes redes rodoviárias. Ele permite simular como uma determinada demanda de tráfego equivalente em veículos individuais que se movem através de uma determinada rede rodoviária. A simulação permite abordar um grande conjunto de tópicos de gerenciamento de tráfego. Cada veículo é modelado explicitamente, tem uma rota própria e se move individualmente pela rede.

2.5 CONSIDERAÇÕES FINAIS

A pesquisa em redes VANETs é bem ampla e a cada dia ganha mais força no meio acadêmico. Isso é devido à popularidade dos carros no dia a dia das pessoas. Essas acabam passando horas dentro de um veículo, seja no traslado para o trabalho ou parados em um congestionamento. Visto isso, a união das soluções veiculares e a popularidade das aplicações de entretenimento como vídeo *streaming* e jogos, novas soluções viáveis são necessárias para trazer essas aplicações para o mundo de redes veiculares.

Para tanto, algumas tecnologias podem ser utilizadas para prover maior programabilidade e gerenciabilidade para a rede onde esteja sendo executada alguma aplicação de entretenimento. Uma dessas tecnologias é SDN, este paradigma de rede tem o objetivo de separar o plano de controle dos dispositivos da rede promovendo maior programabilidade. Esse paradigma aplicado no gerenciamento dos requisitos específicos de QoS das aplicações de *streaming* na rede VANETs pode proporcionar grandes benefícios.

⁶ OMNeT++: <https://omnetpp.org/>

⁷ INET: <https://inet.omnetpp.org/>

⁸ SUMO: <https://sumo.dlr.de/userdoc/>

3 TRABALHOS RELACIONADOS

Este capítulo descreve trabalhos relacionados encontrados na literatura durante a pesquisa desta dissertação. Os trabalhos foram divididos em três categorias que são apresentadas em seções distintas deste capítulo. Na Seção 3.1 são discutidas as propostas que tratam de esquemas que posicionam o controle SDN em camadas distintas do modelo arquitetural para redes veiculares. A Seção 3.2 apresenta trabalhos com estratégias de QoS e priorização de dados utilizadas em aplicações de redes VANETs. A Seção 3.3 aborda artigos sobre os critérios de agregação que definem métricas para organização dos veículos para tomada de decisões. Já a Seção 3.4 elenca e discute trabalhos que utilizam propostas de priorização de dados para prover melhor QoS às aplicações em redes VANETs. Por fim, a Seção 3.5 destaca as considerações finais sobre o estado da arte dos temas e trabalhos discutidos.

3.1 ESQUEMAS DE LOCALIZAÇÃO DO CONTROLE SDN

A proximidade do controlador SDN ao plano de dados pode influenciar o tempo de resposta das decisões, uma vez que as mensagens terão um caminho menor para ir entre o plano de dados e o plano de controle.

Um ponto importante a se considerar em redes VANETs é a alta mobilidade dos veículos, o que ocasiona em topologias muito voláteis no modo V2V e desconexões frequentes do veículo com a infraestrutura, outros problemas são o uso ineficiente de recursos, enlaces com tráfego desbalanceado e os atrasos na comunicação. Para mitigar esses problemas na rede, algumas abordagens utilizam tecnologias específicas como SDN para proporcionar maior controle durante a operação nas VANETs.

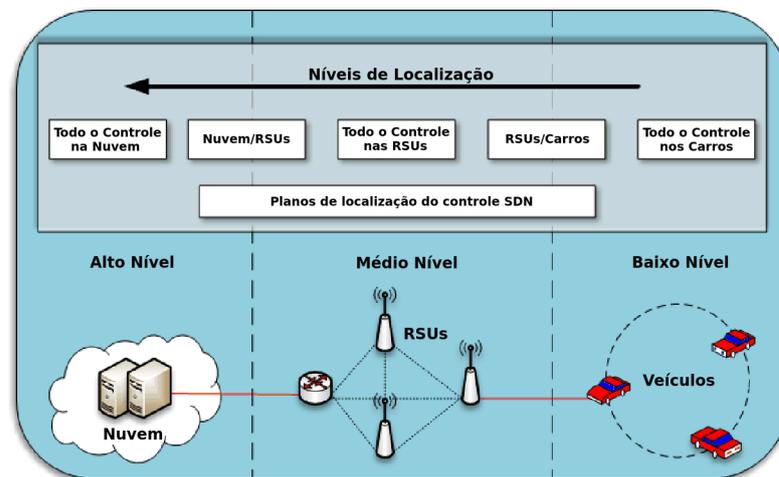
São três as hierarquias principais de controle SDN em redes veiculares observadas, onde em cada uma delas um conjunto de características e objetivos são descritos. Na sequência, segue uma descrição das camadas, bem como a Figura 7 ilustrando os seguintes níveis:

- **Nível Superior:** significa o mais alto nível observado que uma arquitetura pode possuir na solução de redes SDVN, Nuvem pública (Provedor na Internet) ou privada (operadora), Torre Celular, e outros. O objetivo principal da parte da alta localização é a implementação do controle SDN totalmente na Nuvem ou parte dele. Através de nossos estudos observamos que a grande maioria das soluções propostas para o controle de redes SDVN é implementado nesse nível de localização.
- **Nível Intermediário:** soluções que implementam propostas do controle SDN na localização das RSUs em sua totalidade. O objetivo principal observado nesta categoria é a configuração do controle SDN mais próxima do plano de dados da rede veicular, em RSUs ou torres celulares, por exemplo. Muitos dos trabalhos observa-

dos na literatura estão iniciando a implementar o controle SDN nessa localização, com o objetivo de obter melhores tempos de atraso de comunicação de mensagens de controle e menor sobrecarga de sinalização nos controladores.

- **Nível Inferior:** caracteriza todo o controle SDN alocado no mesmo nível do plano de dados, ou seja, nos próprios veículos. O objetivo principal desse nível é observado em cenários onde não há a comunicação V2I, ou seja, a comunicação com a infraestrutura adjacente composta por torres celulares e RSUs. Os trabalhos elencados nesse nível explicam que muitas das vezes o veículo pode não estar conectado à infraestrutura, visto isso, o que requer soluções nas quais os próprios veículos assumam a responsabilidade pelas tomadas de decisões.

Figura 7 – Localização do controle SDN em diferentes níveis



Fonte: Elaborada pelo Autor

Além dos três principais níveis de localização foram observados outros dois níveis, porém esses são níveis interligados ou derivados dos principais descritos anteriormente. Na Tabela 3 são descritos com detalhes os demais níveis interligados.

Observando alguns trabalhos da literatura como (TRUONG; LEE; GHAMRI-DOUDANE, 2015), (SUDHEERA et al., 2016), (HUANG et al., 2017), (VENKATRAMANA; SRIKANTAI AH; MOODABIDRI, 2017), (WANG et al., 2017), (CHEN et al., 2017), (SECINTI et al., 2017) e (ZHU et al., 2018) esses implementam o plano de controle inteiramente na nuvem. Esse tipo de abordagem distancia o controle SDN do plano de dados, ou seja, os veículos, essa distância pode proporcionar um maior atraso na comunicação dos dispositivos com o plano de controle, ocasionando decisões de rede tardias.

A configuração do controle inteiramente na nuvem é vantajosa porque permite a visão global da rede VANETs. Por outro lado, haverá maior atraso na comunicação e enorme quantidade de mensagens de controle que terão que ser processadas neste único ponto centralizado.

Tabela 3 – Interseções dos planos de localização do controle SDN

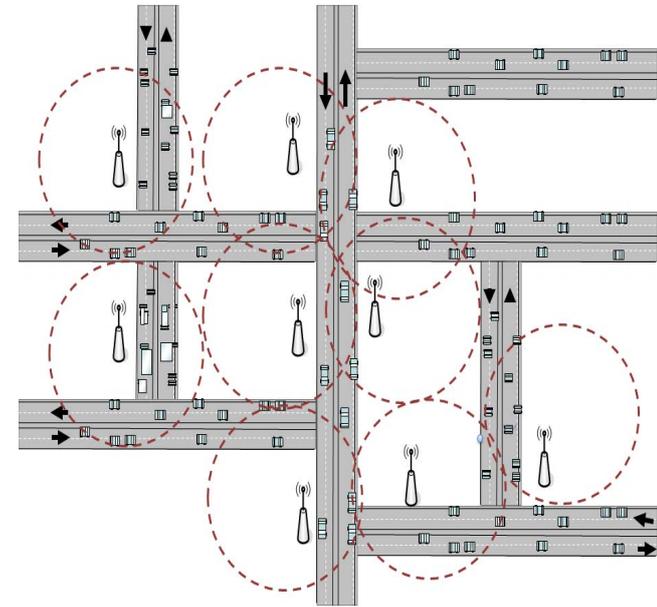
Interseção dos Planos Superior e Médio	São propostas que visam aproximar o gerenciamento de controle SDN dos veículos.	(TRUONG; LEE; GHAMRI-DOUDANE, 2015) (2015), (SUDHEERA et al., 2016), (SALAHUDDIN et al., 2014), (SALAHUDDIN; AL-FUQAHA; GUIZANI, 2015), (DONG et al., 2016), (KU et al., 2014), (HUSSEIN et al., 2017)
Interseção dos Planos Médio e Baixo	São propostas que visam a alocação do gerenciamento de controle SDN tanto nas antenas RSUs quanto nos veículos da rede.	(ALOUACHE et al., 2018) (2018), (ALIOUA; SENOUCI; MOUSSAOUI, 2017) (2017)

Por fim, alguns trabalhos como de (RADHAKRISHNAN et al., 2017) e (SOUA et al., 2017), são propostas que utilizam arquiteturas de localização do controle SDN mais próxima dos veículos, porém não há um esquema de reconfiguração de comunicação entre os controladores locais da rede para uma maior qualidade de transmissão das mensagens de controle.

3.2 HANDOVER EM REDES VANETS

Visto isso, o trabalho de (GHOSH et al., 2015) propõe um mecanismo de *handover* baseado em uma abordagem de cobertura probabilística e não fixa. Neste trabalho é discutido que muitas das políticas tradicionais de *handover* estão baseadas na abordagem reativa, ou seja, na qual o carro reage à sinalização indicando mudanças na conectividade de rede à medida que o veículo se movimenta. Para tanto, a decisão proativa probabilística de *handover* em que o veículo decide quando e onde entregar pode ajudar a desenvolver um mecanismo eficiente que seja possível minimizar a perda de pacotes e a interrupção dos serviços. Essa abordagem probabilística é realizada baseando-se no cálculo do tempo em que um determinado veículo tem para fazer o *handover*, isso sem perder a conexão. O objetivo é fazer o melhor gerenciamento da alocação e desalocação de canal em função do *handover* realizado nos veículos da rede. O algoritmo proativo desenvolvido utiliza dois parâmetros: tempo antes da transferência vertical (TBVH), que é o tempo no qual a transferência deve ocorrer, e tempo de permanência da rede (NDT), que é o tempo que o veículo gasta na cobertura da nova rede. Contudo, a proposta proativa não considera o QoS e QoE das aplicações em execução, isto é um ponto importante para verificar se a solução proposta impacta na qualidade das aplicações. Além do mais, nessa proposta, não

Figura 8 – Arquitetura de *Handover* em uma rede VANETs



Fonte: (SMIDA; FANTAR; YOUSSEF, 2017)

se considera o gerenciamento de *handover* por SDN, o que limita a programabilidade da solução em um cenário maior de rede VANETs. Por exemplo, a utilização do gerenciamento SDN em um nível intermediário, ou seja os controladores nas RSUs, mitigaria os problemas elencados anteriormente.

A proposta de (KAUL et al., 2018) permite que um controlador automaticamente delegue controle SDN de um veículo para outro controlador à medida que o veículo se move na rede VANETs. Nessa proposta, é considerado o algoritmo de *handover* apenas nos controladores locais. O objetivo principal desse trabalho é fazer com que a delegação de controle também possa ser usada para equilibrar a carga entre controladores e garantir a qualidade de aplicação necessária. Porém, essa abordagem do controlador apenas no nível intermediário, o que restringe a visão global da rede necessitando assim de que mais mensagens transitem na rede para intercâmbio de decisões de *handover* entre os controladores, podendo gerar *overhead* de sinalização. Outra característica que pode ser observada nessa proposta é que ela não avalia um cenário realista de rede VANETs, utilizando nas avaliações apenas três veículos. A proposta avalia o impacto da qualidade de serviço observada, através do balanceamento de carga entre controladores, porém não considera critérios chave de *handover* como latência do aplicativo e requisitos de largura de banda.

No trabalho de (SKONDRAS et al., 2017) foi desenvolvido um algoritmo de *handover* vertical para sistemas de nuvem em ambiente VANETs baseado na velocidade do veículo e no tipo de conexão atual. Neste trabalho o veículo é configurado para sempre ter a melhor conexão com a melhor tecnologia levando em consideração a aplicação que está sendo executada na nuvem. Se um veículo estiver em alta velocidade, o processo de *handover*

pode ser desconsiderado. Os objetivos são de reduzir custos das operações, melhorar o gerenciamento da mobilidade do usuário e garantir sempre a melhor conexão ao veículo. Porém, o trabalho não apresenta uma avaliação de desempenho dos requisitos de QoS e QoE das aplicações utilizadas, bem como não considera o *multi-homing*.

Por fim, o trabalho de (SMIDA; FANTAR; YOUSSEF, 2017) apresenta um mecanismo de *handover* com base na estimativa de qualidade de *link* das RSUs vizinhas na rede veicular. O objetivo central da proposta é minimizar o tempo de *handover* minimizando tempo de procura da RSUs mais próxima. Entretanto, a proposta não considera um ambiente heterogêneo de conexões, por exemplo, LTE e 802.11p. Além do mais, o trabalho não considera o fato de o veículo poder ter duas interfaces e assim receber o *streaming* de forma *multi-homing*.

3.3 CRITÉRIOS DE AGREGAÇÃO DE VEÍCULOS

Abordagens que utilizam os preceitos de agregação de informações buscam assim oferecer um serviço único para todos, que sem a agregação seria tratado individualmente. Algumas propostas utilizam agregação para definir as áreas de atuação de um determinado controlador local. Aliado a essa característica com a mobilidade dos veículos entre essas agregações virtuais, a reconfiguração de regras nos dispositivos será muito grande, aumentando o número de mensagens de controle.

O trabalho de (DUAN; LIU; WANG, 2017) propõe um modelo SDN onde os nós veículos são agregados de acordo com dados de mobilidade dos veículos e usando informações SDN globais. Dentro de cada grupo de veículos, um líder do grupo (CH)¹ é selecionado para agregar tráfego de outros veículos e se comunicar com a torre celular, a fim de reduzir a sobrecarga de sinalização. Porém nesse trabalho não foi considerada a mobilidade dos veículos, bem como a quantidade de veículos simulados foi pequena. Isso é um problema, pois em ambientes com alta mobilidade a escolha do veículo CH é dinâmica e o algoritmo desenvolvido pode não funcionar, bem como a quantidade de veículos pode ser grande e ocasionar ocorrência de grande quantidades de dados no *cluster head*.

O trabalho de (LIU; LI, 2017) agrega os pacotes de entrada da camada superior em três filas de prioridade de acordo com o atraso máximo de fluxo tolerável. Esses fluxos de tráfego que possuem a menor taxa de atraso necessária para serem transmitidos na rede recebem a prioridade de fila mais alta para garantir a eficácia dos dados em tempo real. No entanto, esse trabalho não usa o controle SDN para gerenciamento de rede, reduzindo a capacidade de programação. Além disso, a simulação realizada não avalia um cenário realista de VANETs ao não considerar a mobilidade de carros e um número reduzido de veículos.

¹ *Cluster Head*

3.4 ESTRATÉGIAS DE PRIORIZAÇÃO DE TRÁFEGO E QOS

A popularidade das plataformas de vídeo está muito elevada e muitas pesquisas estão sendo realizadas para que essas plataformas sejam implementadas eficientemente em redes VANETs.

Muitas pesquisas vêm sendo desenvolvidas, até mesmo na área de SDVN, para desenvolver soluções viáveis no gerenciamento do tráfego desses dados de vídeo. No trabalho de (ZHANG et al., 2018) é proposto um mecanismo de concorrência com múltiplas prioridades para implementar a diferenciação de serviços e a alocação de recursos. Nesse projeto o tráfego dos veículos é priorizado com base na combinação de contexto e tipos de dados, obtém-se assim uma prioridade mais adequada para alcançar a diferenciação de serviços. Por mais que seja uma solução parecida com a dessa dissertação, no trabalho não tem um mecanismo de gerenciamento programável, por exemplo SDN, além de não considerar a mobilidade dos veículos nos cenários avaliados.

Outro exemplo é o trabalho proposto por (ATWAL; GULERIA; BASSIOUNI, 2018), nele um controlador global mantém um repositório de banco de dados centralizado para uma visualização consistente da topologia e de outras atividades de rede utilizadas pelo planejador para implementar filas relacionadas a QoS e aplicar as políticas específicas de prioridade. Nesse caso a política de priorização de tráfego utilizada é a IEEE 802.11e. Diferente dessa dissertação, o trabalho não considera um esquema de agregação ou *clustering* na solução, aumentando a quantidade de dados e informações que serão processadas pelo algoritmo desenvolvido e assim aumentando a carga de trabalho.

A alta mobilidade dos veículos na rede cria problemas de instabilidade nas topologias formadas. Além disso o grande número de interfaces de comunicação acoplados nas RSUs e os veículos trazem problemas aos tradicionais protocolos de roteamento utilizados nas redes. A partir dessa visualização, (VENKATRAMANA; SRIKANTIAH; MOODABIDRI, 2017) propõe um novo protocolo de roteamento para redes VANETs com gerenciamento de conexão SDN a partir de decisões geograficamente feitas. Um protocolo específico visa aumentar o desempenho de conexões e otimizar a transmissão de pacotes de dados. Para o funcionamento do protocolo desenvolvido, foi desenvolvido um esquema de priorização com base na diferença de velocidade e origem, para conseguir um roteamento de dados mais estável. Porém, esse esquema proposto não utiliza abordagens de agregação dos dados dos veículos para o gerenciamento mais satisfatório do protocolo de roteamento. Outro problema encontrado nessa proposta é que foi mostrado que com a grande quantidade de interfaces de comunicação serão necessários meios de gerenciamento eficazes, porém esta proposta não aborda o handover de comunicação entre as interfaces heterogêneas.

O trabalho de (SMIDA; FANTAR; YOUSSEF, 2017) apresenta um mecanismo de handover e é descrito na seção 3.2 dos trabalhos relacionados ao handover. Além do mecanismo de handover proposto um algoritmo de prioridade foi desenvolvido com listas baseadas na qualidade dos links das RSUs. No entanto, a proposta não considera um ambiente

de conexão heterogêneo, por exemplo, LTE e 802.11p, reduzindo o escopo de ação da proposta. Além disso, o trabalho não reconhece o fato de que o veículo pode ter duas interfaces e, assim, receber o fluxo de dados de forma *multi-homing*, o que poderia ser aplicado como uma função no algoritmo da lista de prioridades.

Devido a um grande número de serviços que podem ser executados em redes VANET, a característica de os dispositivos possuírem grande número de interfaces de comunicação pode se tornar um problema. Isso pode exigir soluções com conexões flexíveis e eficientes entre os dispositivos de rede VANET. A partir disso, (CHEN et al., 2017) propõe um modelo de gerenciamento centralizado usando o paradigma SDN para controlar a alocação de recursos de comunicação dedicados em veículos para satisfazer cada serviço. Esta solução foi introduzida devido à grande variedade de serviços ITS. Esses serviços têm requisitos e custos de qualidade de QoS diferentes a serem atendidos para não diminuir a qualidade fornecida ao usuário final. Para tanto, foi proposto um mecanismo de priorização baseado no tamanho dos dados e no tempo limite de transmissão. O problema é que a solução apresentada não foi avaliada em um cenário realista de redes VANET sendo avaliado apenas na modelagem matemática, além de distanciar o controle SDN do sistema na implementação da nuvem.

Nas redes VANET, embora o equipamento de transmissão de sinal esteja cada vez mais eficiente, ainda podemos encontrar situações em áreas urbanas específicas que não são cobertas. Para isso, (ZHU et al., 2018) propõe um novo modelo que visa melhorar a taxa de cobertura e o desempenho na transmissão de mensagens de emergência na rede VANET. Para isso, o autor do trabalho desenvolveu um esquema de priorização de mensagens de emergência baseado na comunicação das condições de destino, e emergência do tráfego na rede VANET. Um problema observado neste trabalho é a distância do controle SDN do plano de dados da rede VANET, aumentando o tempo para a entrega de mensagens de controle. Outro problema é que esta proposta não usa um modelo de agregação de informações para agilizar no mecanismo de decisão no controlador SDN.

Por fim, podemos observar na Tabela 4 uma comparação dos trabalhos relacionados que foram utilizados nesta dissertação.

Tabela 4 – Comparação dos trabalhos relacionados

Proposta	Objetivo Principal	Critérios para Seleção				Ambiente de Avaliação
		Localização SDN	Handover	Agregação	Priorização	
RADHAKRISHNAN et al. (2017)	Propor um mecanismo para balanceamento de carga de dados nos canais de comunicação da rede VANET	RSUCs	-	Cluster de veículos virtuais gerenciados na RSUC	-	Simulação
HUANG et al. (2017)	Propõe um esquema de gerenciamento de conexões 802.11p e celular em redes VANETs.	Nuvem	Uma proposta de previsão SDN de offloading e handover entre as interfaces de conexão dos veículos	-	-	Simulação
VENKATRAMANA; SRIKANTAIHAH; MOODABIDRI (2017)	Uma proposta de protocolo de roteamento geográfico para melhor desempenho de transmissão de dados.	Nuvem	-	-	Um esquema de priorização com base na diferença de velocidade e sua origem para roteamento de dados mais estável	Emulação
SMIDA; FANTAR; YOUSSEF (2017)	Proposta para minimizar o tempo de procura da RSU mais próxima para realização do handover	-	Mecanismo de handover com base na estimativa de qualidade de link das RSUs	-	Lista de prioridades baseadas na qualidade dos links das RSUs	Simulação
DUAN; LIU; WANG (2017)	Melhorar a sobrecarga de sinalização agregando o tráfego de dados	Nuvem	Handover de dados agregados entre diferentes canais	Proposta de agregação de veículos	-	Simulação
LIU; LI (2017)	Propor um esquema de agregação de prioridades para tráfego de dados em redes VANETs	-	-	Algoritmo de agregação de prioridades	Lista de prioridades com base no atraso tolerável para transmissão de um dado na rede VANET	Simulação
ZHANG et al. (2018)	Proposta de diferenciação de serviços segundo múltiplas prioridades	-	-	Agregação de veículos virtuais	Lista de prioridades para priorização e alocação de recursos de rede	Simulação
ATWAL; GULERIA; BASSIOUNI (2018)	Proposta de priorização de dados com o gerenciamento SDN em redes VANETs	Nuvem e RSUCs	Gestão de mobilidade distribuída entre os controladores locais para o handover	-	Filas de prioridades específicas com base no padrão 802.11e	Emulação
CHEN et al. (2017)	Uma abordagem centralizada de conexões para prover recursos de comunicação dedicada.	Nuvem	-	Um esquema de agregação dos estados da rede no controlador central	Um mecanismo de priorização com base no tamanho dos dados e o tempo limite de transmissão.	Modelagem Matemática
ZHU et al. (2018)	Proposta de mecanismo de gerenciamento SDN para mensagens de emergência trafegados em redes VANETs	Nuvem	-	-	Esquema de priorização de mensagens de emergência com base na transmissão de destino, emergência e condições.	Simulação
GHOSH et al. (2015)	Cálculo probabilístico baseado no tempo de conexão para realização de handover	-	Cálculo probabilístico com base no tempo de conexões	-	-	Testbed
KAUL et al. (2018)	Delegação automática de controles SDN da rede entre antenas RSUCs	RSUCs	Handover de informações com base na potência do sinal das antenas RSUCs	-	-	Emulação
SKONDRAS et al. (2017)	Solução de gerenciamento de handover vertical para redes VANETs	Nuvem	Algoritmo de handover com base na velocidade e tipo de conexão dos veículos	-	Conexões com tipo de serviço específicos (SLAs)	Modelagem Matemática
Esta Dissertação	Arquitetura de agregação hierárquica para priorização de dados, handover geográfico entre controladores locais e esquema de controle SDN próximo ao plano de dados	Nuvem e RSUCs	Handover de comunicação OpenFlow entre controladores locais	Agregação virtual de veículos da rede VANET	Priorização em camadas baseado no tipo de agregação virtual e aplicação em execução	Simulação

3.5 CONSIDERAÇÕES FINAIS

Uma das características mais importantes em redes VANETs é a alta mobilidade dos veículos, o que ocasiona em topologias muito dinâmicas, outros problemas são o uso ineficiente de recursos, o tráfego desequilibrado e os atrasos na comunicação. Para mitigar

esses problemas algumas propostas utilizam a proximidade do controlador SDN ao plano de dados para mais agilidade no gerenciamento da rede, uma vez que devido grande mobilidade dos carros necessita de mecanismos eficientes e ágeis de gerenciamento da rede.

Entretanto, em redes VANETs o gerenciamento de mobilidade e *handover* ainda são alguns dos pontos mais pesquisados e mais desafiadores, visto que essas redes suportam uma grande variedade de aplicações, mecanismos eficazes de gerenciamento de mobilidade se fazem necessários para melhor qualidade das aplicações em execução em redes VANETs. Outro ponto a se considerar é que os esquemas tradicionais de gerenciamento de mobilidade e *handover* muitas das vezes não atendem aos requisitos de redes VANETs, causando a degradação da qualidade das aplicações em execução nessas redes.

Além do mais, as abordagens que utilizam agregação de dados nos dispositivos têm o objetivo de oferecer um serviço único para todos, que sem a agregação seria tratada individualmente. Algumas propostas utilizam agregação para definir as áreas de atuação de um determinado controlador local, porém deve ser observado que as decisões tomadas em agregações específicas podem ser diferentes entre si. Aliado a essa característica com a mobilidade dos veículos entre essas agregações virtuais a reconfiguração de regras nos dispositivos será muito grande, aumentando o número de mensagens de controle.

Além disso, muitos dos trabalhos que utilizam de agregação de dados nas suas propostas fazem para prover melhor eficiência e qualidade. A partir do crescimento e popularidade da Internet e de aplicações multimídia como *streaming* de dados, é necessário mecanismos para fornecer garantia de entrega ou priorização dos dados para melhor QoS e QoE dessas aplicações. Para tanto, os mecanismos de agregação de dados são utilizados para uniformizar e generalizar a qualidade observada pelas aplicações em execução na rede, e em redes VANETs.

A agregação de veículos virtuais, nos controladores localmente configurados nas RSUs da infraestrutura de borda da rede veicular, é definida para prover maior organização e gerenciamento por parte de SDN para o QoS dos dados que são transmitidos na rede.

4 ARQUITETURA PROPOSTA

Este capítulo apresenta o sistema proposto por esta dissertação, que tem por objetivo melhorar os níveis de qualidade de serviço e de experiência para o usuário de aplicações de *streaming* em SDVN através da priorização do tráfego e agregação de veículos por classe de aplicação. A proposta também elabora uma estratégia baseada na hierarquia de controladores SDN, com controlador central com a visão global da rede, bem como com controladores locais mais próximos aos veículos, no caso os veículos, visando reduzir tanto o tempo de comunicação das regras OF quanto o *overhead* de sinalização entre os controladores. Além disso, incluso na proposta, está um algoritmo de *handover* baseado nas informações de localização geográfica e mobilidade dos veículos, este com o objetivo de diminuir o atraso de comunicação entre os veículos e os controladores localmente distribuídos na rede veicular.

As subseções deste capítulo estão organizadas da seguinte forma. A Seção 4.1 apresenta uma visão arquitetural da proposta, contendo os elementos que a constituem bem como o modelo hierárquico de controle SDN. A descrição das características do *framework* proposto SDVNSim é feita na Seção 4.4. Os detalhes do algoritmo de *handover* de controladores locais e seu funcionamento são discriminados na Seção 4.2. Já a Seção 4.3 descreve a proposta de priorização de tráfego a partir de um esquema de agregação de veículos por classe de aplicação. Por fim, as considerações finais são realizadas na Seção 4.5.

4.1 CAMADAS ARQUITETURAIS

A arquitetura proposta possui quatro camadas, as três primeiras conforme modelo SDN, isto é, dados, controle local e controle global e a quarta camada de provedor ou serviços. Cada uma destas com um conjunto de características que devem ser observados para se atingir os objetivos propostos por esta dissertação. A seguir são descritas essas características e requisitos a partir da camada inferior até a superior. Podemos observar na Figura 9 a representação dos módulos desenvolvidos nas respectivas camadas.

4.1.1 Plano de Dados

A primeira camada refere-se ao Plano de Dados e foi desenvolvida dos carros conectados em um ambiente de rede veicular.

Nessa camada estão os carros da rede VANETs. Foi desenvolvida uma camada de *software* no veículo para que suportasse algumas aplicações. Foram desenvolvidos o submódulo de comunicação OF, o submódulo cliente de vídeo *streaming* e o submódulo de comunicação via protocolo *Real-time Transport Protocol* (RTP).

Os carros são habilitados com OF e, dessa forma, podem realizar a comunicação com o controlador SDN da rede veicular. Essa aplicação consulta e administra as tabelas de regras OF alocadas nos veículos para encaminhamento de pacotes. Como padrão, caso o carro não possua regras para uma determinada transmissão, este envia uma mensagem denominada *table miss* para o controlador local.

Para os controladores locais terem ciência da localização dos carros, uma mensagem status foi implementada. A aplicação desenvolvida para a execução das mensagens de status contém um método que leva em consideração a distância percorrida pelo determinado veículo para que este envie a mensagem de localização para o controle local. O objetivo desse modelo é indicar maior ou menor necessidade do controlador ter ciência se há ou não a iminência de handover. Por padrão nesta proposta decidimos enviar as mensagens de localização a partir da distância percorrida pelo veículo. Neste caso, definimos a distância de trinta metros percorridos pelo veículo para que este envie a mensagem de localização para o controlador local. Essas informações no controle local irão formar a visão local da rede.

4.1.2 Plano de Controle Local

Na camada correspondente ao plano de controle local encontra-se a infraestrutura de borda que proporcionará mecanismos de decisão para a operação e gerenciamento da rede. Nessa camada podem ser identificados os controladores locais, integrados a cada RSUs pertencente às VANETs e, portanto, responsáveis pelos carros sob a cobertura de tais pontos de acesso das redes veiculares. Controladores locais trocam mensagens OF com os carros servidos pelas respectivas RSUs da rede.

O módulo de controle SDN possui a aplicação OF que é responsável por processar as solicitações dos carros. Essas solicitações são enviadas através de mensagens OF. As decisões tomadas são para o gerenciamento dos níveis de prioridades dos serviços executados na rede veicular, através do algoritmo de priorização.

O algoritmo de priorização organiza as solicitações enviadas pelos carros em oito agrupamentos de níveis de prioridades. Cada nível especifica uma prioridade de transmissão específica, indo do mais baixo ao mais alto nível. Para as aplicações de *streaming*, o código de priorização tende a alocar para o agrupamento de maiores níveis de prioridade. As demais aplicações que executem na rede irão ser alocadas em níveis de prioridade menores, caso seja necessário. Outra função relacionada a esse algoritmo é a realocação de prioridades, ou seja, caso seja necessária a realocação de um carro para prioridades maiores, o algoritmo o realizará.

No controle local também é organizado, a partir das mensagens de localização dos carros, a visão local da rede. Essas visões locais são contextos parciais da região geográfica de atuação das RSUs. Elas são transmitidas para o controle central de tempos em tempos, para ser formada a visão global da rede. Por padrão nesta proposta decidimos utilizar o

tempo de dois minutos para que o controle local envie a visão local para o controle global. Nesta proposta não foi implementada a comunicação e troca de informações entre os controladores locais diretamente.

4.1.3 Plano de Controle Global

Este é o plano localizado em um nível mais alto do controle SDN, ou seja, o controlador global está localizado na nuvem do provedor de conteúdo ou ISP, podendo ter comunicação com todos os controladores locais.

O controlador global que irá organizar e processar as informações contextuais de posição e qualidade de sinal obtidas a partir das visões parciais dos controladores locais. O controle global também coordena, por exemplo, as decisões de *handover* entre controladores locais, bem como identificação e inserção do tráfego proveniente da Internet no agrupamento virtual adequado à sua respectiva classe de qualidade de serviço gerenciada pelo novo controlador local após o *handover*.

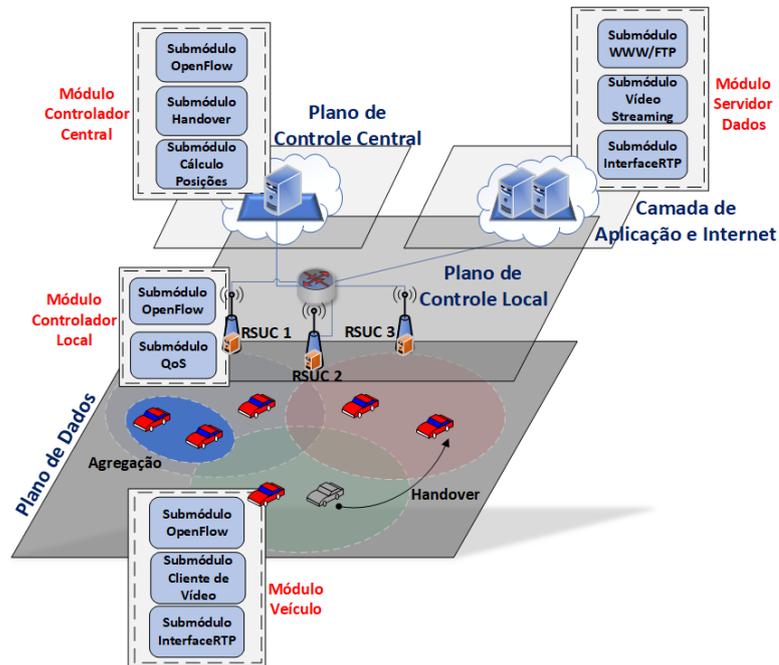
O algoritmo de *handover* desenvolvido no controlador central usa informações transmitidas pelos controladores locais, acerca de suas respectivas visões locais. Com estas informações, o controlador central calcula a distância euclidiana da posição dos veículos com os controladores locais, para assim decidir qual o valor mais próximo entre Carro-Antena.

4.1.4 Camada de Aplicação e Internet

A camada de aplicação e Internet não é específica da proposta, mas contempla o ambiente de nuvem e de servidores de origem remotos que geram o tráfego de provedores de conteúdo que fornecerão diversas aplicações às redes veiculares, como navegação na Web, *streaming* de vídeo, transferência de arquivos, entre outros.

Nessa camada temos implementados dispositivos para receber e transmitir dados. Primeiramente temos a aplicação *World Wide Web* (WEB), bem como a aplicação *File Transfer Protocol* (FTP), e a aplicação servidora de *streaming* de vídeos RTP que enviará os *frames* de vídeo para os veículos solicitantes.

Figura 9 – Arquitetura em camadas



Fonte: Elaborada pelo Autor

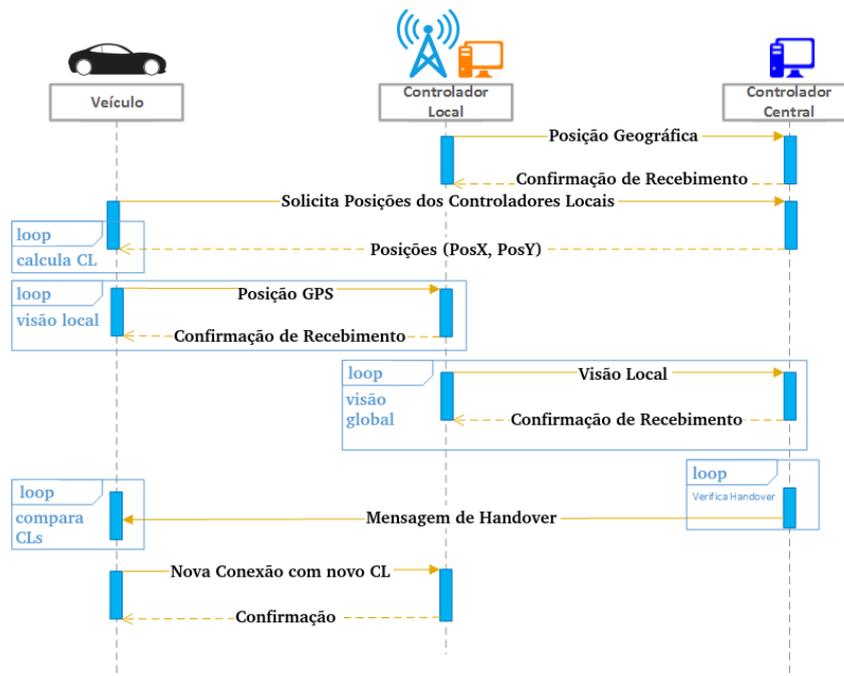
4.2 HANDOVER DE CONTROLADORES LOCAIS SDN

Um ponto bastante importante e abordado amplamente pela literatura é a mobilidade dos veículos. Essa característica pode causar desconexões frequentes dos carros com as RSUs que formam a infraestrutura de rede VANET. Aplicações que rodam sobre essa rede precisam ter soluções viáveis observando essa mobilidade para não degradar a comunicação de dados.

Em nossa solução estamos fazendo a comunicação Veículo-Road Side Unit Controller (RSUC) via protocolo OF. Inicialmente, o carro solicita as informações de localização dos controladores locais da rede ao controlador central. Este responde com os dados de posição de todos os controladores locais para o veículo. No carro foi desenvolvido um algoritmo específico para tratar esses dados calculando sua posição atual com a posição dos demais controladores locais. No veículo também foi aplicado um dispositivo que utiliza o *Global Positioning System* (GPS) para obter informações de localização. A partir disso, comparando e verificando qual controlador local está mais próximo do veículo, este será selecionado inicialmente, como o controlador para comunicação OF entre a infraestrutura e o veículo. Cabe destacar que essa verificação é realizada quando o veículo for admitido na rede veicular, para este saber com qual controlador local comunicar inicialmente. Posteriormente, após a formação das visões locais e global da rede, o algoritmo de *handover* localizado no controlador central realizará os cálculos de decisão. Para melhor visualização das etapas da execução do algoritmo desenvolvido podemos verificar na Figura 10 uma

ilustração com um diagrama de sequência da proposta. Já na Figura 11 é apresentado um fluxograma do funcionamento do algoritmo de *handover* desenvolvido nesta dissertação.

Figura 10 – Diagrama de sequência de funcionamento do Handover

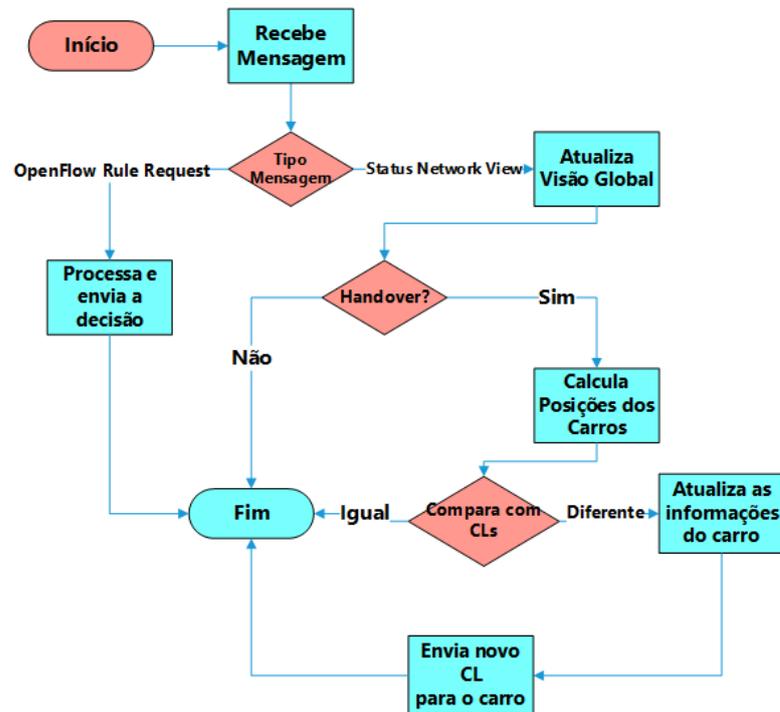


Fonte: Elaborada pelo Autor

No fluxograma são apresentadas as etapas que ocorrem quando dos procedimentos de reunião de informação, preparação e execução de *handover* entre os controladores locais. Inicialmente o controlador global verifica as mensagens OF de requisição que chegam. Verificando o tipo da mensagem, e caso esta seja uma solicitação para atualização da visão global da rede o controlador central processa a requisição. Posteriormente o controlador verifica se a funcionalidade de realização do *handover* está habilitada no sistema. Caso esteja habilitado, o controlador central calcula as informações. Essas informações são processadas calculando a posição GPS dos carros armazenados e comparando com relação aos controladores locais da rede, utilizando para tanto a distância euclidiana. Caso o resultado obtido seja a posição geográfica de um controlador local diferente ao que está atualmente atribuído no veículo, as deslocações são enviadas do controlador central para o carro específico.

Para que seja formada a visão local, os carros enviam suas posições com base em informações do GPS a cada cem metros. Este valor foi definido por padrão nesta proposta. Após a formação e atualizações das visões locais, as RSUCs enviam essas informações para o controlador central na nuvem. O controlador central, utilizando as visões recebidas, gera uma visão global. O algoritmo de *handover*, utilizando da visão global, calcula e compara as posições dos veículos em comparação com os controladores locais. Caso um determinado

Figura 11 – Fluxograma de funcionamento algoritmo desenvolvido



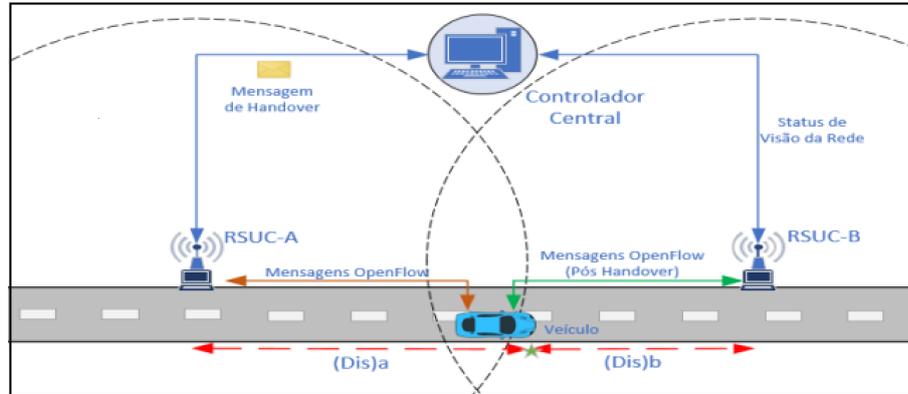
Fonte: Elaborada pelo Autor

veículo esteja mais próximo de um controlador local em relação ao que está atualmente vinculado, o controlador envia uma mensagem de *handover* para que o carro vincule a comunicação com o controlador local mais próximo.

Durante a simulação, e a mobilidade do veículo, esse carro pode se distanciar do controlador inicial a ele designado, e se aproximar de outro posicionado na rede. Nesse caso, nosso algoritmo executado no controlador central, utilizando de informações da visão global da rede, detecta que esta aproximação está em curso, e envia uma mensagem para troca automática de controlador local, sem necessidade de um novo cálculo e novas mensagens para decidir com qual controlador deve se comunicar. As informações sobre o funcionamento do algoritmo de *handover* descritas nesse parágrafo são ilustradas na Figura 12

Podemos observar na Figura 12 que após a detecção pelo controlador central que a distância do carro para com a RSUC-B, denominado pela sigla (Dis)b, está menor do que a distância do veículo para com o controlador local RSUC-A, denominado (Dis)a, o controlador central envia uma mensagem de status para o veículo para que esse realize o *handover*, e que a partir desse momento realize a comunicação OF com o controlador local RSUC-B.

Por fim, cabe ressaltar que a solução proposta por esta dissertação está utilizando o *handover* para reduzir o tempo de comunicação entre carro e os controladores locais da infraestrutura de borda da rede VANETs, bem como a diminuição de pacotes trafegados

Figura 12 – Esquema de *Handover* proposto nessa dissertação

Fonte: Elaborada pelo Autor

na rede, o que será demonstrado nas seções da avaliação.

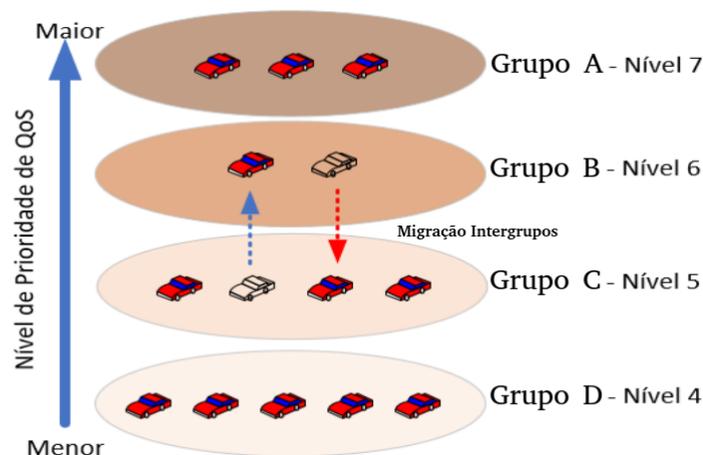
4.3 ALGORITMO DE PRIORIZAÇÃO

A solução proposta funciona com base nas informações OF de solicitação de prioridade QoS para transmissão de dados. Essas solicitações são tratadas pelos controladores locais. Para cada solicitação de prioridade de QoS, o algoritmo verifica o tipo de aplicação solicitada pelo veículo. Esta solução foi desenvolvida para priorização de dados, e foi implementada no *framework* OMNeT++, em linguagem de programação C++. Ele utiliza informações que são coletadas dos dispositivos segundo a visão da rede.

Como consequência disso, o algoritmo criará uma representação virtual do veículo que será armazenada em uma determinada classe de agregação para veículos baseada no tipo de aplicação. Para isso, por padrão na proposta desta dissertação, estamos determinando um limiar máximo de dez usuários por agregado de veículos formando o que denominamos de grupo virtual. A partir disso, caso a aplicação que solicitou prioridade seja de *streaming* de vídeo o algoritmo verificará o limiar atual da agregação com prioridade mais alta, caso não esteja no valor máximo o controlador local irá adicionar a nova requisição ao grupo virtual. Caso contrário, verificará o limiar de uma agregação imediatamente inferior, e assim sucessivamente. Apresentamos uma representação gráfica do funcionamento do algoritmo proposto na Figura 13. Nela podemos observar os veículos virtuais alocados em cada agregação com determinado nível de prioridade.

Porém, na rede veicular, tráfegarão também várias outras aplicações e serviços diferentes além de *streaming* de vídeo. Nesse caso, definimos que o algoritmo avalie a carga do controlador local, e caso os agregados de vídeo estejam abaixo de 50%, com relação à carga total, o algoritmo pode alocar níveis de prioridade altas a estas demais aplicações. Este limiar de 50% foi definido por padrão para ser utilizada nesta solução.

Figura 13 – Representação do algoritmo de agregação e priorização



Fonte: Elaborada pelo Autor

A solução também considera o *handover* na migração intergrupos. Quando acontece

um *handover* o veículo irá ser associado a um grupo, dependendo de sua prioridade, e das condições dos grupos virtuais do novo controlador local. Isto é, caso o controlador local verifique que pode alocar o veículo ao mesmo grupo de sua prioridade antes do *handover*, o algoritmo o fará. Porém, caso não seja possível a alocação no mesmo grupo o controlador irá colocar em um ligeiramente inferior enquanto não abre espaço na prioridade original antes do *handover*.

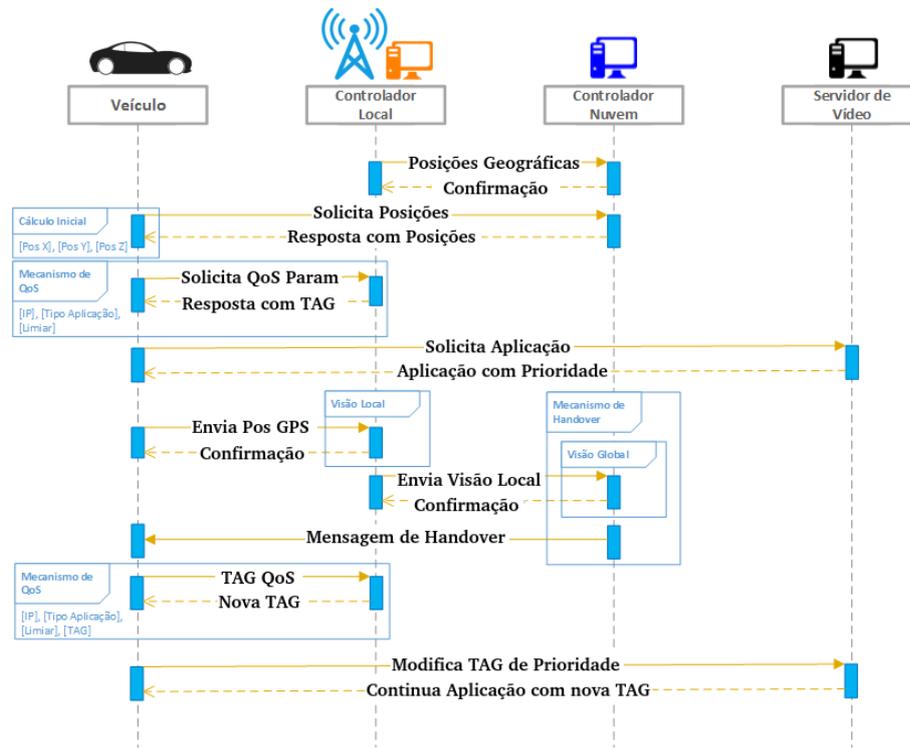
Outra característica da solução é a funcionalidade de realocação de prioridades dependendo do término de envio de dados de vídeo de um determinado veículo. Por exemplo, veículo A fez a solicitação de prioridade primeiro que o veículo B. Nesse cenário, como exemplo, o carro A foi alocado no nível de prioridade 6, enquanto o veículo B foi alocado no nível 5. Desta forma, após a transferência completa dos *frames* de vídeo do veículo A, o algoritmo realoca o veículo B para o nível superior, caso este não tenha terminado de transmitir seus *frames*. Assim, o controlador envia uma mensagem OF com a nova informação de prioridade para o veículo B, para ele receber o restante de seus *frames* em um nível superior de prioridade. Essa funcionalidade também pode ser observada na Figura 13.

O objetivo principal do algoritmo é prover suporte para o controlador SDN configurado nas RSUs, nas decisões de QoS e priorização de dados na rede. Com base na característica e especificações do protocolo 802.1p e as tags de priorização de dados, definimos esse algoritmo para prover qualidade de serviço para aplicações de *streaming* de vídeo em rede veiculares.

Ademais, podemos observar na Figura 14 um diagrama de sequência da interação entre os mecanismos que foram desenvolvidos. Podemos observar que inicialmente o carro solicita para o controlador centralizado na nuvem as informações das localizações dos controladores locais. Após obter esses dados, o carro realiza o cálculo inicial para definir com qual controlador mais próximo irá se comunicar. Cabe destacar que esse mecanismo é realizado apenas quando estes adentram a rede veicular. O objetivo é ter de início das comunicações o controlador mais próximo geograficamente. Após certo tempo, devido à mobilidade dos carros no cenário, o algoritmo de *handover* no controlador central irá realizar os demais cálculos para as demais decisões de *handover* geográfico.

Podemos ver também na mesma Figura 14 que após o *handover* o veículo precisa informar ao controlador local qual *TAG* de prioridade foi alocada para ele realizar as comunicações de prioridade. O controlador local, realiza os mesmos procedimentos descritos anteriormente para alocação e decisão de priorização. Porém, neste caso, o controlador local tenta alocar a mesma *TAG* de prioridade que o carro tinha antes de ser realizado o *handover*. O objetivo é manter a prioridade das transmissões dos *frames* de vídeo que estão sendo transmitidos. Porém, se o carro receber uma tag diferente, este informa ao servidor de vídeo o novo valor de prioridade e este agora passará a enviar os demais *frames* de vídeo que restam com a nova *TAG*.

Figura 14 – Diagrama de Sequência Geral



Fonte: Elaborada pelo Autor

4.4 SDVNSIM: *FRAMEWORK* DE SIMULAÇÃO PARA REDES VEICULARES DEFINIDAS POR *SOFTWARE*

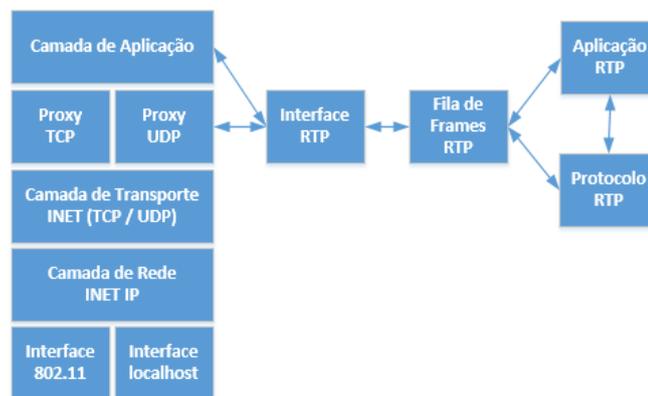
Esta seção apresenta o funcionamento dos componentes desenvolvidos no *framework* SDVNSim. O objetivo de criar um ambiente de simulação para validar a nossa proposta, foi desenvolvido um novo *framework* para redes SDVN. Essa solução contém módulos que possibilitam ao desenvolvedor criar uma rede veicular com suporte ao protocolo de comunicação OF entre os planos de dados e controle. Nosso *framework* foi desenvolvido dentro do ambiente de programação de simulações OMNeT++, contendo como base o módulo de rede INET, o simulador de mobilidade SUMO e o simulador de redes veiculares, Veins. A seguir é feita uma síntese dos módulos que foram programados. Cabe ressaltar também, que a partir do estado da arte da literatura não foi identificada outra solução que simule vídeos reais em redes SDVN.

4.4.1 Componentes

- **OFCar:** Esse é um módulo que foi desenvolvido para implementar o veículo com comunicação via protocolo OF com controladores na borda da rede RSUC. Podemos observar na Figura 15 a representação gráfica do módulo OFcar. Alguns submódulos fazem parte do módulo principal OFcar. O primeiro submódulo que destacamos é

a camada de aplicação que foi desenvolvida no carro. Esse submódulo tem os objetivos de comunicação via mensagens OF com os controladores da rede veicular, e o cálculo do controlador local mais próximo ao veículo para iniciar a comunicação. Podemos observar também na Figura 15 submódulos que tratam da *streaming* de vídeo. Temos o submódulo Interface RTP que recebe os *frames* de *streaming* da camada de transporte e informa à camada de aplicação da chegada das informações. O submódulo Fila de *Frames* RTP organiza sequencialmente os *dados* que vão chegando. Por fim, os submódulos Aplicação RTP que é a aplicação final que recebe e processa os dados de *streaming* e o Protocolo RTP que gerencia o processo de recepção dos dados.

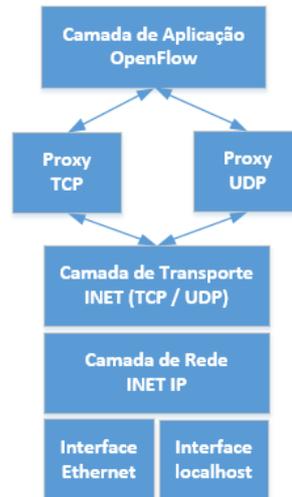
Figura 15 – Módulo OFcar



Fonte: Elaborada pelo Autor

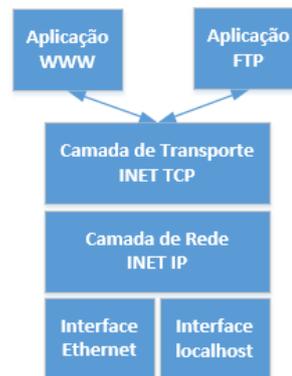
- **RSUC:** Esse é o conjunto de submódulos que formam o RSUs com controle SDN para oferecer a função de controle da rede local através da comunicação OF. A representação gráfica do módulo RSUC pode ser vista na Figura 16. Através das mensagens OF o controlador local da rede VANET mantém uma comunicação de controle para decisões de parâmetros de QoS. Nesta camada, o *framework* SDVNSim estende o simulador OMNeT++. Foi desenvolvida uma camada de aplicação com a lógica de controle SDN implementada. A implementação do módulo do controlador central utiliza a mesma ideia aqui apresentada.
- **Módulo Internet:** Esse é o módulo com um conjunto de roteadores interconectados para simular um tráfego de dados pela internet.
- **Servidor de Aplicações:** Esse é um dispositivo que foi programado para simular aplicações WEB e FTP que irão gerar dados que serão trafegados na rede veicular a partir de uma solicitação de um veículo. Podemos observar a representação gráfica do módulo servidor de aplicações na Figura 17.

Figura 16 – Módulo RSUC



Fonte: Elaborada pelo Autor

Figura 17 – Módulo Servidor de Aplicações

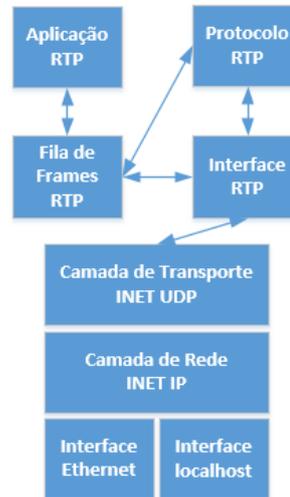


Fonte: Elaborada pelo Autor

- **Servidor de Vídeo:** Esse é um servidor que recebe ou envia *frames* de vídeo através do protocolo de mensagens *streaming* RTP. Podemos observar na Figura 18 uma representação do módulo servidor de vídeo implementado. Alguns submódulos fazem parte do módulo principal. Um destes submódulos é a aplicação RTP em que está armazenada as informações para gerar os *frames* de vídeo que serão transmitidos. Temos também a fila de *frames* RTP que organiza os dados para transmissão. Temos o módulo do protocolo RTP que gerencia a transmissão dos dados. E por fim, o módulo interface RTP que processará os dados que serão enviados com a específica métrica de QoS. Nesse módulo é implementada uma estrutura de dados que organiza os valores de QoS para seus específicos endereços dos carros solicitantes de vídeo. Ou seja, ao iniciar a transmissão do *streaming* dos dados, a interface configura os

valores de QoS específicos para os veículos de destino de cada *frame*.

Figura 18 – Módulo Servidor de Vídeo



Fonte: Elaborada pelo Autor

- **Mensagem OF:** Com base na estrutura das mensagens OF que são utilizadas neste protocolo na versão 1.3, foi desenvolvido um módulo que implementa as mensagens *OFPT_HELLO*, *OFPT_TABLE_MISS*, *OFPFC_ADD*, *OFPFC_MODIFY_STRICT* e *OFPFC_DELETE_STRICT*.

A mensagem *OFPT_HELLO* é a troca entre controlador e veículo quando uma conexão é estabelecida. Outra mensagem é a *OFPT_TABLE_MISS* que é trocada entre veículo e controlador, informando a este que estará faltando uma determinada regra de fluxo no carro. As mensagens *OFPFC_ADD* adicionarão entradas na tabela de fluxos. Já as mensagens *OFPFC_MODIFY_STRICT* modificarão entradas já existentes na tabela. Por fim, as mensagens *OFPFC_DELETE_STRICT* apagarão entradas específicas na tabela de fluxo.

Por fim, o protocolo RTP provê um intermédio de comunicação entre a camada de aplicação e a camada de transporte da rede. Tem como objetivo ordenar os *frames* de vídeo para serem repassados para a camada de aplicação cliente de vídeo.

4.5 CONSIDERAÇÕES FINAIS

Esta dissertação propõe uma arquitetura SDVN hierárquica que engloba dois níveis de controladores SDN, *handover* baseado em localização, mapeamento de priorização de fluxo na parte com fio com um esquema de agregação virtual de veículos com aplicativos que possuem requisitos similares de QoS.

Como foi observado, o sistema proposto por esta dissertação tem por objetivo melhorar os níveis de qualidade de serviço e de experiência para o usuário de aplicações de *streaming* de vídeo em redes SDVN. Essa solução é realizada através de grupos virtuais que organizam as informações de QoS dos veículos, levando em consideração o tipo de aplicação e a carga máxima de armazenamento de informações do controlador local. Não obstante, a migração das prioridades *interclusters*, leva em consideração a movimentação do veículo na rede através do *handover*.

A proposta também especifica uma estratégia de hierarquia de controle SDN baseado na sua localização mais próxima dos veículos, visando assim alcançar a diminuição do *overhead* de sinalização nos controladores.

Ademais, incluso na proposta, está um algoritmo de *handover* baseado nas informações de localização geográfica e mobilidade dos veículos, este com o objetivo de melhorar o atraso de comunicação entre os veículos e os controladores localmente distribuídos na rede veicular, baseado na mobilidade e posição geográfica desses veículos na rede.

5 AVALIAÇÃO EXPERIMENTAL

Este capítulo apresentará a avaliação dos objetivos da proposta descritos no capítulo 1. Foram utilizadas algumas extensões do *framework* OMNeT++ para simular a arquitetura, mecanismos e algoritmos propostos, bem como um ambiente de cidade realista baseado em um bairro de Recife, através de um modelo extraído de uma ferramenta do simulador Sumo, e geradores de streaming de vídeo através da ferramenta Evalvid. A seção 5.1 descreve os recursos e ferramentas utilizados, bem como a aplicação de vídeo que será utilizada nas simulações. A seção 5.2 apresenta e discute as métricas que foram escolhidas para as avaliações realizadas. Na seção 5.3 são descritos os parâmetros utilizados, bem como os valores que foram escolhidos para a execução dos experimentos. A seção 5.4 descreve o cenário simulado de um bairro da cidade de Recife utilizado nos experimentos. Na seção 5.5, são apresentados e discutidos os experimentos realizados, primeiramente analisamos o impacto do *overhead* de sinalização, depois verificamos a qualidade de experiência observada da proposta de priorização e o atraso de reconexão em um cenário de alto *overhead* de sinalização. Por fim, na seção 5.6 são feitas as considerações finais acerca dos experimentos avaliados e resultados.

5.1 RECURSOS UTILIZADOS

O *framework* escolhido como base de construção do SDVNSim e do cenário das simulações dessa dissertação foi o OMNeT++¹. Esse *framework* é uma biblioteca e estrutura de simulação C++ extensível, modular e baseada em componentes, principalmente para construir simuladores de rede.

O INET *Framework*² é uma biblioteca de modelo de código aberto para o ambiente de simulação do OMNeT++. Fornece protocolos, agentes e outros modelos para pesquisadores e estudantes que trabalham com redes de comunicação. Ele contém modelos para a pilha de protocolos da Internet (TCP, UDP, IPv4, IPv6, OSPF, BGP, etc.), protocolos de camada de rede com e sem fios (Ethernet, *Point-to-Point Protocol* (PPP), IEEE 802.11, etc), suporte para mobilidade, protocolos *Mobile ad hoc network* (MANET), *Differentiated Services* (DiffServ), *Multiprotocol Label Switching* (MPLS) com sinalização *Label Distribution Protocol* (LDP), *Resource Reservation Protocol - Traffic Engineering* (RSVP-TE) e diversos modelos de aplicativos.

O SUMO³ é um pacote de simulação de tráfego rodoviário projetado para lidar com grandes redes rodoviárias. Ele permite simular como uma determinada demanda de tráfego equivalentes em veículos individuais que se movem através de uma determinada rede

¹ OMNeT++: <https://omnetpp.org/>

² INET: <https://inet.omnetpp.org/>

³ SUMO: <http://sumo.sourceforge.net/>

rodoviária. A simulação permite abordar um grande conjunto de tópicos de gerenciamento de tráfego. É puramente microscópico, ou seja, cada veículo é modelado explicitamente, tem uma rota própria e se move individualmente pela rede.

O *framework* VEINS⁴ é uma plataforma open source para executar simulações de redes veiculares. Ele é baseado nos *frameworks* descritos anteriormente, o OMNeT++ e SUMO.

Um *software* de avaliação do vídeo trafegado na rede foi utilizado.. Essa aplicação é denominada Evalvid ⁵, e é bastante utilizada para avaliação de aplicações de *streaming* (KLAUE; RATHKE; WOLISZ, 2003). Ele é um kit de ferramentas para uma avaliação unificada da qualidade da transmissão de vídeo. O EvalVid tem uma estrutura modular, possibilitando a troca, a critério dos usuários, tanto do sistema de transmissão subjacente quanto dos codecs, por isso é aplicável a qualquer tipo de esquema de codificação e pode ser usado tanto em experimentos reais como em experimentos de simulação.

As medições de qualidade de vídeo digital devem basear-se na qualidade percebida do vídeo real que está sendo recebido pelos usuários do sistema de vídeo digital. Visto isso o Evalvid utiliza métricas de QoE que servem como extensão aos parâmetros de QoS e fornecem avaliação das redes de modo mais próximo à percepção dos usuários, permitindo avanços nas transmissões de aplicações de áudio e vídeo e podem proporcionar melhorias nos protocolos. Uma das métricas mais usadas de QoE é o PSNR (*Peak Signal to Noise Ratio*). Ele estima a qualidade do vídeo, em decibéis, comparando a transmissão original com a obtida no receptor (WU; RAO, 2005). Para cada faixa de valores de PSNR, há uma qualificação para o vídeo que foi recebido pelo usuário conforme mostra a Tabela 5.

Tabela 5 – Qualidade estimada de vídeo a partir do PSNR

PSNR (dB)	>37	31 - 37	25 - 31	20 - 25	<20
Qualidade	Excelente	Bom	Aceitável	Baixo	Péssimo

Para avaliação do PSNR foi utilizado o vídeo Akiyo (University)⁶. Todas as simulações foram feitas com o vídeo da mesma duração, codec, número de quadros (300 quadros) e resolução (352x288). Foi escolhido o vídeo Akiyo, pois através da leitura na literatura que utiliza a ferramenta Evalvid esse vídeo é o mais comum nas avaliações em simulações de QoE e, com isso, os resultados podem servir de comparação para outros trabalhos. O *framework* Evalvid foi integrado ao simulador SDVNSim desenvolvido para permitir envio e avaliação de vídeos.

5.2 MÉTRICAS

Nesta Seção são apresentadas as métricas utilizadas para avaliar o desempenho da proposta desta dissertação.

⁴ VEINS: <https://veins.car2x.org/>

⁵ Evalvid: <http://www2.tkn.tu-berlin.de/research/evalvid/fw.html>

⁶ Akiyo Video: <https://www.youtube.com/watch?v=onfjPHNU9EM>

- **Overhead de Sinalização:** A quantidade de mensagens de controle OF que é recebida por cada controlador em cada cenário de abstração de controle definida nessa dissertação.
- **Qualidade de Experiência:** Níveis de PSNR utilizados para verificar a qualidade de experiência obtida na transmissão de vídeo com prioridades específicas desenvolvidas nessa proposta.
- **Atraso de Reconexão:** Tempo para que um carro se reconecte à rede e inicie as comunicações OF com um novo controlador local, após a realização do *handover* geográfico. Essa métrica é usada para avaliação do esquema de *handover* de sinalização proposto.

5.3 PARÂMETROS

- **Velocidade máxima dos veículos (m/s):** Velocidade máxima alcançada pelos veículos nos cenários das simulações realizadas por esta dissertação. A variação da velocidade dos veículos possibilita estudar o comportamento de nossa proposta perante diferentes níveis de velocidade dos carros no cenário.
- **Potência do sinal RSUCs e veículos (mW):** Potência do sinal em miliwatts para os dispositivos RSUCs e veículos nos cenários avaliados. Escolhemos esse parâmetro pois cenários com potências diferentes permitem estudar o comportamento da proposta diante de coberturas distintas de rádio.
- **Quantidade de veículos:** Quantidade máxima de veículos que trafegam no cenário no momento da simulação.
- **Área dos cenários (km²):** Área do cenário proposto para a avaliação dos algoritmos propostos por esta dissertação.
- **Tempo de Simulação:** O tempo de simulação com que as simulações foram executadas nos experimentos.

Com o objetivo de validar a escolha dos parâmetros elencados anteriormente, os utilizados nesta dissertação foram escolhidos a partir da leitura de artigos que os utilizam para avaliação de desempenho de suas propostas. Os artigos levantados como referência na escolha foram MARTINEZ et al. (2009), LEE; OH; GERLA (2012), DUAN; LIU; WANG (2017), LUO et al. (2017), ALLOWISH et al. (2017), SKONDRAS et al. (2017), MA; KANELOPOULOS; TRIVEDI (2017), ARIF; WANG; BALAS (2018) e PATEL; KAUSHIK (2018). Os parâmetros acima elencados são usualmente utilizados para avaliações de desempenho de redes veiculares, e no caso desta dissertação esses parâmetros influenciam nos resultados das métricas mencionadas anteriormente.

5.3.1 Fatores e Níveis

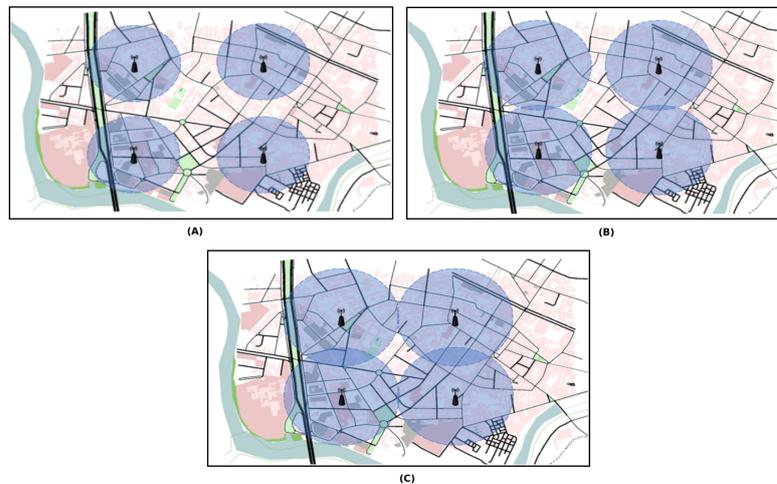
Para a efetivação e avaliação das simulações que foram executadas, foram escolhidos alguns fatores e seus valores de níveis com base na literatura. A velocidade máxima atingida por um veículo na via em metros por segundo, de acordo com SKONDRAS et al. (2017), MARTINEZ et al. (2009) e LUO et al. (2017). A potência de sinal das antenas das RSUs e dos veículos em miliwatt segundo ALOWISH et al. (2017) e DUAN; LIU; WANG (2017). A quantidade máxima de carros que trafegam por vez na área simulável segundo LEE; OH; GERLA (2012), MA; KANELOPOULOS; TRIVEDI (2017) e ARIF; WANG; BALAS (2018). A área em quilômetros quadrados da simulação segundo MARTINEZ et al. (2009) e ALOWISH et al. (2017). E o tempo das simulações realizadas em segundos, de acordo com ALOWISH et al. (2017) e PATEL; KAUSHIK (2018). A Tabela 6 ilustra os níveis que foram atribuídos para cada fator.

Tabela 6 – Fatores e Níveis especificados para as avaliações

Fatores	Velocidade (m/s)			Potência do Sinal dos Veículos (mW)			Potência do Sinal das RSUs (mW)			Quantidade de Carros	Área dos Cenários (km ²)	Tempo de Simulação (s)
	6	9	12	3	4	5	3	4	5			
Níveis	6	9	12	3	4	5	3	4	5	200	2	500s

Podemos observar também na Figura 19 a representação dos sinais das antenas RSUC no determinado mapa do bairro de Boa Vista. Com os valores de 3mW para (A), 4mW para (B) e 5mW para (C), respectivamente.

Figura 19 – Representação das potências dos sinais utilizados



Fonte: Elaborada pelo Autor

5.4 CENÁRIO DE EXECUÇÃO DOS EXPERIMENTOS

O cenário de avaliação utilizado tem como base as vias do bairro da Boa Vista situado em Recife-PE, Brasil. Como podemos observar na Figura 20, ilustra à esquerda o mapa via satélite do bairro, e ao lado direito um ambiente simulável que foi utilizado para construir os modelos propostos nesta dissertação.

Figura 20 – Mapas representativos do bairro da Boa Vista na cidade de Recife-PE



Fonte: Elaborada pelo Autor

Para tanto, para a conversão do mapa foi necessária a utilização de um *software* específico, que é disponibilizado em conjunto com a plataforma SUMO, denominado OSM Web Wizard⁷. Esse *software* permite a especificação de uma determinada área geográfica para simulação, bem como a especificação da quantidade de veículos que irão trafegar nela.

5.5 EXPERIMENTOS

Essa seção apresenta três experimentos e analisa os resultados nas execuções. Em cada um deles foram utilizados diversos cenários. Primeiramente é avaliado o *overhead* de sinalização de dois níveis de abstração de controladores SDN utilizados na arquitetura proposta. Segundo, avaliamos o impacto de QoE em aplicações de vídeo *streaming*, utilizando o modelo de priorização proposto. Por fim, realizamos uma análise do atraso de reconexão entre veículos e RSUCs no *handover* geográfico entre controladores locais, em um cenário de grande *overhead* de sinalização.

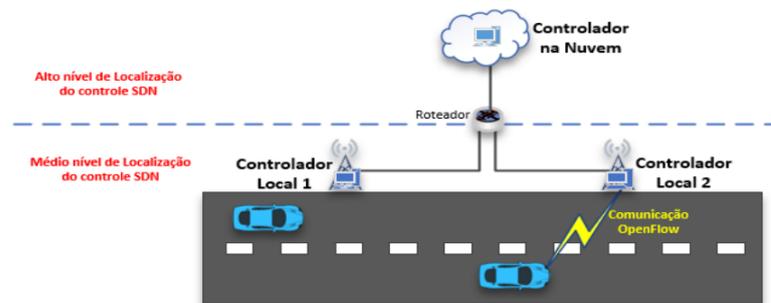
5.5.1 Experimento 1 - *Overhead* de sinalização em arquiteturas de localização de controle SDN

O objetivo desse experimento é analisar o impacto do *overhead* de sinalização em duas diferentes abordagens de localização de controladores em redes SDVN. Para isso, avaliamos o *overhead* de sinalização no controlador centralizado na nuvem utilizado na proposta de *handover* geográfico, e os controladores locais nas RSUCs utilizados para decisões de QoS.

⁷ OSM: <http://sumo.dlr.de/wiki/Tutorials/OSMWebWizard>

Podemos observar na Figura 21 uma representação da topologia utilizada para avaliação do *overhead*.

Figura 21 – Representação gráfica da topologia utilizada nas simulações de Overhead

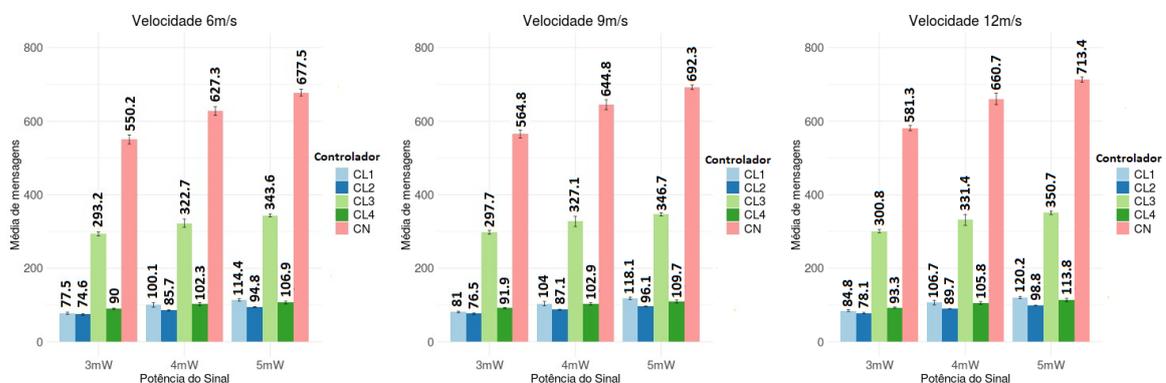


Fonte: Elaborada pelo Autor

Em nossos experimentos, os veículos solicitam informações de prioridade de dados via mensagens OF. Além disso ocorre a comunicação dos veículos e controladores locais com o controlador centralizado na nuvem através do mecanismo de *handover*. Nesse caso foi avaliado o impacto de *overhead* de sinalização nas duas posições distintas, na nuvem e localmente nas RSUs. Cabe destacar que o *overhead* de sinalização durante a configuração inicial na comunicação do veículo com o controlador global foi levado em consideração nos experimentos.

Os valores médios das quantidades de mensagens OF que foram recebidas por cada controlador SDN em sua respectiva localização pode ser observada na Figura 22. Esses valores médios foram calculados a partir dos dados obtidos em 30 simulações para os fatores e níveis especificados nessa seção.

Figura 22 – Quantidade média de pacotes OpenFlow recebidos. (CL - Controlador Local, CN - Controlador na Nuvem)



Fonte: Elaborada pelo Autor

Foi observado que em cenários que utilizam o plano de controle mais próximo da rede de dados ocasiona em diminuição da quantidade de mensagens que os controladores locais

irão receber. Podemos observar essa redução na Tabela 7.

Ou seja, a abordagem de controle SDN distribuído na rede VANETs, e mais próximo ao plano de dados, ocasiona menores níveis de *overhead* de sinalização entre os controladores.

Cabe mencionar que a quantidade de mensagens de controle recebidas pelo controlador local 3 (CL3) são superiores às recebidas pelos demais. Isso ocorre pois na região de abrangência desse controlador acontece um congestionamento em uma via bastante movimentada. Isto acaba ocasionando uma maior quantidade de mensagens de controle entre os veículos e o CL3, pelo tempo que os carros estão na região.

Contudo, ainda podemos observar que a quantidade de redução de mensagens de controle na abordagem de localização distribuída é bastante superior, alcançando níveis de redução acima de 80% com movimentação normal, e acima de 45% em condições de congestionamento.

Tabela 7 – Porcentagem de redução de mensagens de controle. Comparação Individual com cada CL em relação ao central.

Velocidade	Controlador Local	Potência do Sinal		
		3mW	4mW	5mW
6m/s	CL1	85,9%	84,0%	83,1%
	CL2	86,4%	86,3%	86,0%
	CL3	46,7%	48,5%	49,2%
	CL4	83,6%	83,6%	84,2%
9m/s	CL1	85,6%	83,8%	82,9%
	CL2	86,4%	86,4%	86,1%
	CL3	47,2%	49,2%	49,9%
	CL4	83,7%	84,0%	84,1%
12m/s	CL1	85,4%	83,8%	83,1%
	CL2	86,5%	86,4%	86,1%
	CL3	48,2%	49,8%	50,8%
	CL4	83,9%	83,9%	84,0%

A partir disso, observamos uma maior quantidade de mensagens de controle em controladores localizados na nuvem. Assim, em nossa arquitetura devido à distribuição dos controladores locais em pontos determinados da rede podemos observar uma quantidade média de mensagens menor. Concluímos que essa diminuição das mensagens de controle pode ocasionar menores níveis de processamento dos controladores da rede VANETs.

Ademais, outra observação a ser feita através da análise dos resultados mostrados na Figura 22 é que os níveis de potência de sinal têm impacto na quantidade de mensagens recebidas pelos controladores locais.

Para a velocidade máxima de 6m/s e comparando a porcentagem de aumento da quantidade de mensagens observadas com os dispositivos com potência de sinal de 3mW

para 4mW observamos um aumento de 14.1% das mensagens de controle recebidas. Já comparando de 4mW para 5mW temos um aumento de 8,0%.

Para a velocidade de 9m/s, e realizando as mesmas comparações descritas anteriormente, temos que de 3mW para 4mW observamos um aumento de 13,5% das mensagens de controle recebidas. Já de 4mW para 5mW temos um aumento de 7,9%.

Já para a velocidade de 12m/s, e comparando a porcentagem de aumento da quantidade de mensagens observadas com os dispositivos com potência de sinal de 3mW para 4mW temos um aumento de 13,7% das mensagens de controle recebidas. Já de 4mW para 5mW temos um aumento de 7,8%.

Concluimos através desses resultados, que isso pode ocorrer pelo motivo de o veículo possuir maior tempo de comunicação com a infraestrutura de rede VANETs, possuindo maior potência do sinal.

Outra observação feita dos dados é do impacto da velocidade dos veículos na quantidade de mensagens processadas. Podemos observar na Tabela 8 que em velocidades menores a quantidade de mensagens de controle reduz.

Tabela 8 – Porcentagem de redução de mensagens de controle. Comparação agrupando os valores dos controladores locais.

Velocidade	Potência do Sinal		
	3mW	4mW	5mW
6m/s	2,70%	2,63%	2,62%
9m/s	3,13%	3,67%	3,13%
12m/s	4,18%	4,10%	4,19%

Podemos concluir a partir disso, que como nosso algoritmo de *handover* está executando no controlador na nuvem, o excedente das mensagens de controle também entrará na conta das quantidades de mensagens recebidas. Ou seja, observando a Tabela 8, essa será a porcentagem de mensagens a mais que o controlador na nuvem está recebendo para processar o *handover* geográfico. Com isso, vemos uma leve diminuição da porcentagem de mensagens de *handover* em velocidades menores, ou seja, reforçando a conclusão de que em velocidades menores temos diminuição das mensagens OF processadas nos controladores.

Concluimos com isso que em níveis menores de velocidade o veículo tende a não se aproximar com muita frequência de outros controladores locais, ou seja, necessitando pouco da atuação do controlador central na nuvem para gerenciar a comunicação dos veículos sempre com os controladores locais mais próximos, e assim, diminuindo a quantidade de mensagens OF trafegadas.

Resumindo, concluimos através das análises realizadas nessa seção que abordagens que utilizam o controle SDN próximo dos veículos apresentam menores níveis de *overhead* de sinalização de mensagens OF nos controladores, em comparação com a abordagem cen-

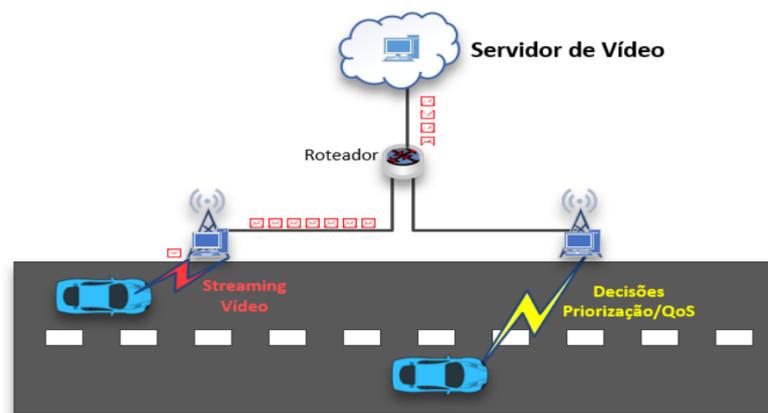
tralizada na nuvem. Também concluímos que mesmo com a quantidade de mensagens OF excedentes de *handover* do algoritmo proposto tem níveis baixos de impacto no *overhead* observado nos cenários avaliados.

5.5.2 Experimento 2 - QoE em aplicação de vídeo streaming

O objetivo desse experimento é verificar a viabilidade do algoritmo de priorização proposto, em melhorar a qualidade de QoE nos *frames* e vídeos trafegados na arquitetura desenvolvida. Nosso objetivo principal é aplicar níveis de prioridades maiores para aplicações de vídeo *streaming* para assim manter níveis de qualidade de experiência aceitáveis para os *frames* de vídeo trafegados na rede. A partir disso, nossa avaliação verifica o impacto de qualidade de experiência observada.

Com isso, e analisando os dados obtidos através dos fatores definidos obteve se uma série de resultados. Estamos utilizando 95% de nível de confiança em nossas análises e para a coleta dos dados de PSNR executamos 30 vezes as simulações. Podemos observar na Figura 23 a seguir a representação da topologia utilizada para a avaliação do QoE de vídeo.

Figura 23 – Representação gráfica da topologia utilizada nas simulações de QoS



Fonte: Elaborada pelo Autor

Para a consolidação dos experimentos desta seção foi necessária a utilização da ferramenta Evalvid para a configuração do vídeo real que foi trafegado na rede veicular. E o cálculo do PSNR relativo à qualidade de experiência obtida ao fim de cada simulação.

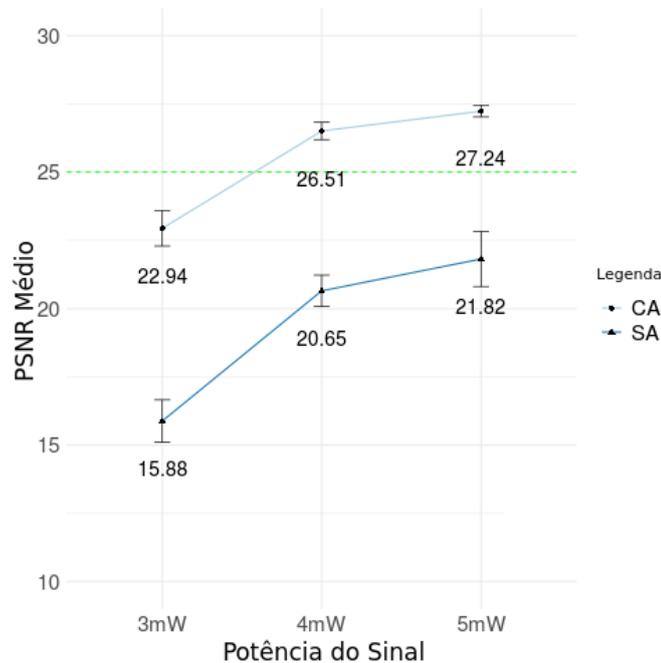
Observamos através das análises dos dados coletados, valores médios de PSNR em cenários que utilizam o algoritmo proposto, em comparação com a execução sem o esquema de priorização.

Foi verificado que utilizando o algoritmo de agregação de prioridades e alocação de QoS proposto, a qualidade do vídeo transmitido durante a simulação manteve níveis constantes de QoE acima do valor 25, que é considerado um nível de qualidade bom, enquanto que

em simulações que não utilizam o algoritmo proposto não mantêm valores constantes de qualidade, ficando abaixo do nível de qualidade bom. Podemos verificar na Figura 24 os valores que foram coletados de PSNR médio das simulações executadas.

Para a verificação do impacto que a potência de sinal tem na QoE calculada, agrupamos os resultados com base na potência de sinal. Com isso obtivemos a Figura 24. Para tanto, em termos numéricos, e verificando a porcentagem de aumento na QoE, podemos observar que para a potência de sinal de 3mW nossa proposta obteve ganho de 44.4% nos níveis de PSNR médios. Já para a potência de sinal de 4mW nossa proposta obteve ganhos de 28.3%. Por fim, para a potência de 5mW nossa proposta obteve ganhos de 24.8% nos níveis de PSNR.

Figura 24 – Comparação dos resultados de PSNR obtidos. (CA - Com Algoritmo de Prioridades, SA - Sem Algoritmo de Prioridades)



Fonte: Elaborada pelo Autor

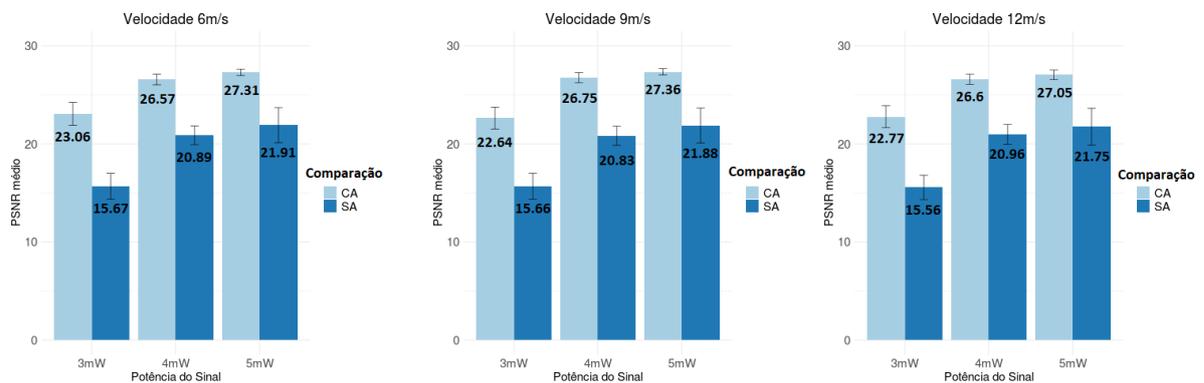
Concluimos assim que nossa solução, a partir dos níveis de PSNR, os resultados coletados ficaram com níveis considerados bons para a qualidade de experiência de vídeo. Podemos levar em consideração o mecanismo de *handover* geográfico que é realizado e que foi descrito anteriormente. Pois assim, os carros, mesmo modificando a comunicação com outros controladores locais, o algoritmo de priorização mantém níveis de priorização elevados para *streaming* de vídeo.

Cabe destacar na Figura 24 que para a potência de sinal de 3mW os valores de PSNR ficaram abaixo do nível 25, considerado um resultado de qualidade de experiência de vídeo baixa. A porcentagem de redução da qualidade comparando com o nível mínimo de PSNR 25 foi 8.2%. Porém, mesmo assim, os cenários utilizando o algoritmo de priorização,

mantiveram níveis de qualidade maiores do que cenários que não utilizam. A porcentagem de redução de qualidade para estes, sem a estratégia de priorização, foi de 36.4%.

Através das análises realizadas observamos que o algoritmo de priorização consegue manter níveis aceitáveis de PSNR. Isso pode ser explicado através do tempo que o veículo irá permanecer conectado a uma determinada torre RSUs, ou seja, não causando desconexões constantes durante a transmissão dos *frames* de vídeo a partir do servidor na nuvem.

Figura 25 – Análise estatística para as avaliações de QoE. (CA - Com Algoritmo de Prioridades, SA - Sem Algoritmo de Prioridades)



Fonte: Elaborada pelo Autor

Outra observação que podemos fazer com relação aos resultados coletados pode ser visto na Tabela 9. Nesta tabela pode ser observado que para a potência de sinal de 3mW os ganhos de QoE estão acima de 45%. Já para a potência de sinal de 4mW e 5mW, nossa proposta manteve ganhos de PSNR acima de 25%.

Cabe destacar também que a realocação das *TAGs* de prioridade está afetando nos ganhos de qualidade. Ou seja, por mais que os veículos se desloquem mais rapidamente, e conseqüentemente modifiquem a comunicação entre os controladores locais, o algoritmo de priorização, com base na *TAG* antes do *handover* ser realizado, está mantendo ganhos de qualidade maiores que 25% no cenário analisado.

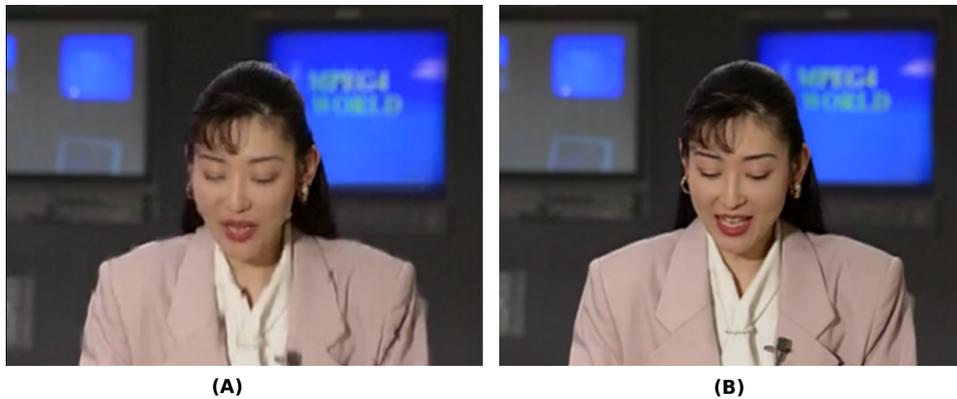
Tabela 9 – Comparação individual, observando as velocidades máximas, dos ganhos de PSNR

Velocidade	Potência do Sinal		
	3mW	4mW	5mW
6m/s	47,1%	27,1%	24,6%
9m/s	44,5%	28,4%	25,0%
12m/s	46,3%	26,9%	24,3%

Também é possível observar a qualidade do vídeo real simulado nos cenários avaliados. Podemos observar na Figura 26 a comparação entre (A) que é uma cena do vídeo

trafegando sem o esquema de priorização, e (B) que é uma cena do vídeo simulado com o algoritmo de priorização e QoS proposto.

Figura 26 – Comparação de uma cena do vídeo simulado.



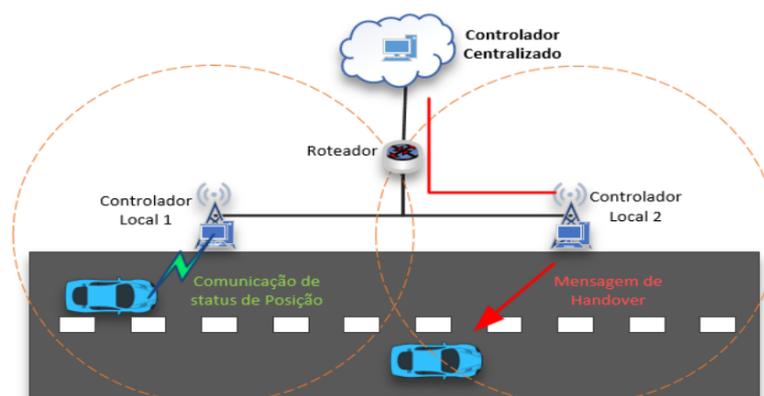
Fonte: Elaborada pelo Autor

Através dessas duas imagens é possível perceber uma maior qualidade no frame (B) em comparação com o frame (A). Percebemos a imagem mais nítida, e menos borrada, o que evidencia uma melhor qualidade de experiência da proposta de priorização de dados.

5.5.3 Experimento 3 - Atraso de reconexão Veículo-RSUC após *handover* geográfico, em ambiente de grande *overhead* de sinalização

Observamos no experimento 1 descrito anteriormente que para a velocidade máxima de 12m/s obtivemos um maior *overhead* de sinalização. Neste caso decidimos analisar o impacto deste resultado de *overhead* no atraso de reconexão dos carros entre as antenas RSUCs.

Figura 27 – Representação gráfica da topologia utilizada nas simulações de Handover



Fonte: Elaborada pelo Autor

Calculamos no experimento 1 que os veículos na velocidade máxima de 12m/s a quantidade de mensagens de controle aumenta, aumentando o *overhead* de sinalização. A partir disso, verificamos o impacto desse resultado na reconexão durante o *handover* entre controladores locais. Podemos observar na Figura 27 a representação gráfica da topologia utilizada para a avaliação.

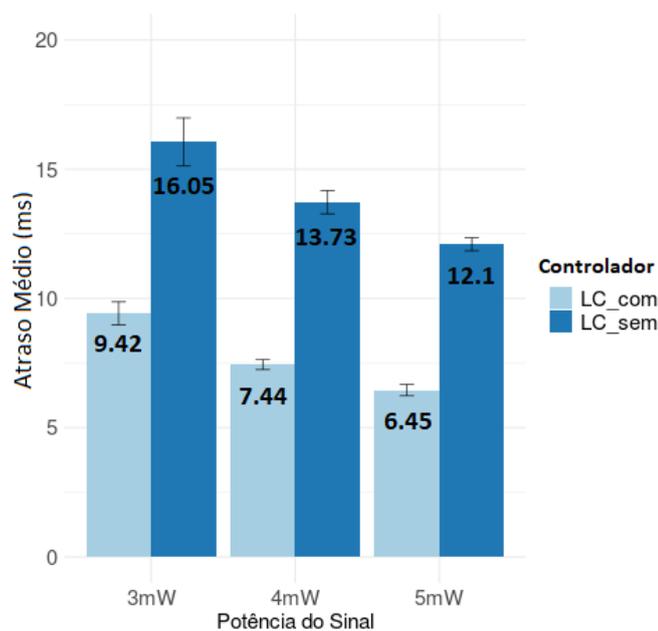
Através da análise feita dos resultados coletamos alguns dados e observações nos gráficos modelados. Os resultados obtidos consideram intervalo de confiança de 95%.

Para o *handover* estamos agrupando os resultados coletados do atraso de transmissão das mensagens OF nas antenas RSUCs. Nesse caso, decidimos agrupar os valores coletados com base na potência do sinal dos dispositivos. Estamos usando esse agrupamento para comparar e verificar o impacto na abordagem utilizando o algoritmo de *handover* desenvolvido e sem a sua utilização.

A partir dos resultados coletados, observamos que na velocidade com alto *overhead* de sinalização, variando as potências de sinal dos dispositivos, obtivemos diferentes tempos médios de reconexão.

Calculamos que na potência de sinal maior dos dispositivos ocorre maiores níveis de redução do tempo de conexão. Tivemos redução de 41,3%, 45,8% e 45,9% da proposta em relação ao *baseline*, para as potências de sinal de 3mW, 4mW e 5mW, respectivamente. Podemos observar que quanto maior a potência do sinal maior a redução do tempo para reconexão na Figura 28.

Figura 28 – Tempo de reconexão médio observado. (LCsem - Controlador Local sem algoritmo, LCcom - Controlador Local com algoritmo)



Fonte: Elaborada pelo Autor

Realizamos também a análise para verificar a redução entre os níveis de potência de sinal, dos resultados utilizando a proposta. De 3mW para 4mW tivemos redução de 21,0% e de 4mW para 5mW tivemos redução de 13,3%. Já para sem a proposta, de 3mW para 4mW observamos redução de 14,4% e de 4mW para 5mW tivemos redução de 11,8%.

Através das análises realizadas anteriormente podemos concluir que com potências de sinal maiores o controlador centralizado na nuvem atua menos vezes. Isso ocorre já que o veículo trocará menos vezes de controladores locais, vindo que é maior a porcentagem de redução do tempo de reconexão em potência de sinal mais altas. Podemos observar na Figura 28 que os níveis do atraso de reconexão utilizando o algoritmo proposto obtiveram tempos de atraso médio abaixo de cenários que não utilizam esquemas de gerenciamento de *handover*.

Por fim, concluímos que utilizando o algoritmo proposto obteremos menores níveis de atraso para reconexão dos veículos no cenário com grande *overhead* de sinalização nos controladores locais. Ocasionalmente na diminuição do tempo para que o veículo receba as decisões SDN através das mensagens OF. Nesse caso, através do algoritmo de *handover* geográfico obtemos melhor qualidade e agilidade na transmissão das mensagens de decisão OF.

5.6 CONSIDERAÇÕES FINAIS

Neste capítulo estamos realizando três avaliações para verificar a eficiência das propostas. Primeiramente elencamos e discutimos sobre a diminuição do *overhead* de sinalização da nossa proposta de localização do controle SDN mais próximo dos veículos. Observamos reduções acima de 80% em condições de mobilidade normal e acima de 45% em congestionamento.

Outra avaliação foi o de qualidade de experiência dos usuários para com a aplicação de *streaming* de vídeo executada na rede veicular. Com isso, verificamos que utilizando nossa proposta de priorização e agregação alcançamos níveis de qualidade do vídeo transmitido acima de 25%.

Por fim, avaliamos o atraso da reconexão dos veículos para com as antenas da infraestrutura de borda. Verificamos que nos experimentos do *overhead* de sinalização no nível de velocidade máxima maior temos mais *overhead* de sinalização. Com isso, utilizando nossa proposta de *handover* geográfico tivemos redução do tempo para reconexão acima de 40% em comparação com a *baseline*.

6 CONCLUSÃO

Este Capítulo resume os principais pontos propostos e discutidos nesta dissertação. Na Seção 6.1 as considerações finais deste trabalho são apresentadas. Na Seção 6.2, são descritas as contribuições obtidas. Na Seção 6.3 são apresentadas as limitações encontradas para a realização das propostas bem como algumas dificuldades. Por fim, na Seção 6.4 serão discutidos alguns pontos que podem ser explorados em trabalhos futuros.

6.1 CONCLUSÕES

Primeiramente este trabalho apresentou uma arquitetura hierárquica baseada em um gerenciamento SDN mais próximo do plano de dados da rede VANET. Ou seja, foi proposto um esquema onde os controladores SDN estão alocados nas antenas RSUs na borda da infraestrutura de comunicação, caracterizando assim comunicações V2I.

Essa proposta foi concebida a partir da leitura da literatura que aborda as redes VANETs e a constatação que grande parte das soluções que utilizam o gerenciamento de rede SDN, utilizam o controlador centralizado na nuvem. Visto isso, este trabalho avaliou o impacto do *overhead* de sinalização de mensagens de controle OF na abordagem na nuvem e proposta por este trabalho próximo do plano de dados da rede VANETs.

Através dos resultados coletados observamos que a abordagem proposta neste trabalho apresentou níveis de *overhead* de sinalização menores do que em abordagens que utilizam o controle SDN centralizado na nuvem.

Este trabalho também apresentou um esquema de *handover* geográfico entre controladores locais SDN. Essa ideia foi concebida a partir da leitura da literatura sobre propostas que utilizam os controladores SDN próximos às redes VANETs. Foi observado que durante a mobilidade do veículo na rede este tende a se distanciar dos controladores onde está sendo feito o gerenciamento de seus dados trafegados na rede, causando distanciamento da comunicação do controlador local com o veículo, e assim maior atraso de transmissão de mensagens de controle SDN. Visto isso, este trabalho propôs um algoritmo alocado em um controlador centralizado que gerencia a comunicação controlador-veículo com base na proximidade geográfica destes.

Através das análises realizadas dos dados coletados das simulações executadas verificamos que nossa solução obteve níveis de atraso médio na comunicação média de mensagens de controle OF menores em comparação com cenários que não utilizam o esquema de *handover*.

Por fim, este trabalho também propôs um algoritmo de priorização de dados de aplicações de *streaming* de vídeo para suprir os requisitos de aplicações de vídeo com níveis de qualidade de experiência do usuário para com a aplicação aceitáveis. Observamos que

atualmente vem aumentando gradualmente o consumo de dados de *streaming* em aplicações como vídeo. Visto isso, propomos um esquema que prioriza esses dados de *streaming* em redes VANETs com uma abordagem de gerenciamento SDN, através do agrupamento de prioridades.

Foi observado, através dos resultados coletados, que nossa proposta mantém níveis aceitáveis de QoE para uma aplicação de *streaming* de vídeo executando na rede VANETs. Porém, observamos que a potência do sinal impacta na qualidade, sendo que em ocasiões que o sinal dos dispositivos está baixo, nosso algoritmo não consegue manter os níveis bons de qualidade. Porém, mesmo assim, em comparação com cenários que não utilizam a priorização de dados, em ocasiões de potência de sinal baixo nossa proposta apresentou resultados mais satisfatórios.

6.2 CONTRIBUIÇÕES

Podemos destacar como contribuição principal, propor uma arquitetura hierárquica baseada em agregação de dispositivos para priorização de dados e QoS em redes veiculares definidas por *software*.

Primeiramente destacamos algumas ferramentas desenvolvidas que ficam como contribuições à academia para futuras pesquisas: (1) Um novo *framework* para simulações de redes SDVN denominado SDVNSim; (2) Um ambiente simulável de tráfego de *streaming* de vídeo real;

E as contribuições para o avanço do conhecimento sobre redes VANETs temos: (1) Um sistema baseado em grupos para decisões de QoS através do gerenciamento SDN; (2) Uma estratégia de conectividade baseada em mensagens OF simuladas; (3) Um esquema e algoritmo de *handover* geográfico entre controladores SDN localizados próximos ao plano de dados VANETs; (4) Uma análise sobre o *overhead* de sinalização em duas abordagens de localização do controle SDN.

Por fim, contribuímos com alguns códigos programados para a configuração e análise de algumas propostas. No Apêndice A desta dissertação podemos observar dois códigos em linguagem Shell Script. O primeiro foi construído com o intuito de transformar os *traces* de vídeo real processados pela ferramenta Evalvid em *traces* aceitos pelo *framework* OMNeT++, pois o Evalvid apenas mantém suporte ao código para *traces* simulados no simulador NS3. O segundo código é para a análise do vídeo simulado e cálculo dos níveis de PSNR e QoE.

6.3 LIMITAÇÕES E DIFICULDADES ENCONTRADAS

Algumas limitações foram observadas no desenvolvimento das propostas desta dissertação. A primeira foi a limitação do simulador com o suporte à integração da ferramenta VEINS com o INET *framework*. Para que fosse possível a implementação da proposta de *streaming*

de vídeo nos veículos fazia-se necessário um esquema de endereçamento e comunicação de rede o que o *framework* VEINS não tem implementado. Visto isso, foi necessário configurar um modelo de rede baseado no INET para os veículos do VEINS, o que demandou muito tempo e pesquisa para o desenvolvimento.

Outra dificuldade que podemos elencar é o não suporte do Evalvid para construção de *traces* de vídeo reais para a plataforma OMNeT++. Nesse caso foi necessário criar um algoritmo que está disponível no apêndice para traduzir o *trace* de vídeo real gerado no Evalvid compatível com o OMNeT++.

Por fim, como estamos simulando vídeo real a avaliação deste em vários níveis e cenários demanda muito tempo, sendo observadas simulações de até três horas para serem concluídas. Além da grande quantidade de dados que o OMNeT++ gera, sendo necessário bastante tempo para a separação dos resultados de interesse.

6.4 TRABALHOS FUTUROS

Através das propostas elencadas por esta dissertação abaixo sugerimos algumas possíveis extensões à pesquisa desenvolvida neste trabalho.

- Propor um esquema de *handover* heterogêneo para a arquitetura proposta através de interfaces LTE.
- Melhorar o algoritmo de *handover* proposto para que as decisões e tabelas de regras armazenadas nos controladores locais seja distribuída aos controladores mais próximos em função da mobilidade do veículo.
- Melhorar o algoritmo de *handover* para considerar outros critérios de *handover*, como carga do controlador, latência do aplicativo e requisitos de largura de banda, etc.
- Desenvolver um modelo de adaptação automática da potência do sinal dos veículos observando sua posição geográfica em relação à comunicação com o controlador local.
- Arquitetar e implementar controladores SDN nos veículos, visando o gerenciamento em topologias unicamente V2V.
- Organizar e melhorar o algoritmo de agregação para que assim seja possível a implementação de zonas de segurança para que apenas os veículos interessados recebam os determinados dados.
- Verificar o impacto da implementação de um esquema de controle SDN em drones no *framework* desenvolvido, para verificar impacto do consumo de energia.

- Distribuir a carga de processamento das RSUCs entre os carros em uma abordagem FOG.

REFERÊNCIAS

- AKYILDIZ, I. F.; NIE, S.; LIN, S.-C.; CHANDRASEKARAN, M. 5G roadmap 10 key enabling technologies. *Computer Networks*, Elsevier, v. 106, p. 17–48, 2016.
- AL-TURJMAN, F. QOS—aware data delivery framework for safety-inspired multimedia in integrated vehicular-iot. *Computer Communications*, Elsevier, v. 121, p. 33–43, 2018.
- ALIOUA, A.; SENOUCI, S.-M.; MOUSSAOUI, S. DSDIVN: A distributed software-defined networking architecture for infrastructure-less vehicular networks. In: SPRINGER. *International Conference on Innovations for Community Services*. [S.l.], 2017. p. 56–67.
- ALIYU, A.; ABDULLAH, A. H.; KAIWARTYA, O.; CAO, Y.; LLORET, J.; ASLAM, N.; JODA, U. M. Towards video streaming in iot environments: Vehicular communication perspective. *Computer Communications*, Elsevier, v. 118, p. 93–119, 2018.
- ALOUACHE, L.; NGUYEN, N.; ALIOUAT, M.; CHELOUAH, R. Toward a hybrid SDN architecture for V2V communication in IOV environment. In: IEEE. *Software Defined Systems (SDS), 2018 Fifth International Conference on*. [S.l.], 2018. p. 93–99.
- ALOWISH, M.; TAKANO, Y.; SHIRAISHI, Y.; MORII, M. Performance evaluation of a cluster based routing protocol for vanets. *Journal of communications*, v. 12, n. 2, p. 137–144, 2017.
- APOSTOLOPOULOS, J. G.; TAN, W.-t.; WEE, S. J. Video streaming: Concepts, algorithms, and systems. *HP Laboratories, report HPL-2002-260*, 2002.
- ARANITI, G.; CAMPOLO, C.; CONDOLUCI, M.; IERA, A.; MOLINARO, A. LTE for vehicular networking: a survey. *IEEE communications magazine*, IEEE, v. 51, n. 5, p. 148–157, 2013.
- ARIF, M.; WANG, G.; BALAS, V. E. Secure vanets: trusted communication scheme between vehicles and infrastructure based on fog computing. *Stud. Inform. Control*, v. 27, n. 2, p. 235–246, 2018.
- ATWAL, K. S.; GULERIA, A.; BASSIOUNI, M. SDN-based mobility management and QOS support for vehicular ad-hoc networks. In: IEEE. *2018 International Conference on Computing, Networking and Communications (ICNC)*. [S.l.], 2018. p. 659–664.
- BANAVALIKAR, B. G. *Quality of service (QoS) for multi-tenant-aware overlay virtual networks*. [S.l.]: Google Patents, 2015. US Patent App. 14/229,692.
- CAVIN, D.; SASSON, Y.; SCHIPER, A. On the accuracy of manet simulators. In: ACM. *Proceedings of the second ACM international workshop on Principles of mobile computing*. [S.l.], 2002. p. 38–43.
- CHEN, J.; ZHOU, H.; ZHANG, N.; XU, W.; YU, Q.; GUI, L.; SHEN, X. Service-oriented dynamic connection management for software-defined internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 18, n. 10, p. 2826–2837, 2017.

-
- CISCO, V. N. I. Global mobile data traffic forecast update, 2015–2020 white paper. *Document ID*, v. 958959758, 2016.
- COSTA, L. R. Openflow e o paradigma de redes definidas por software. 2013.
- DONG, B.; WU, W.; YANG, Z.; LI, J. Software defined networking based on-demand routing protocol in vehicle ad hoc networks. In: IEEE. *Mobile Ad-Hoc and Sensor Networks (MSN), 2016 12th International Conference on*. [S.l.], 2016. p. 207–213.
- DUAN, X.; LIU, Y.; WANG, X. SDN enabled 5G-VANET: Adaptive vehicle clustering and beamformed transmission for aggregated traffic. *IEEE Communications Magazine*, IEEE, v. 55, n. 7, p. 120–127, 2017.
- EK, N. Ieee 802.1 p, q-qos on the mac level. *Apr*, v. 24, p. 0003–0006, 1999.
- FONTES, R. D. R.; CAMPOLO, C.; ROTHENBERG, C. E.; MOLINARO, A. From theory to experimental evaluation: Resource management in software-defined vehicular networks. *IEEE Access*, IEEE, v. 5, p. 3069–3076, 2017.
- FOUNDATION, O. N. Software-defined networking: The new norm for networks. 04 2012.
- GANESHKUMAR, P.; GOKULAKRISHNAN, P. Emergency situation prediction mechanism: a novel approach for intelligent transportation system using vehicular ad hoc networks. *The scientific world journal*, Hindawi, v. 2015, 2015.
- GHOSH, A.; PARANTHAMAN, V. V.; MAPP, G.; GEMIKONAKLI, O.; LOO, J. Enabling seamless V2I communications: toward developing cooperative automotive applications in VANET systems. *IEEE Communications Magazine*, IEEE, v. 53, n. 12, p. 80–86, 2015.
- GUEDES, D.; VIEIRA, L. F. M.; VIEIRA, M.; RODRIGUES, H.; NUNES, R. V. Redes definidas por software: uma abordagem sistêmica para o desenvolvimento de pesquisas em redes de computadores. *Minicursos do Simpósio Brasileiro de Redes de Computadores-SBRC*, v. 30, n. 4, p. 160–210, 2012.
- HE, Z.; ZHANG, D.; LIANG, J. Cost-efficient heterogeneous data transmission in software defined vehicular networks. In: IEEE. *High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conference on Embedded Software and Systems (ICESSE), 2015 IEEE 17th International Conference on*. [S.l.], 2015. p. 666–671.
- HUANG, C.-M.; CHIANG, M.-S.; DAO, D.-T.; PAI, H.-M.; XU, S.; ZHOU, H. Vehicle-to-infrastructure (V2I) offloading from cellular network to 802.11 p Wi-Fi network based on the software-defined network (SDN) architecture. *Vehicular Communications*, Elsevier, v. 9, p. 288–300, 2017.
- HUSSEIN, A.; ELHAJJ, I. H.; CHEHAB, A.; KAYSSI, A. SDN VANETs in 5G: An architecture for resilient security services. In: IEEE. *Software Defined Systems (SDS), 2017 Fourth International Conference on*. [S.l.], 2017. p. 67–74.
- JABALLAH, W. B.; CONTI, M.; LAL, C. A survey on software-defined vanets: Benefits, challenges, and future directions. *arXiv preprint arXiv:1904.04577*, 2019.

- KAIWARTYA, O.; ABDULLAH, A. H.; CAO, Y.; ALTAMEEM, A.; PRASAD, M.; LIN, C.-T.; LIU, X. Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects. *IEEE Access*, IEEE, v. 4, p. 5356–5373, 2016.
- KAUL, A.; XUE, L.; OBRACZKA, K.; SANTOS, M. A.; TURLETTI, T. Handover and load balancing for distributed network control: Applications in its message dissemination. In: IEEE. *2018 27th International Conference on Computer Communication and Networks (ICCCN)*. [S.l.], 2018. p. 1–8.
- KLAUE, J.; RATHKE, B.; WOLISZ, A. Evalvid: A framework for video transmission and quality evaluation. In: SPRINGER. *International conference on modelling techniques and tools for computer performance evaluation*. [S.l.], 2003. p. 255–272.
- KRANENBURG, R. V. *The Internet of Things: A critique of ambient technology and the all-seeing network of RFID*. [S.l.]: Institute of Network Cultures, 2008.
- KREUTZ, D.; RAMOS, F. M.; VERISSIMO, P. E.; ROTHENBERG, C. E.; AZODOLMOLKY, S.; UHLIG, S. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, IEEE, v. 103, n. 1, p. 14–76, 2015.
- KU, I.; LU, Y.; GERLA, M.; ONGARO, F.; GOMES, R. L.; CERQUEIRA, E. Towards software-defined VANET: Architecture and services. In: IEEE. *Ad Hoc Networking Workshop (MED-HOC-NET), 2014 13th Annual Mediterranean*. [S.l.], 2014. p. 103–110.
- LARA, A.; KOLASANI, A.; RAMAMURTHY, B. Network innovation using openflow: A survey. *IEEE communications surveys & tutorials*, IEEE, v. 16, n. 1, p. 493–512, 2014.
- LEE, E.-K.; OH, S. Y.; GERLA, M. RFID assisted vehicle positioning in VANETs. *Pervasive and Mobile Computing*, Elsevier, v. 8, n. 2, p. 167–179, 2012.
- LIU, K.; LI, C. An efficient adaptive frame aggregation scheme in vehicular ad hoc networks. In: IEEE. *Wireless Communications and Signal Processing (WCSP), 2017 9th International Conference on*. [S.l.], 2017. p. 1–6.
- LUO, Q.; LI, C.; YE, Q.; LUAN, T. H.; ZHU, L.; HAN, X. Cft: A cluster-based file transfer scheme for highway VANETs. In: IEEE. *Communications (ICC), 2017 IEEE International Conference on*. [S.l.], 2017. p. 1–6.
- MA, X.; KANELOPOULOS, G.; TRIVEDI, K. S. Application-level scheme to enhance vanet event-driven multi-hop safety-related services. In: IEEE. *2017 international conference on computing, networking and communications (ICNC)*. [S.l.], 2017. p. 860–864.
- MARTINEZ, F. J.; TOH, C.-K.; CANO, J.-C.; CALAFATE, C. T.; MANZONI, P. Realistic radio propagation models (RPMs) for VANET simulations. In: IEEE. *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*. [S.l.], 2009. p. 1–6.
- MAXWELL, J. *Dia a dia com Maxwell: Dicas e conselhos do maior especialista em liderança da atualidade*. 1a edição. ed. [S.l.]: Thomas Nelson Brasil, 2012. ISBN 978-8566997347.

- MCKEOWN, N.; ANDERSON, T.; BALAKRISHNAN, H.; PARULKAR, G.; PETERSON, L.; REXFORD, J.; SHENKER, S.; TURNER, J. Openflow: enabling innovation in campus networks. *ACM SIGCOMM Computer Communication Review*, ACM, v. 38, n. 2, p. 69–74, 2008.
- MUSSA, S. A. B.; MANAF, M.; GHAFOR, K. Z.; DOUKHA, Z. Simulation tools for vehicular ad hoc networks: A comparison study and future perspectives. In: IEEE. *Wireless Networks and Mobile Communications (WINCOM), 2015 International Conference on*. [S.l.], 2015. p. 1–8.
- PAPADIMITRATOS, P.; FORTELLE, A. D. L.; EVENSSEN, K.; BRIGNOLO, R.; COSENZA, S. Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation. *IEEE communications magazine*, IEEE, v. 47, n. 11, 2009.
- PATEL, A.; KAUSHIK, P. Improving QOS of VANET using adaptive cca range and transmission range both for intelligent transportation system. *Wireless Personal Communications*, Springer, v. 100, n. 3, p. 1063–1098, 2018.
- RADHAKRISHNAN, I.; SOUAY, R.; PALATTELLAZ, M. R.; ENGEL, T. An efficient service channel allocation scheme in sdn-enabled vanets. In: IEEE. *Ad Hoc Networking Workshop (Med-Hoc-Net), 2017 16th Annual Mediterranean*. [S.l.], 2017. p. 1–7.
- RAMOS, F. M.; KREUTZ, D.; VERISSIMO, P. Software-defined networks: On the road to the softwarization of networking. *Cutter IT journal*, 2015.
- REHMAN, S. ur; KHAN, M. A.; IMRAN, M.; ZIA, T. A.; IFTIKHAR, M. Enhancing quality-of-service conditions using a cross-layer paradigm for ad-hoc vehicular communication. *IEEE Access*, IEEE, v. 5, p. 12404–12416, 2017.
- SALAHUDDIN, M. A.; AL-FUQAHA, A.; GUIZANI, M. Software-defined networking for RSU clouds in support of the internet of vehicles. *IEEE Internet of Things journal*, IEEE, v. 2, n. 2, p. 133–144, 2015.
- SALAHUDDIN, M. A.; AL-FUQAHA, A.; GUIZANI, M.; CHERKAOUI, S. RSU cloud and its resource management in support of enhanced vehicular applications. In: IEEE. *Globecom Workshops (GC Wkshps), 2014*. [S.l.], 2014. p. 127–132.
- SALIHIN, S. S.; NISSIRAT, L. A.; NOOR, R. M.; AHMEDY, I. Handover schemes for vehicular ad-hoc networks over long term evolution advanced: A survey. In: IEEE. *2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA)*. [S.l.], 2018. p. 1–7.
- SECINTI, G.; CANBERK, B.; DUONG, T. Q.; SHU, L. Software defined architecture for vanet: a testbed implementation with wireless access management. *IEEE Communications Magazine*, IEEE, v. 55, n. 7, p. 135–141, 2017.
- SHEET, D. K.; KAIWARTYA, O.; ABDULLAH, A. H.; CAO, Y.; HASSAN, A. N.; KUMAR, S. Location information verification using transferable belief model for geographic routing in vehicular ad hoc networks. *IET Intelligent Transport Systems*, IET, v. 11, n. 2, p. 53–60, 2016.

- SKONDRAS, E.; MICHALAS, A.; SGORA, A.; VERGADOS, D. D. A vertical handover management scheme for VANET cloud computing systems. In: IEEE. *2017 IEEE Symposium on Computers and Communications (ISCC)*. [S.l.], 2017. p. 371–376.
- SLIPPERYBRICK. *Backseat Child Navigation Concept For Kids*. 2019. Disponível em: <<http://www.slipperybrick.com/search/Backseat+child+navigation+concept+for+kids>>.
- SMIDA, E. B.; FANTAR, S. G.; YOUSSEF, H. Predictive handoff mechanism for video streaming in a cloud-based urban VANET. In: IEEE. *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*. [S.l.], 2017. p. 1170–1177.
- SOUA, R.; KALOGÉITON, E.; MANZO, G.; DUARTE, J. M.; PALATTELLA, M. R.; MAIO, A. D.; BRAUN, T.; ENGEL, T.; VILLAS, L. A.; RIZZO, G. A. Sdn coordination for ccn and fc content dissemination in vanets. In: *Ad Hoc Networks*. [S.l.]: Springer, 2017. p. 221–233.
- SUDHEERA, K. K.; MA, M.; ALI, G. M. N.; CHONG, P. H. J. Delay efficient software defined networking based architecture for vehicular networks. In: IEEE. *Communication Systems (ICCS), 2016 IEEE International Conference on*. [S.l.], 2016. p. 1–6.
- TANENBAUM, A. S.; WETHERALL, D. *Computer networks*. [S.l.]: Prentice hall, 1996.
- TOMOVIC, S.; YOSHIGOE, K.; MALJEVIC, I.; RADUSINOVIC, I. Software-defined fog network architecture for iot. *Wireless Personal Communications*, Springer, v. 92, n. 1, p. 181–196, 2017.
- TRUONG, N. B.; LEE, G. M.; GHAMRI-DOUDANE, Y. Software defined networking-based vehicular adhoc network with fog computing. In: IEEE. *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*. [S.l.], 2015. p. 1202–1207.
- UFRN-GTA. *IEEE 802.1p - Qos na camada MAC*. 2002. Disponível em: <https://www.gta.ufrj.br/grad/02_2/802.1p/>.
- VEGNI, A. M.; BIAGI, M.; CUSANI, R. Smart vehicles, technologies and main applications in vehicular ad hoc networks. In: *Vehicular technologies-deployment and applications*. [S.l.]: IntechOpen, 2013.
- VENKATRAMANA, D. K. N.; SRIKANTAIHAH, S. B.; MOODABIDRI, J. SCGRP: SDN-enabled connectivity-aware geographical routing protocol of vanets for urban environment. *IET Networks*, IET, v. 6, n. 5, p. 102–111, 2017.
- WANG, X.; WANG, C.; ZHANG, J.; ZHOU, M.; JIANG, C. Improved rule installation for real-time query service in software-defined internet of vehicles. *IEEE Transactions on Intelligent Transportation Systems*, IEEE, v. 18, n. 2, p. 225–235, 2017.
- WU, H.; RAO, K. Perceptual video quality metrics—a review. In: *Digital video image quality and perceptual coding*. [S.l.]: CRC press, 2005. p. 191–216.
- XU, L. D.; HE, W.; LI, S. Internet of things in industries: A survey. *IEEE Transactions on industrial informatics*, IEEE, v. 10, n. 4, p. 2233–2243, 2014.

ZHANG, Y.; RUI, L.; HUI, G.; QIU, X. et al. A QOS guarantee mechanism based on multi-priority bionic competition model in vehicular edge etwork. In: IEEE. *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*. [S.l.], 2018. p. 1–5.

ZHU, W.; GAO, D.; ZHAO, W.; ZHANG, H.; CHIANG, H.-P. SDN-enabled hybrid emergency message transmission architecture in internet-of-vehicles. *Enterprise Information Systems*, Taylor & Francis, v. 12, n. 4, p. 471–491, 2018.

APÊNDICE A – CÓDIGOS SHELL SCRIPT

A.1 CÓDIGOS FONTE

Listing A.1 – Código em linguagem de programação Shell Script para conversão de traces de vídeo do Evalvid para o formato aceito na plataforma Omnet++

```

1  #!/bin/bash
3
5  #-----#
5  # Shell Script to convert trace Evalvid to Omnet++ accepted trace      #
7  #-----#

9  ## =====
9  ## Author: Alexsanderson Vieira Santos
11 ## Email: avs@cin.ufpe.br, thibleu@gmail.com
11 ## =====

13 #name of modified archive

15 newArchive="$3_omnetpp.mpg.gdf"

17 if [ $# -lt 3 ];then
19     echo "Wrong parameters:"
19     echo "Usage: ./transformOmnet <Frame-Rate> <Initial-Delay> <Trace evalvid>"
21     exit 1
23 fi

23 if [ -e $3 ];then
25
27     #set frame rate

29     echo "$1 [frames/second] Frame Rate" >> $newArchive

31     #set initial delay

33     echo "$2 [seconds] Initial Delay" >> $newArchive

35     #transform trace
35     sed -e 's/I/I-Frame/g' -e 's/P/P-Frame/g' -e 's/H/H-Frame/g' $3 | awk '{print (
37         $3*8)" [bits] "$2}' >> $newArchive

39     echo "Done!"
39     echo "New archive generated: $newArchive"

41 else

43     echo "Trace $3 doesn't exist!"
43     exit 1
45

```

```
fi
```

Listing A.2 – Código em linguagem de programação Shell Script para calculo automatizado dos valores de PSNR dos traces de vídeo simulados

```

2  #!/bin/bash

4  ## =====
   ## Author: Alexsanderson Vieira Santos
6  ## Email: avs@cin.ufpe.br, thibleu@gmail.com
   ## =====
8

10 if [ "$1" = "-r" ];then
    rm -rf *_remontado.*
12    rm -rf *_dump
    rm -rf ref_*
14    echo " "
    echo "REMOVED!"
16    echo " "

18 elif [ "$1" = "-h" ];then

20    echo " "
        echo "USAGE: sudo ./calcPSNR <RECEIVER.pcap> <SENDER.pcap>"
22    echo "CLEAR PSNR FILES: sudo ./calcPSNR -r"
    echo "CLEAR ALL FILES: sudo ./calcPSNR -rp"
24    echo " "

26 elif [ "$1" = "-rp" ];then
    rm -rf *_remontado.*
28    rm -rf *_dump
    rm -rf ref_*
30    rm -rf *.pcap
    echo " "
32    echo "REMOVED ALL!"
    echo " "
34

   elif [ $# -eq 2 ];then
36
        tcpdump -ttvnr $1 > receiver_dump
38
        tcpdump -ttvnr $2 > sender_dump
40
        ./exec/etmp4 -f -0 sender_dump receiver_dump results/st_akiyo akiyo_cif.mp4
            akiyo_remontado.mp4 200
42
        ffmpeg -i akiyo_remontado.mp4 -pix_fmt yuv420p ref_conv_akiyo_cif.yuv
44
        ./exec/psnr 352 288 420 akiyo_cif.yuv ref_conv_akiyo_cif.yuv
46

48    echo " "
    echo "DONE!"
50    echo " "

```

```
52 elif [ $# -ge 3 ];then
54     echo " "
55     echo "Wrong parameters or excessive number of them!"
56     echo "Use -h to help descriptions!"
57     echo " "
58
59     else
60
61     echo " "
62     echo "Wrong parameters:"
63     echo "Use -h to help descriptions!"
64     echo " "
65     exit 1
66
67 fi
```