

Universidade de São Paulo
Instituto de Matemática e Estatística
Bacharelado em Ciência da Computação

Patrick Abrahão Menani

**Transmissão de vídeo em VANETs
Utilizando a técnica DASH**

São Paulo
Novembro de 2018

Transmissão de vídeo em VANETs Utilizando a técnica DASH

Monografia final da disciplina
MAC0499 – Trabalho de Formatura Supervisionado.

Orientador: Prof. Dr. Alfredo Goldman
Co-orientador: Dr. Roger Immich

São Paulo
Novembro de 2018

Resumo

O emprego de carros equipados com equipamentos de vídeo, assim como carros conectados e autônomos, vem ganhando força. Os números de serviços e aplicações a disposição em 'Vehicular Ad-hoc Networks (VANETs)' estão seguindo a mesma tendência, além do novo conceito de comunicação 'Vehicle-to-everything (V2X)'. Esse tipo de rede é contemplada com um componente central de 'Intelligent Transportation Systems (ITS)', que provê um suporte para uma grande variedade de aplicações, incluindo serviço de vídeo. Esses serviços permeiam a rede com conteúdo de vídeo diariamente. Como resultado deste aumento abrupto do tráfego de vídeo, a propensão de erros devido a interferências na rede e por congestionamento cresce. Levando isso em conta, a adoção de novas tecnologias/paradigmas como '5G' e 'Fog/Edge computing' é aconselhada. Os sistemas de rede sem fio 5ª geração (5G) é projetada para suportar um número massivo de comunicações em uma rede de alta densidade. Adicionalmente, 'Fog' ou 'edge computing' é um método utilizado para tornar os serviços de 'Cloud' mais próximos da extremidade das redes. Isso permite prover aplicações com conhecimento do contexto da rede assim como comunicações com uma latência menor. Entretanto, a mera adoção dessas tecnologias apenas mitiga os problemas durante a transmissão do vídeo, mas melhorias são necessárias para prover ao usuário final uma melhor 'Quality of Experience (QoE)'.

Com um estudo da literatura atual de 'Fog', VANETs, de transmissão de vídeo e de simuladores próprios para a implementação destes ambientes, este trabalho visa a implementação de um método já existente de transmissão e baseado nessa implementação, serão feitas análises estatísticas para chegar a uma conclusão em relação ao QoE e QoS deste método para o usuário final.

Palavras-chave: VANETs, DASH, transmissão, vídeo, 5G, Fog, QoE, QoS.

Abstract

The use of cars equipped with video equipment, as well as connected and autonomous cars, has been gaining strength. The disponibility of services and applications in 'Vehicular Ad-Hoc Networks' (VANETs) has been following the same trend, besides of the new concept of communication 'Vehicle-to-everything' (V2X). Thus new kind of network is contemplated with a central component, the 'Intelligent Transportation Systems (ITS)', that provides support for a large variety of applications, including video service. These services permeates the network with video content daily. As a result of this abrupt raise in the video traffic, the propensity of errors due to network interference and congestion raises as well. Taking this into account, the adoption of new technologies/paradigms like '5G' and 'Fog/Edge computing' is advised. The new 5th Generation (5G) is pojected to support a massive number of communications in a high density network. Additionally, 'Fog' and 'Edge Computing' is a method to put the 'Cloud' services near to the network's extremities. It allows to provide applications with networks context awareness and lower latency. However, the simple adoption of this technologies just mitigates the problems of video transmission, but improvements are required to provide a better 'Quality of Experience (QoE)'.

Studying the current literature of 'Fog', VANETs, video transmission and proper simulators for implementations of this kind of enviroment, this work aims to implement an already existing method of video transmission and based on this implementation, statistical analyzes will be carried out to arrive at a conclusion in relation to QoE and QoS of this method to the application end user in this context.

Keywords: VANETs, DASH, transmission, video, 5G, Fog, QoE, QoS.

Conteúdo

I	Parte Objetiva	1
1	Introdução	3
2	Background	5
2.1	Reprodução de Vídeos	5
2.2	Técnica de transmissão DASH	5
2.2.1	Preparação do vídeo	6
2.2.2	Avaliação da rede	7
2.2.3	Transmissão	7
2.3	VANETs	7
2.3.1	Arquiteturas de VANETs	7
2.3.2	Desafios Atuais	8
2.4	Simuladores	9
2.4.1	OMNeT++	9
2.4.2	INET	11
2.4.3	SimuLTE	12
2.4.4	SUMO e Veins	12
2.5	QoS e QoE	13
2.5.1	QoS	13
2.5.2	QoE	13
3	Trabalhos Relacionados	15
3.1	VANETs	15
3.2	Transmissão de vídeo	15
3.3	Transmissão sem fio	15
3.4	Simulação de cenários de rede	16
4	Análise dos cenários com o DASH	17
4.1	Cenários Experimentais	17
4.2	Análise dos Resultados	19
4.2.1	Avaliação do Cenário 1	20
4.2.2	Avaliação do Cenário 2	24

5 Conclusões	29
Bibliografia	31
II Parte Subjetiva	33
Desafios e Frustrações	35
A contribuição do curso	37
Próximos Passos	39
Agradecimentos	41

Parte I
Parte Objetiva

Capítulo 1

Introdução

Nos últimos anos houve um grande aumento na utilização de aplicações de vídeo. Estas aplicações estão presentes em redes sociais, serviços de vídeo sob demanda e até em jogos interativos. Para aumentar ainda mais a demanda, cada vez mais são produzidos mais e melhores dispositivos móveis ([Adobe Digital Index \(2017\)](#)), colocando um grande poder de processamento nas mãos dos usuários. Isto permite o fácil acesso à produção, consumo e disponibilização de conteúdos multimídia. Todas as facilidades levam a um considerável aumento no tráfego de rede. Somente para citar um exemplo, de acordo com a Cisco, até 2021 o tráfego global de toda a Internet deverá ser 127 vezes maior do que foi em 2005. De todo este tráfego, é esperado que a transmissão de vídeo represente mais de 82% do total ([Cisco \(2017\)](#)).

Essas novas aplicações tem ganhado espaço em diversos novos cenários do nosso cotidiano, como por exemplo os veículos automotores que desde sua concepção e invenção nos séculos 18 e 19, sempre foram vistos como um facilitador na locomoção humana. Porém, com o avanço tecnológico recente em dispositivos portáteis, foi possível a adição de componentes com uma capacidade de processamento e aquisição de informação nos veículos que vem mudando essa percepção que temos deles. Esse ambiente conhecido como redes veiculares *AD-HOC* (*VANETs*) é composto por uma estrutura capaz de fornecer dados para veículos que trafegam em uma determinada área, e outra estrutura que configura o conjunto de veículos que utilizarão o serviço. Redes veiculares tem apresentado atualmente vantagens como controle eficiente de tráfego, maior segurança e possibilidade de desenvolvimento de novos recursos ([Shanhe Yi \(2015\)](#)).

As *VANETs* utilizam os conceitos de *Computação em Nuvem e em Borda/Névoa* ([Tesnim Mekki \(2017\)](#), [SALIM BITAM e ZEADALLY \(2015\)](#), [Shanhe Yi \(2015\)](#)). O crescente uso de computação em nuvem trouxe novos paradigmas para a aplicação desta tecnologia nos cenários em desenvolvimento atualmente, como: latência não confiável, falta de mobilidade e pontos de localização para disponibilização da rede. Portanto, novas tecnologias vem sendo estudadas para solucionar esses novos problemas, como a computação em névoa e de borda. A computação em névoa aparece nesse cenário provendo uma maior maleabilidade próxima aos usuários finais das aplicações, podendo prover uma melhor latência e confiabilidade das condições da rede local e pode ser utilizada em conjunto com a computação em nuvem, que seria responsável pela rede central ([Shanhe Yi \(2015\)](#)). Porém, alguns trabalhos ainda citam a necessidade que temos de desenvolver e aprimorar ainda mais alguns aspectos destas técnicas para elas se tornarem realmente úteis em cenários reais ([Rong Yu e Stein Gjessing \(2013\)](#); [SALIM BITAM e ZEADALLY \(2015\)](#); [Shanhe Yi \(2015\)](#); [Tesnim Mekki \(2017\)](#)).

Levando-se em consideração o grande crescimento na demanda pela transmissão de vídeo, o desenvolvimento e aprimoramento das redes veiculares, as novas técnicas de transmissão de longo alcance e os simuladores existentes de redes, este trabalho visa a implementação de um cenário simulado de redes veiculares utilizando comunicação 5G, com o intuito de estudar o comportamento de aplicações de vídeo utilizando uma técnica específica de transmissão. Baseado neste estudo, este trabalho pretende avaliar a viabilidade da implementação desta técnica em ambientes reais de *VANETs*.

Capítulo 2

Background

2.1 Reprodução de Vídeos

A tecnologia de vídeo é utilizada para gravação de vídeos para reprodução em aparelhos digitais e é composto por um fluxo de imagens em um determinado formato que traduz a sequência das imagens (Frames) em um fluxo fluído para a reprodução. Alguns exemplos desses formatos são MPEG-4, HVEC e Theora e a implementação desses formatos e chamada de codecs como OpenH264, Xvid e x264 (Immich (2017)).

Na tecnologia de vídeo a compressão de vídeo é um dos elementos mais importantes, pois ela facilita a transmissão do vídeo reduzindo o tamanho dos vídeos, podendo aproveitar melhor a banda de transmissão das aplicações. A tecnologia de compressão funciona eliminando partes redundantes dos vídeos que podem ser facilmente reconstruídas do lado da reprodução do vídeo (Immich (2017)).

Dentro das reproduções de vídeo uma outra parte importante são os elementos de codificação do vídeo. Eles definem as relações entre os grupos de imagens (Group of Pictures, GoP) de um vídeo comprimido definindo assim a inter relação de reprodução das imagens do vídeo. O método mais utilizado atualmente é o H.264 (Immich (2017)), ele consiste em dividir os frames do vídeo em GoPs e cada frame do GoP é de um tipo específico que possui uma característica específica para o funcionamento da codificação. Esses tipos de frames são os Intra-coded Frame (I-Frame), Predictive-coded Frame (P-Frame) e o Bi-directionally Frame (B-Frame). O frame I é responsável por carregar um frame só e pode ser reproduzido sem a necessidade de consulta e análise dos outros frames, ele é a representação de uma imagem estática do vídeo. O frame P é responsável pela transição dos frames I e P anteriores durante o vídeo e necessita da informação do frame anterior para realizar a transição. O frame B realiza a mesma transição do frame P porém, necessita da informação dos dois frames, tanto o anterior quanto o subsequente a ele. Os frames P e B são frames de movimento compostos por vetores de transição entre as imagens do vídeo.

Com essa implementação o codec pode realizar a reprodução de um vídeo de forma fluída sem que o usuário perceba que são imagens em alinhamento sendo reproduzidas. Porém, ao se perder um desses frames, dependendo de qual deles é perdido, o vídeo pode ter travamentos ou descontinuações que atrapalham a reprodução fluída do vídeo.

2.2 Técnica de transmissão DASH

A crescente demanda por mídias digitais e o avanço tecnológico em serviços de internet e aparelhos móveis gerou uma necessidade da adaptação do modelo tradicional de oferecimento

dos conteúdos de mídias digitais sensíveis, como os vídeos. Comumente era função do usuário a adaptação da qualidade destes serviços de acordo com a percepção dele em relação a sua qualidade de banda e sinal da rede. Com o foco na melhoria da qualidade de experiência do usuário ao utilizar o serviço de transmissão de vídeos, foi desenvolvida a técnica de transmissão adaptativa em HTTP (ou Dynamic Adaptive Streaming over HTTP, DASH), atualmente utilizada por diversos provedores de serviço de vídeo online como YouTube, Netflix e Facebook.

A técnica do DASH consiste em avaliar as condições atuais da rede em que o usuário do serviço de transmissão está inserido e procura adaptar dinamicamente a qualidade do vídeo de acordo com a qualidade da rede, como ilustra a figura 2.1. Os programas que implementam a técnica, realizam algumas preparações do vídeo antes de realizar a transmissão para o usuário, como a preparação dos diversos segmentos de vídeos em diversas qualidades, para que possa ocorrer a troca de qualidade sem que o usuário tenha uma experiência ruim durante a troca, e a preparação de um manifesto que determina como são esses segmentos.

Essa seção tem como o objetivo elucidar aspectos importantes do funcionamento e implementação da técnica em sua forma mais comum, sem entrar em detalhes das implementações específicas de cada provedor de serviço de vídeo.

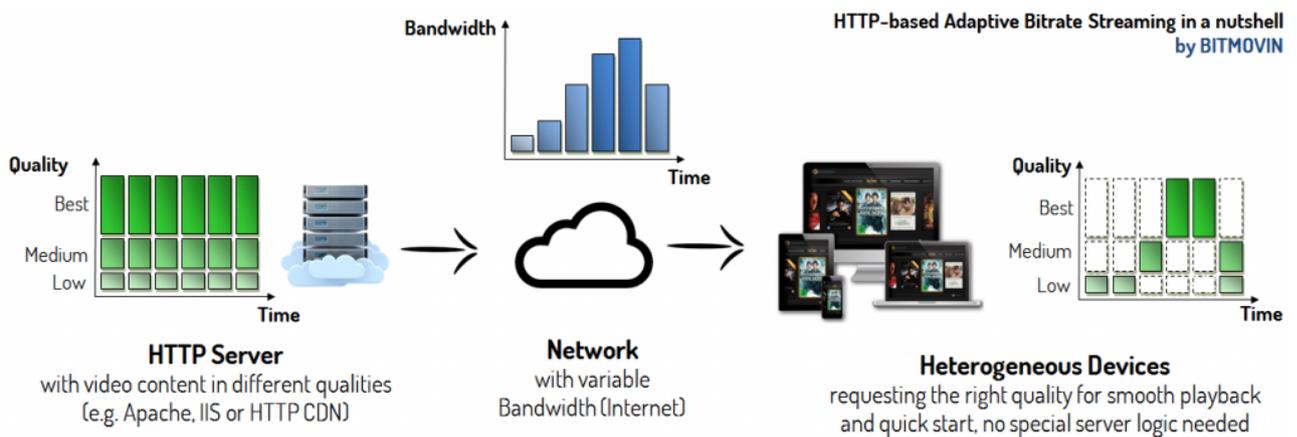


Figura 2.1: Exemplo do funcionamento da técnica DASH. *MPEG-DASH (2018)*

2.2.1 Preparação do vídeo

Entre muitos aspectos importantes do DASH, um dos iniciais é a preparação do vídeo em suas diversas qualidades, estas definidas por cada vídeo e serviço de transmissão. Nesta etapa o vídeo a ser transmitido pelo serviço é preparado tendo novos vídeos gerados em algumas qualidades variando de qualidades mais baixas até a qualidade original do vídeo.

Tendo os novos vídeos gerados, sendo o de qualidade alta a qualidade original do vídeo como base, é criado um ou mais manifestos de descrição de mídia (Media Presentation Description, MPD). Este manifesto descreve a segmentação de qualidade do vídeo baseado em um segmento mínimo que pode variar de um a quinze segundos do vídeo, baseado nesse segmento mínimo os vídeos gerados tem sua segmentação alinhada para que a reprodução possa variar de qualidade a cada intervalo de tempo da transmissão, correspondente ao segmento escolhido para a geração do MPD.

O MPD é um arquivo de dados hierárquico que pode conter um dos períodos do vídeo ou todos os períodos agregados, no caso de diversos MPDs eles possuem a indicação hierárquica de execução do vídeo para cada manifesto anterior e subsequente. Cada período pode conter

componentes de mídia referentes ao vídeo sendo transmitido, como diferentes qualidades do vídeo, diferentes linguagens, diferentes qualidades de áudio ou diferentes ângulos de visão. Cada período pode ser sub-dividido em segmentos que possuem representações diferentes do vídeo, possibilitando o usuário alterar características do vídeo, como legenda, áudio, etc., de acordo com sua preferência.

Ao final da preparação dos vídeos, os manifestos possuem as informações dos arquivos dos vídeos segmentados para a variação de qualidade e das características do vídeo.

2.2.2 Avaliação da rede

O DASH avalia a rede de forma a escolher a melhor qualidade disponível para aquela transmissão em um determinado tempo de reprodução do vídeo. Em geral, a avaliação que a técnica realiza em relação a qualidade da rede consiste em avaliar a banda disponível para a transmissão e assim tomar uma decisão de qual é o melhor segmento de vídeo a ser enviado. Existem algumas outras aplicações da técnica que consideram outros fatores, como o tipo de dispositivo, Desktop ou Mobile, o tipo de conexão, Wi-Fi ou LTE, porém esse trabalho foca na versão original do DASH sobre a avaliação da rede.

2.2.3 Transmissão

A transmissão do DASH é feita a partir de requisições HTTP e a cada requisição o algoritmo toma a decisão de qual será a próxima qualidade, baseada na qualidade da rede avaliada pelo envio e recebimento do segmento anterior. Portanto, a transmissão é feita a partir de um segmento inicial e subsequentemente muda suas requisições quando detecta uma mudança na qualidade da rede.

2.3 VANETs

A computação em nuvem (Cloud Computing, CC) se tornou indispensável para a implementação de diversas aplicações de grande escala ao redor do mundo nos últimos 20 anos. Porém, com o crescente avanço da utilização de serviços móveis e limitações das tecnologias atuais de transmissão sem fio, novos paradigmas surgiram para atenuar problemas causados pelo uso da computação em nuvem.

Dentre os novos paradigmas as redes veiculares ad-hoc (vehicular ad-hoc networks, VANETs) surgiram como uma arquitetura para lidar com a disponibilidade de recursos computacionais para os computadores embarcados em veículos. A CC tradicional sofre com a disponibilidade de banda e com a latência em relação aos veículos, por serem grandes centros estáticos que não possuem a capacidade de acompanhar o dinamismo do movimento do tráfego veicular. Como opção para isso as VANETs propõem uma arquitetura mais dinâmica que tenta atenuar os efeitos do uso de CC tradicional em redes veiculares.

2.3.1 Arquiteturas de VANETs

Conforme [Shanhe Yi \(2015\)](#), as redes veiculares se encaixam hoje no que chamamos de redes de computação de névoa (FCN) (do inglês "fog computing networks"). Essas redes são divididas em três partes: os servidores de larga escala, como as nuvens de serviços que se situam em diversas partes do globo, os pequenos servidores, que se encontram próximos aos usuários das redes (Road Side Unit, RSU), e os dispositivos de borda, que são dispositivos

que se conectam às redes e possuem capacidade de processamento e memória, estes elementos estão representados na figura 2.2.

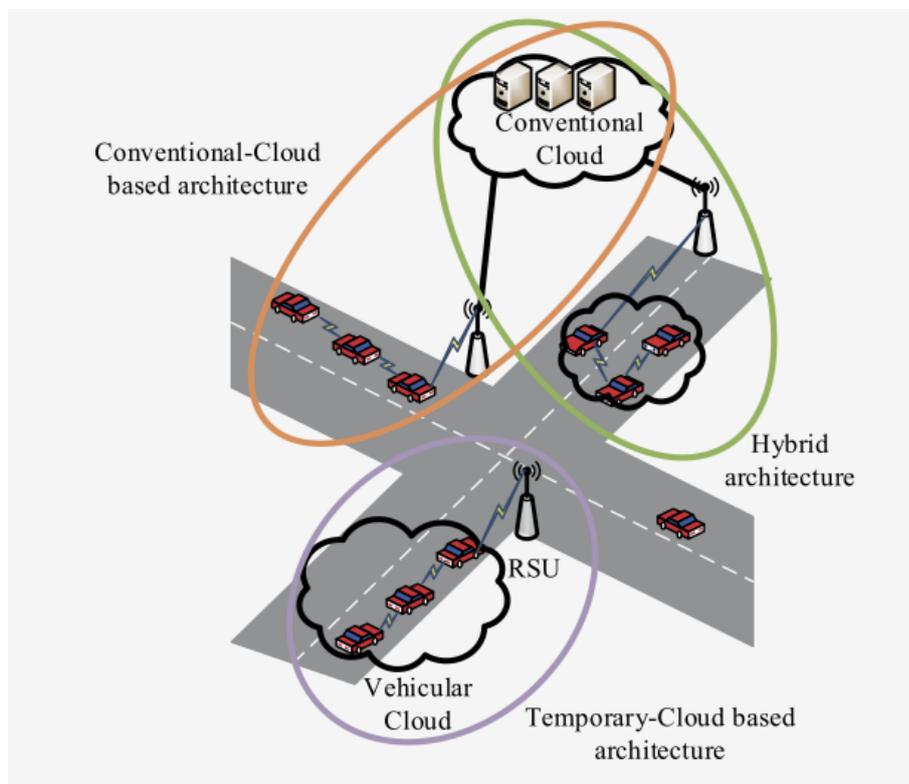


Figura 2.2: Exemplo das possíveis arquiteturas de comunicação das VANETs. *Tesnim Mekki (2017)*

Dentro desse cenário podemos classificar as conexões entre os três elementos centrais da arquitetura das VANETs: veículo com veículo (Vehicle to Vehicle, V2V) que se encaixa em toda conexão que dois dispositivos de borda, os veículos, fazem entre eles para realizar troca de dados, veículo com a nuvem (Vehicle to Cloud, V2C) que descreve toda a conexão direta entre os elementos de borda com a nuvem central de processamento, veículo para borda (Vehicle to RSU) sendo a conexão entre os elementos de borda com os elementos de processamento de borda e por fim veículo para tudo (Vehicle to Everything, V2X) que descreve quando os elementos de borda possuem uma conexão híbrida contemplando todas as anteriores (Noori (2012); Rong Yu e Stein Gjessing (2013); Shanhe Yi (2015); Tesnim Mekki (2017)). A figura 2.2 ilustra as quatro formas de conexão descritas acima.

2.3.2 Desafios Atuais

As FCN, segundo Shanhe Yi (2015), contém dispositivos heterogêneos que podem diferir em capacidade de processamento, memória, banda de internet, basicamente em tudo que compõe o dispositivo capaz de se conectar a rede, o que torna a predição pela rede acerca dos dispositivos um trabalho muito difícil. Portanto, tais redes devem possuir mecanismos de identificação de características para que cada dispositivo determine suas próprias características, para que então a rede possa definir a melhor abordagem para cada dispositivo conectado.

Outro desafio que estas redes apresentam é no modo como seriam as linguagens e modelos de programação adequados e mais aceitáveis para a produção portátil de aplicações entre redes diferentes, o que ainda vem sendo amplamente estudado na área de pesquisa de FCN, de acordo com Shanhe Yi (2015). Um ponto que também é levantado como um desafio é a forma

de cobrança e monitoramento; não pode haver um modelo sustentável para uma FCN sem ser em conjunto de um modelo comercial. Um aspecto importante, que é considerado um grande desafio, é a alocação e o gerenciamento de recursos desta rede e a disposição destes recursos aos usuários da rede, como e quais servidores prestariam serviços a um determinado usuário e como garantir o uso adequado deste serviço. As redes deste gênero também apresentam desafios na área de segurança e privacidade, pois é necessário a garantia de autenticação de usuário, controle de acesso, detecção de invasões e a garantia da privacidade dos usuários da rede.

Em relação ao tema deste trabalho, a transmissão de vídeos em uma rede móvel, um dos desafios é a sensibilidade deste tipo de conteúdo e a alocação e gerenciamento de recursos. O maior desafio nesta parte é manter o veículo, que neste cenário é encarado como um dispositivo de borda, conectado aos pequenos servidores locais e prever sobre a movimentação deste veículo para que a sua máquina virtual (MV), representação virtual do que é o veículo na rede dos servidores locais e quais serviços ele usa, que possui os dados de sua transmissão de vídeo possua sempre o menor atraso possível em relação a ele.

2.4 Simuladores

Devido a grande demanda de prevermos o funcionamento de aplicações em cenários reais de rede e a dificuldade de gerarmos cenários reais de teste, os simuladores de rede são de grande importância para conseguirmos validar o comportamento dessas aplicações em cenários reais. Essa seção tem como objetivo introduzir os simuladores utilizados neste trabalho e elucidar seu funcionamento.

2.4.1 OMNeT++

O OMNeT++ (OMNeT++ (2018)) é um framework de simulação discreto de redes, de código aberto, implementado em C++ e que possui diversos elementos de rede previamente configurados, seguindo os mais diversos padrões, que podem ser utilizados em simulações de rede. Ele possui uma estrutura de configuração onde podem ser configurados cenários, de forma visual ou em formato de texto, em seus arquivos .NED (figura 2.3), os quais descrevem a topologia geral de rede, e o arquivo de configuração .ini (figura 2.4) que descreve os elementos de configuração dos módulos da rede, como largura de banda, ruído na rede, aplicações disponíveis, etc. O simulador possui uma IDE implementada utilizando a IDE Eclipse para facilitar a utilização e compatibilidade com diversos sistemas operacionais, como Linux, Windows e MAC.

O framework possui uma interface de execução visual onde o usuário pode visualizar as simulações e acompanhar as simulações de algumas formas como, passo a passo e identificar o funcionamento correto da rede, de forma rápida e analisar o comportamento geral das interações dos módulos da rede, ou de forma expressa e analisar apenas o resultado final das simulações. Além da interface visual, o usuário pode optar pela execução em linha de comando, onde as simulações podem ocorrer de forma mais rápida, devida a abstração visual que toma muito poder computacional. Nesta interface por linha de comando o usuário pode executar lotes de simulações com o intuito de extrair informações gerais do comportamento geral da simulação em diversas situações.

Outro ponto do funcionamento do simulador são os arquivos de saída gerados pelo monitoramento dos módulos da rede simulada. Os arquivos de saída podem ser configurados nos módulos para que a saída possua os vetores de monitoramento de diversos aspectos dos módulos da rede como delay de pacotes, perda de pacotes, requisições por período de

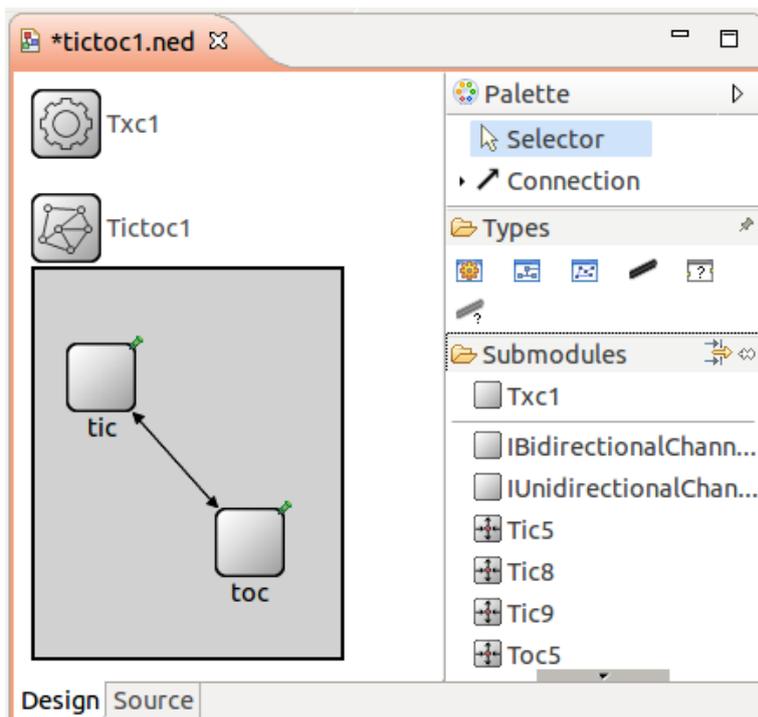


Figura 2.3: Exemplo do editor NED de configuração de rede. OMNeT++ (2018)

tempo, etc. Esses arquivos de monitoramento são de grande utilidade para o processamento estatístico dos dados de saída das simulações de rede, permitindo ao usuário uma análise personalizada e completa dos dados de rede simulados. Informações mais detalhadas do funcionamento dos módulos de monitoramento e de execução do simulador podem ser encontrados na documentação da página oficial, que possui uma excelente cobertura quanto ao seu uso e funcionamento.

O OMNeT++ vem sendo utilizado em ambientes de simulações em variados artigos acadêmicos que tocam o âmbito das simulações de características de rede como Denis Rosário (2013); Jair Jose Ferronato (2015), por sua facilidade de uso e por seus frameworks de extensão, como INET e Veins, que ampliam ainda mais a capacidade de simulação de redes.

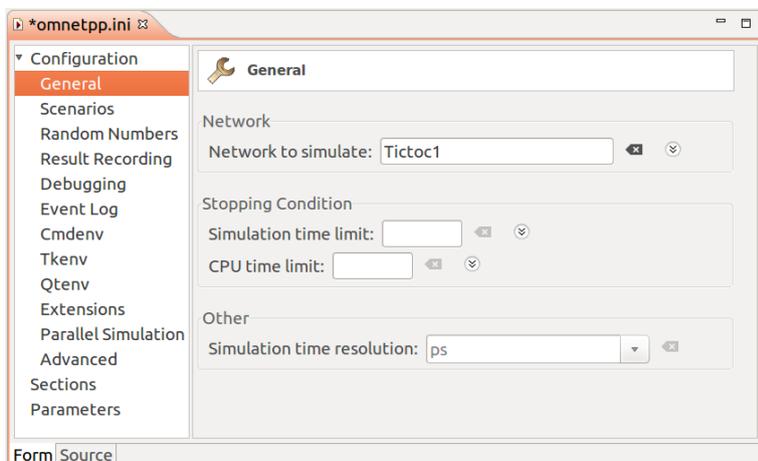


Figura 2.4: Exemplo do editor INI de configuração de rede. OMNeT++ (2018)

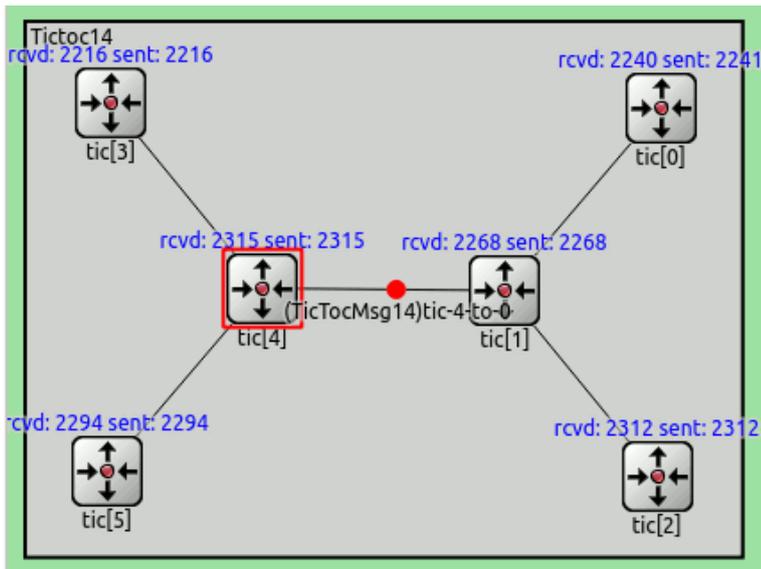


Figura 2.5: Exemplo de uma simulação visual e seus elementos de monitoramento, neste caso os pacotes recebidos (*rcvd*) e enviados (*sent*). *OMNeT++* (2018)

2.4.2 INET

O INET (INET (2018)) é uma biblioteca de código aberto de modelos para o framework OMNeT++. Ela surge como uma forma de estender as funcionalidades de rede que o OMNeT++ possui, como protocolos de rede, como TCP, UDP, IPv4, IPv6, OSPF, BGP, etc., protocolos de conexão tanto cabeados como wireless, como Ethernet, PPP, IEEE 802.11, etc., suporte para mobilidade dos nós, protocolos MANET, diversos modelos de aplicações, como HTTP e VoIP, muitos outros protocolos e componentes adicionais que o INET incrementa ao framework original.

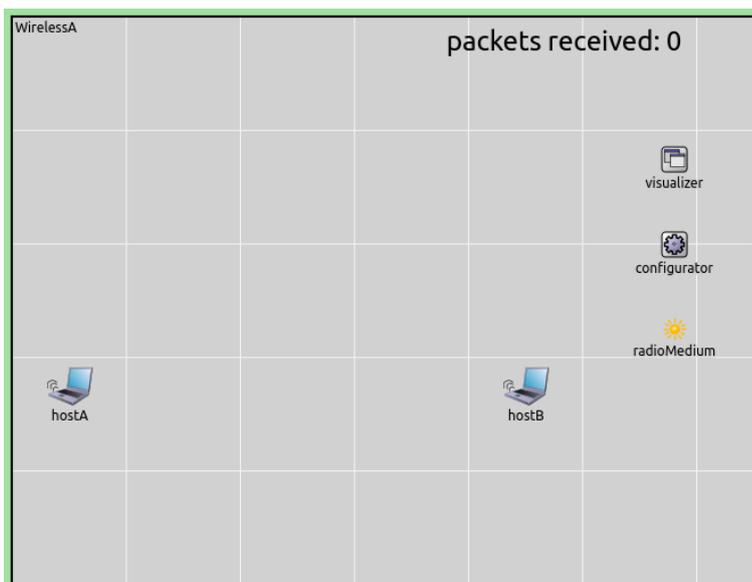


Figura 2.6: Exemplo dos nós adicionais do INET que possuem comunicação wireless. *INET* (2018)

O INET utiliza a mesma hierarquia de funcionamento do OMNeT++ e realiza as simulações da mesma forma.

2.4.3 SimuLTE

O SimuLTE [SimuLTE \(2018\)](#) é uma ferramenta que estende as funcionalidade dos frameworks INET e OMNeT++, possibilitando a avaliação de simulações complexas de redes LTE e redes avançadas de LTE, como 3GPP. Ele permite o uso de antenas e receptores adaptados ao cenário LTE para a execução de simulações de redes que se beneficiam desses equipamentos como MANETs, VANETs, etc.

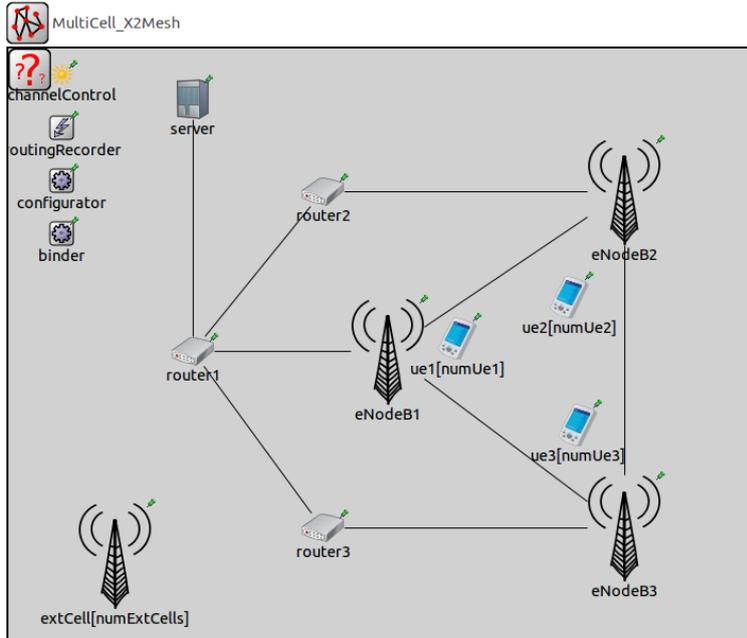


Figura 2.7: Exemplo de uma rede LTE implementada no SimuLTE. [SimuLTE \(2018\)](#)

2.4.4 SUMO e Veins

O framework Veins ([Veins \(2018\)](#)) é um simulador código aberto de redes urbanas com mobilidade. Da mesma forma que o INET e o SimuLTE ele expande o OMNeT++ adicionando ferramentas e módulos para a simulação de redes urbanas, simulando a movimentação e conectividade dos nós de forma mais realista o possível sem prejudicar a performance das simulações. Ele faz uso do SUMO ([Sumo \(2018\)](#)), um simulador de mobilidade urbana, para a geração de arquivos de mobilidade urbana para simulações de rede com integração ao OMNeT++. O Veins pode ser integrado ao INET e ao SimuLTE para variações nas simulações.

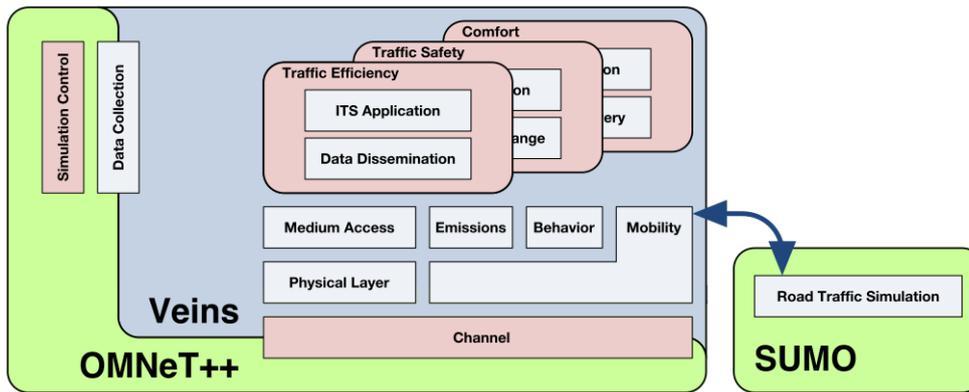


Figura 2.8: Arquitetura de funcionamento das integrações entre o Veins, SUMO e OMNeT++.
Veins (2018)

2.5 QoS e QoE

2.5.1 QoS

A qualidade de serviço (Quality of Service, QoS) define o nível de entrega objetivo de um determinado serviço de uma aplicação. No artigo [Shanhe Yi \(2015\)](#) os autores definem QoS em quatro pontos, conectividade, confiabilidade, capacidade e atraso, e a colocam como um desafio de toda aplicação em uma FCN. Baseado nesses quatro pontos algumas métricas podem ser avaliadas para determinar o QoS de uma aplicação em uma determinada rede como: o atraso na entrega dos pacotes, que mede a latência de um pacote ser enviado e recebido pelo requisitante, a perda desses pacotes na rede, que mede a taxa de entrega de informação, a largura média de banda, que mede a capacidade da rede, etc.

2.5.2 QoE

A qualidade de experiência (Quality of Experience, QoE) define o nível de entrega subjetivo de uma aplicação. O QoE envolve diversos elementos dependendo do tipo de aplicação que está sendo avaliada. Podemos efetuar diversas formas de avaliação com o usuários para determinar o sentimento do usuário ao utilizar a aplicação avaliada, porém, conseguimos extrair informações subjetivas de algumas métricas objetivas, a fim de prever de forma mais exata se a aplicação está performando de uma forma agradável para o usuário.

Em aplicações de vídeo foram desenvolvidas algumas métricas matemáticas de QoE em relação a reprodução final do vídeo, a fim de determinar a experiência final do usuário. Existem métricas que consideram a similaridade do vídeo enviado com um vídeo em perfeito estado como similaridade estrutural (Structural Similarity Index, SSIM) e o ruído do vídeo transmitido (Peak signal-to-noise ratio, PSNR), ambas as métricas tem a intenção de representar a percepção do usuário em relação ao vídeo transmitido de forma matemática.

Alguns outros elementos de uma aplicação de vídeo podem ser analisados para avaliar o impacto final da aplicação sobre a experiência do usuário. Eventos de mudança de qualidade no caso das aplicações com o DASH, supondo que qualidades maiores e mais próximas a original do vídeo agradem mais o usuário do que qualidades inferiores. A taxa de atraso durante a reprodução do vídeo, pois travamentos ou descontinuidades do vídeo podem afetar a experiência do usuário. O atraso da reprodução do vídeo, indicando travamentos que podem afetar a experiência, entre outros aspectos mais específicos de cada aplicação.

Capítulo 3

Trabalhos Relacionados

3.1 VANETs

Diversos estudos vem sido desenvolvidos no âmbito dos desafios e soluções para a implementação real destas redes, como o manejo eficiente dos nós da rede, o gerenciamento dos recursos disponíveis nos dispositivos, o gerenciamento dos recursos de rede, segurança e privacidade, o consumo eficiente de energia e a comunicação e coordenação dos elementos integrantes da rede (Rong Yu e Stein Gjessing (2013); SALIM BITAM e ZEADALLY (2015); Tesnim Mekki (2017)).

3.2 Transmissão de vídeo

Para o tópico de transmissão de vídeo existem alguns estudos que explicam e tentam propor de forma prática o uso de aplicações de vídeo para cenários como os de *VANETs*. Como é o caso de Immich (2017), onde o autor explica os conceitos chaves para a realização e avaliação de uma transmissão de vídeo. No quesito de transmissão o autor demonstra os fatores que influenciam em uma transmissão como a perda de frames chave para uma determinada codificação de vídeo e os fatores de compressão e descompressão de um arquivo que podem gerar perdas durante a transmissão. Em relação as análises de qualidade dessas transmissões o autor propõe e aplica métricas subjetivas de QoE e métricas objetivas de QoS. O autor também propõem modos de otimizar a transmissão do vídeo nesses cenários.

Em E. Setton (2018) e Shiwen Mao (2001) os autores propõem alguns mecanismos de rede e de transmissão de vídeo para se obter uma forma resiliente de transmissão de vídeo em redes Ad-Hoc sem fio. Em E. Setton (2018) o autor está mais preocupado com o design da estrutura de rede utilizada e propõem uma forma de estruturar a rede para que seja possível o melhor aproveitamento para este tipo de mídia, enquanto os autores de Shiwen Mao (2001) possuem uma proposta de uma técnica de transmissão em redes deste gênero para se obter o melhor aproveitamento da transmissão em tempo real dos arquivos de mídia de vídeo.

3.3 Transmissão sem fio

Outro ponto de estudo em redes veiculares são as tecnologias de comunicação sem fio, existem algumas formas possíveis para a comunicação nestas redes. Nos trabalhos de Immich (2017); Rong Yu e Stein Gjessing (2013); Shanhe Yi (2015); Tesnim Mekki (2017) os autores citam um padrão de Wi-Fi específico para o cenário como o WAVE, para a comunicação entre os elementos veiculares destas redes, em geral padrões de curto e médio alcance.

Em [Rong Yu e Stein Gjessing \(2013\)](#); [Teshim Mekki \(2017\)](#) os autores também citam o uso das tecnologias de longo alcance como 3G e 4G como uma opção para a transmissão e conexão entre essas redes e os elementos não veiculares destas redes e/ou com a comunicação direta entre esses elementos da rede e as nuvens convencionais que são responsáveis pelos processamentos pesados. Porém, nenhum desses trabalhos anteriores cita ou idealiza o uso da tecnologia 5G que vem sendo desenvolvida. Como contraponto nos trabalhos [Mohsen Nader Tehrani e Yanikomeroglu; Rupendra Nath Mitra \(2015\)](#) os autores analisam e propõem a tecnologia 5G como uma opção de comunicação de longo alcance para esses tipos de rede.

3.4 Simulação de cenários de rede

Devido a complexidade de implementação real que o cenário das *VANETs* apresentam, diversos simuladores e frameworks foram criados com o intuito de simular de forma realista esses ambientes. Alguns trabalhos demonstram a capacidade e eficiência que o OMNeT++ [OMNeT++ \(2018\)](#), um simulador de redes que possui diversos frameworks de trabalho, pode ter para simulações de aplicações em ambientes de redes. Em [Denis Rosário \(2013\)](#) os autores propõem uma forma de realizar simulações de vídeo através de um framework proposto por eles que já possui as ferramentas de análise tanto de QoE quanto de QoS, o M3WSN FRAMEWORK que funciona como extensão para o OMNeT++. Neste artigo é feita a proposta e análise do framework que possui embarcada as funcionalidades de transmissão e análise de mídia de vídeo, com a integração das funcionalidades do Evalvid, um outro framework de análise da transmissão de arquivos de vídeo. Mas, neste trabalho os autores não abordam o uso desses simuladores em ambientes de VANETs e não citam a possibilidade de integração com simuladores como Veins e SUMO para a implementação deste tipo de rede.

Outros trabalhos, como [Jair Jose Ferronato \(2015\)](#); [Noori \(2012\)](#), fazem uso da combinação do OMNeT++ com outros de seus frameworks como o Veins e o SUMO, para simularem tráfego urbano de forma realista. Nestes dois trabalhos os autores analisam de forma objetiva a capacidade de se realizar uma simulação de VANETs em conjunto com uma simulação realista de tráfego com a combinação dos três frameworks citados acima, porém, eles não analisam o aspecto da integração com alguns outros tipos de simuladores como o INET para as simulações mais aproximadas de redes sem fio.

Capítulo 4

Análise dos cenários com o DASH

4.1 Cenários Experimentais

As VANETs tem sua implementação em diversos tipos de cenários (Shanhe Yi (2015); Tesnim Mekki (2017)), baseado em alguns desses casos de uso foram definidos dois cenários experimentais onde poderiam ser avaliadas as métricas de QoE e QoS da aplicação que implementa a técnica DASH.

O primeiro cenário consiste em uma simulação de duas estradas que se cruzam em um determinado ponto, onde existem duas antenas de LTE 5G que fazem a transmissão entre os veículos e as redes locais de FNC e este por sua vez faz a conexão ao servidor de CC alocado distante do cenário. Neste cenário, os carros se movimentam em velocidades relativamente altas, entorno de 90 - 110 km/h, como mostra a figura 4.1. O segundo consiste de um cenário de simulação das rotas da cidade universitária da USP em São Paulo e possui a mesma configuração de rede que o cenário anterior porém, variando sua velocidade de 20 - 50 km/h, como demonstra a figura 4.2. Ambos os cenários tem seus arquivos de movimentação e de rotas descritos no repositório Git (Git-patrickpa (2018)) deste trabalho e foram gerados utilizando o OpenStreetMap (OpenStreetMap (2018)) e o SUMO (Sumo (2018)).

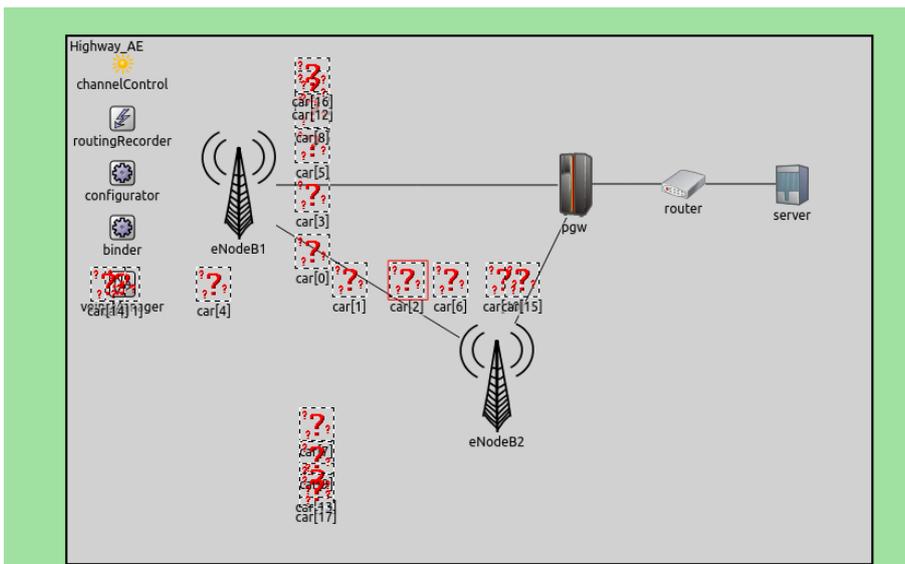


Figura 4.1: Cenário de simulação do OMNeT++ que representam as duas rodovias se cruzando.

Para as avaliações experimentais foram preparadas simulações com dez carros que possuem a aplicação, gerando lotes de experiências para cada cenário veicular e de transmissão

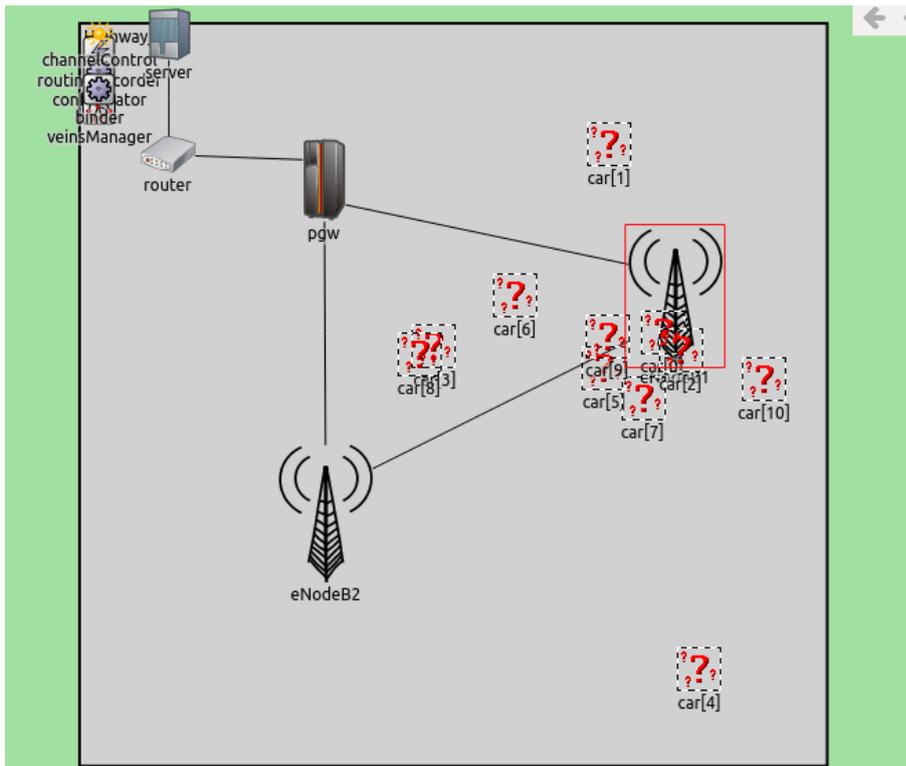


Figura 4.2: Cenário de simulação do OMNeT++ que representa a cidade universitária da USP em São Paulo.

de vídeo. Dentro desses cenários o número total de carros sendo simulados poderia variar sempre acima do necessário de carros com a aplicação, podendo ir de trinta a setenta carros conectados a rede. No geral, os carros que não possuíam a aplicação em análise estavam rodando aplicações genéricas ou estavam apenas conectados a rede sem trocar informações com o servidor de CC para fins de deixar a simulação mais realista.

Cada simulação dos experimentos utiliza uma janela de setenta segundos para a avaliação do envio de dois vídeos pela aplicação, um vídeo de treze segundos mais dinâmico que representa uma rua com diversas pessoas se movimentando e um outro vídeo de dez segundos com alguns carros parando em um semáforo de trânsito, este um pouco menos dinâmico e com menos elementos se movimentando. Ambos os vídeos estão disponíveis na plataforma do YouTube (Bill (2018a,b)) e foram analisados durante as simulações deste trabalho. Para a facilitação da análise dos dados o DASH foi configurado com um MPD de intervalo de um segundo para cada pacote e sua variação do vídeo, portanto, podemos ter variação de qualidade do vídeo a cada um segundo de simulação, ou seja, a cada segundo de simulação deveríamos ter um segundo de reprodução do vídeo a partir da primeira requisição da aplicação.

Os cenários levam em conta a capacidade de banda e de alcance das redes 5G, tanto para os emissores quanto para os receptores. Todos os demais detalhes e aspectos técnicos das simulações podem ser analisados no repositório desta monografia [Git-patrickpa \(2018\)](#), pois nem todos eles são de relevância para o entendimento das simulações e seus resultados estatísticos.

4.2 Análise dos Resultados

Como já foi mencionado anteriormente neste trabalho, existem métricas que podem avaliar o desempenho objetivo da aplicação na rede, QoS, e métricas que medem as qualidades mais subjetivas para o usuário da aplicação, QoE. Ambas refletem o desempenho geral da aplicação e o impacto final dela para o usuário em um cenário experimental.

Para a análise de QoS da aplicação foram definidas duas métricas para avaliação deste parâmetro. A primeira delas é a perda de pacotes médios na rede, essa métrica consiste em avaliar quantos pacotes a aplicação requisitou e quantos pacotes ela recebeu do servidor, assim podemos medir a consistência e confiabilidade da rede em relação a aplicação. A segunda métrica de avaliação é o atraso médio de entrega dos pacotes na rede pela aplicação, assim podemos medir o tempo de atraso entre a aplicação requisitar um pacote de vídeo e receber esse pacote para exibição.

Utilizando essas duas métricas nós podemos avaliar a performance objetiva da aplicação em relação a rede e o usuário, pois a perda elevada de pacotes significa uma alta falta de informação para a reprodução do vídeo e um alto atraso de entrega de pacotes pode significar uma demora para reprodução do vídeo que o usuário requisitou.

Para a análise de QoE da aplicação foram definidas três métricas de avaliação desse parâmetro. A primeira delas é a qualidade média do vídeo em todos os nós que continuam a aplicação, assim podemos avaliar se o vídeo foi reproduzido em uma qualidade alta, média ou baixa, sendo essas respectivamente 720p (1500 Kbps), 540p (800 Kbps) e 320p (400 Kbps). Portanto, podemos avaliar a experiência do usuário quanto ao vídeo reproduzido, pois podemos supor que vídeos com uma qualidade maior agradam mais o usuário do que os de qualidade inferior. A segunda métrica é o atraso de reprodução dos segmentos do vídeo, cada segmento do vídeo deveria ser reproduzido a cada segundo após o anterior e se o atraso de reprodução de cada segmento for muito maior do que alguns mili segundos o usuário pode ter sua experiência comprometida devido a travamentos durante a reprodução. A terceira e última métrica é o tamanho do buffer por segundo de simulação, essa métrica tem como objetivo medir a quantidade de recarregamentos que aplicação faz do vídeo, pois em situações de muito recarregamento o usuário pode experimentar travamentos da aplicação e mudanças de qualidade devida a aplicação e não a rede. No caso da implementação da aplicação com o DASH utilizada nas simulações o tamanho mínimo do buffer para reprodução é de três segmentos e cada segmento representa um segundo de vídeo e seu tamanho máximo é de dez segmentos. Portanto, se houver pelo menos um segmento em buffer o vídeo é reproduzido, mas caso hajam apenas três acontece o recarregamento, a aplicação consegue carregar até dez segmentos, proporcionando ao usuário uma reprodução mais fluída do vídeo. Porém, em casos de recarregamento a aplicação tende a recarregar segmentos do vídeo para futuras reproduções.

Todos os resultados foram retirados de trinta simulações de cada combinação de cenário e vídeo e todos os scripts de avaliação e de geração dos cenários podem ser encontrados no repositório git deste trabalho [Git-patrickpa \(2018\)](#).

4.2.1 Avaliação do Cenário 1

O cenário um consiste no cenário das auto estradas que se cruzam e possuem uma velocidade média de 90 - 110km/h.

QoS - Atraso Médio na entrega de Pacotes	
Média	0.273 segundos
Desvio Padrão	0.314 segundos

Tabela 4.1: *QoS - Análise do tempo médio de atraso na entrega de pacotes com 10 carros com a aplicação, resumizando ambos os vídeos no cenário das auto estradas.*

QoS - Perda Média de Pacotes	
Média	0.971 %
Desvio Padrão	0.026 %

Tabela 4.2: *QoS - Análise da perda média na entrega de pacotes com 10 carros com a aplicação, resumizando ambos os vídeos no cenário das auto estradas.*

Tamanho médio do Buffer vs. Tempo - 10 carros x Cenário 1 x Vídeo 1

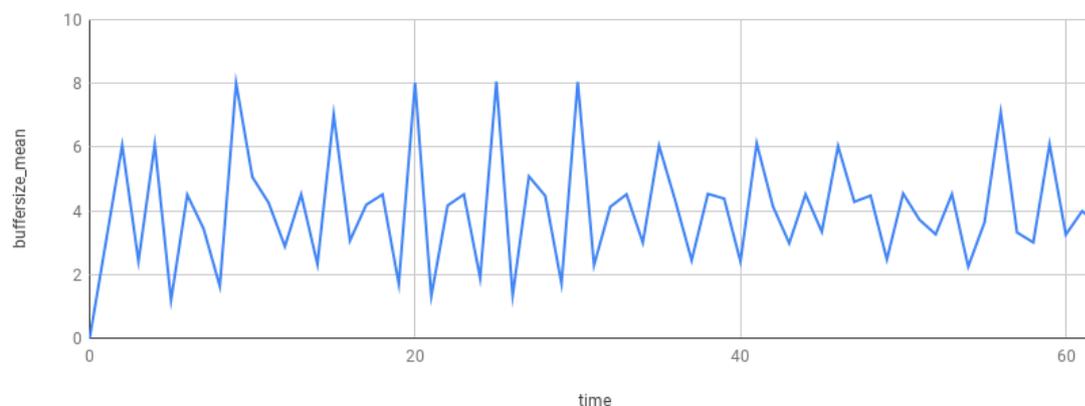


Figura 4.3: *QoE - Resultados do experimento de tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário das auto estradas.*

Em relação às métricas de QoS podemos analisar que segundo a tabela 4.1 o atraso médio dos pacotes em média possuem menos do que 0.3 segundo e o desvio padrão não desloca além de 0.3 segundo, o que nos permite inferir que os pacotes que chegaram tiveram um atraso geral abaixo de 1 segundo. Portanto, conseguimos inferir que em geral os pacotes requisitados chegam em menos de 1 segundo da requisição pela aplicação e que enquanto temos um segmento de 1 segundo do vídeo sendo executado o outro pacote para a exibição do próximo segmento já terá chegado, assim propiciando uma execução contínua dos vídeos em questão neste cenário.

Da mesma forma que o atraso dos pacotes a perda de pacotes é bem baixa, abaixo de 1% dos pacotes, segunda a tabela 4.2. Portanto, a confiabilidade da aplicação na rede é mantida e em geral podemos assumir que a aplicação teve um bom desempenho na rede quanto a garantia de entrega de pacotes nas diversas aplicações.

Qualidade Média x Tempo - 10 carros x Cenário 1 x Vídeo 1

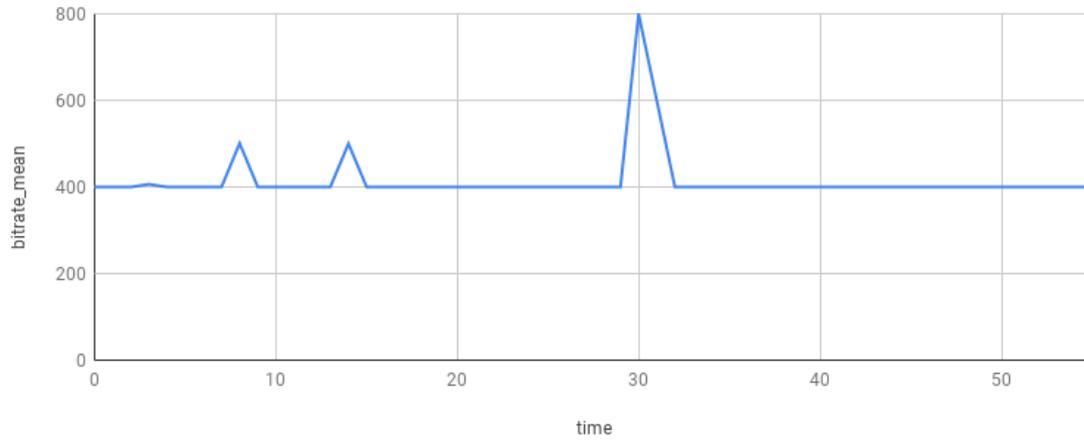


Figura 4.4: *QoE - Resultados do experimento da qualidade média do vídeo x tempo, com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário das auto estradas.*

Tamanho médio do Buffer vs. Tempo - 10 carros x Cenário 1 x Vídeo 2

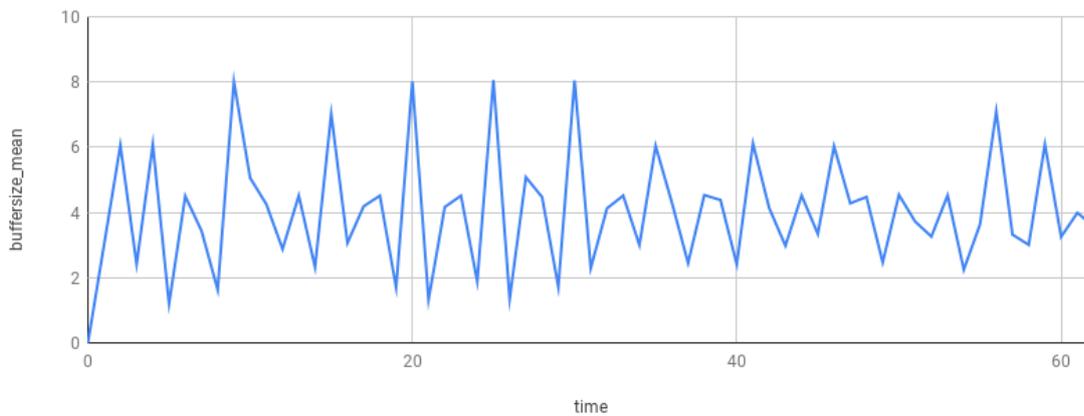


Figura 4.5: *QoE - Resultados do experimento de tamanho médio do buffer x tempo, com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário das auto estradas.*

QoE - Buffer Size	
Média	4.495
Desvio Padrão	2.752

Tabela 4.3: *QoE - Média e desvio padrão para o experimento tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo e no cenário das auto estradas.*

QoE - Bit Rate	
Média	402.051 Kbps
Desvio Padrão	7.105 Kbps

Tabela 4.4: *QoE - Média e desvio padrão para o experimento de qualidade média do vídeo x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário das auto estradas.*

Quanto as métricas de QoE, se analisarmos as figuras 4.3 e 4.5 podemos observar que os eventos de recarregamento acontecem de forma bem parecida tanto para o vídeo 1, de 13 segundos, quanto para o vídeo 2, de 9 segundos, e em geral não temos o buffer abaixo de 1, o que nos permite assumir que sempre possuíamos 1 segundo do vídeo para execução da aplicação no geral. Essa informação é corroborada pelas médias e desvios padrões que temos desta métrica nas tabelas 4.3 e 4.5, onde podemos observar que a média dos buffers se mantém perto de 4.5 fragmentos carregados e o desvio padrão fica em torno de 2.7. Portanto, a aplicação no geral sempre possui um segmento para o DASH reproduzir, os eventos de recarregamento acontecem, porém não parecem interferir na execução da aplicação.

Se analisarmos a figura 4.4 e a figura 4.6 é possível verificar que o vídeo de maior duração possui uma variação melhor da qualidade do vídeo, em alguns pontos ficando com a média em torno de 600 Kbps, e que o vídeo 2, de menor duração, tende a ficar com a sua variação em 400 kbps, o que pode ser pelo fato de a rede não melhorar para os carros a tempo de poder ocorrer a variação de qualidade, dado que a qualidade sempre inicia na mais baixa e melhora de acordo com o tempo e qualidade da rede. No geral, analisando as tabelas 4.4 e 4.6 podemos notar que a média da qualidade dos segmentos de vídeo é em torno dos 400 Kbps. Portanto, a rede não tem qualidade o suficiente, de acordo com os critérios do DASH, de transportar pacotes maiores de vídeo e assim ele mantém a qualidade baixa valorizando a reprodução contínua do vídeo, em vez da alta qualidade.

Segundo a tabela 4.7 podemos analisar que a média de atraso de reprodução da aplicação é inferior a 0.1 segundos, o que pode ocasionar alguns travamentos perceptíveis ao usuário, porém não muito significantes. Se levarmos em conta o desvio padrão, podemos inferir que em alguns casos a variação foi maior e o usuário pode experimentar alguns ligeiros travamentos entre o começo de um segmento e o início do próximo.

QoE - Buffer Size	
Média	4.685
Desvio Padrão	2.643

Tabela 4.5: *QoE - Média e desvio padrão para o experimento tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário das auto estradas.*

Qualidade Média x Tempo - 10 carros x Cenário 1 x Vídeo 2

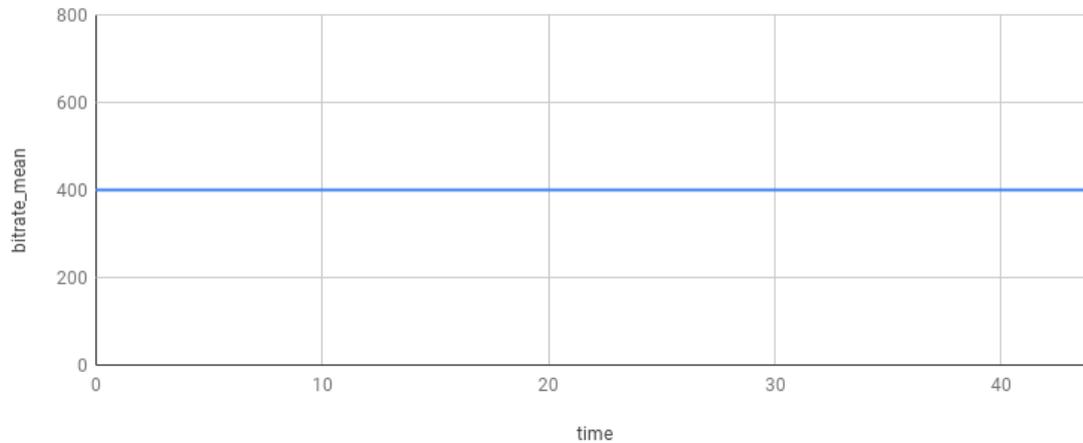


Figura 4.6: *QoE - Resultados do experimento da qualidade média do vídeo x tempo, com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário das auto estradas.*

QoE - Bit Rate	
Média	400.010 Kbps
Desvio Padrão	0.92 Kbps

Tabela 4.6: *QoE - Média e desvio padrão para o experimento de qualidade do vídeo x tempo com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário das auto estradas.*

QoE - Atraso de Reprodução	
Média	0.0912 segundos
Desvio Padrão	0.240 segundos

Tabela 4.7: *Média e Desvio Padrão do atraso de reprodução dos segmentos levando em consideração a análise dos dois vídeos.*

4.2.2 Avaliação do Cenário 2

O cenário dois consiste no cenário que simula as rotas internas de cidade universitária da USP em São Paulo e possuem uma velocidade média de 20 - 50km/h.

QoS - Atraso Médio na entrega de Pacotes	
Média	2.728 segundos
Desvio Padrão	2.962 segundos

Tabela 4.8: *QoS - Análise do tempo médio de atraso na entrega de pacotes com 10 carros com a aplicação, resumindo ambos os vídeos no cenário da cidade universitária.*

QoS - Perda Média de Pacotes	
Média	0.660 %
Desvio Padrão	0.596 %

Tabela 4.9: *QoS - Análise da perda média na entrega de pacotes com 10 carros com a aplicação, resumindo ambos os vídeos no cenário da cidade universitária.*

Tamanho médio do Buffer vs. Tempo - 10 carros x Cenário 2 x Vídeo 1

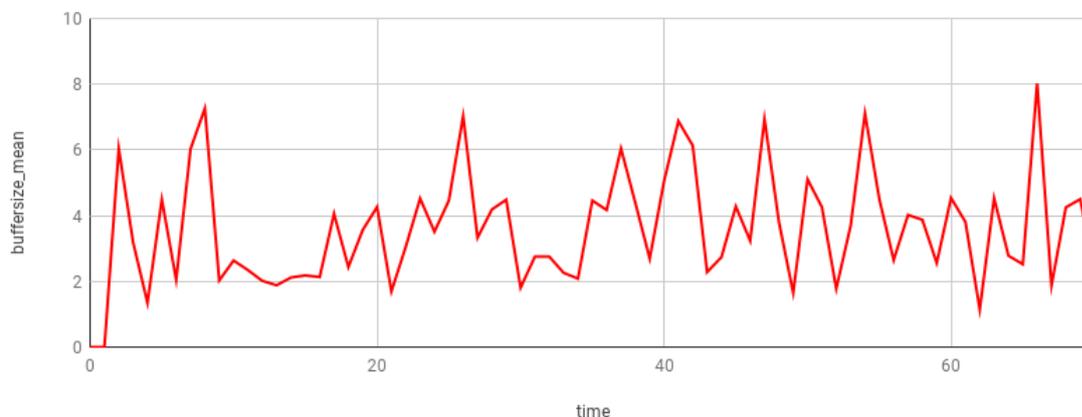


Figura 4.7: *QoE - Resultados do experimento de tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário da cidade universitária.*

Em relação as métricas de QoS do segundo cenário de avaliação temos que o atraso médio entre os pacotes é relativamente alto quando consideramos o caso de cada segmento de vídeo de 1 segundo, pois a média é de 2.7, segundo a tabela 4.8, o que representa mais do que o dobro em relação ao tempo de execução de um segmento e o desvio padrão corrobora a ideia. Portanto, podemos assumir que o atraso na rede é ruim em cenários do gênero, se comparado ao primeiro cenário.

Se analisarmos a perda média de pacotes temos que a relação é menor do que 1% dos pacotes segundo a tabela 4.9, então temos que a confiabilidade da aplicação em redes urbanas como a da USP é boa, permitindo uma alta entrega de informação na rede.

Quanto as métricas de QoE se analisarmos a média do tamanho do buffer, ele se comporta de forma muito parecido com o cenário 1 em ambos os vídeos, segundo as figuras 4.7 e 4.10. O tamanho médio do buffer está sempre acima de 1, o que nos permite assumir que sempre temos 1 segmento de vídeo pelo menos carregado para a reprodução em média. Os eventos

QoE - Buffer Size	
Média	3.599
Desvio Padrão	2.300

Tabela 4.10: *QoE - Média e desvio padrão para o experimento tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário da cidade universitária.*

Qualidade Média x Tempo - 10 carros x Cenário 2 x Vídeo 1

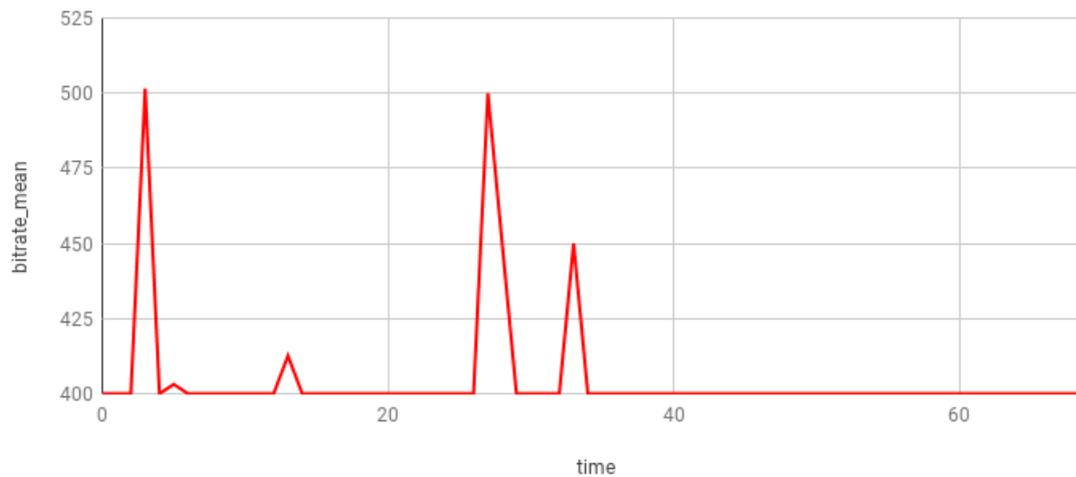


Figura 4.8: *QoE - Resultados do experimento da qualidade média do vídeo x tempo, com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário da cidade universitária.*

QoE - Bit Rate	
Média	402.461 Kbps
Desvio Padrão	8.527 Kbps

Tabela 4.11: *QoE - Média e desvio padrão para o experimento de qualidade média do vídeo x tempo com 10 carros com a aplicação, reproduzindo o primeiro vídeo no cenário da cidade universitária.*

QoE - Buffer Size	
Média	3.792
Desvio Padrão	2.455

Tabela 4.12: *QoE - Média e desvio padrão para o experimento tamanho médio do buffer x tempo com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário da cidade universitária.*

QoE - Bit Rate	
Média	400.034 Kbps
Desvio Padrão	0.81 Kbps

Tabela 4.13: *QoE - Média e desvio padrão para o experimento de qualidade do vídeo x tempo com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário das auto estradas.*

Qualidade Média x Tempo - 10 carros x Cenário 2 x Vídeo 2

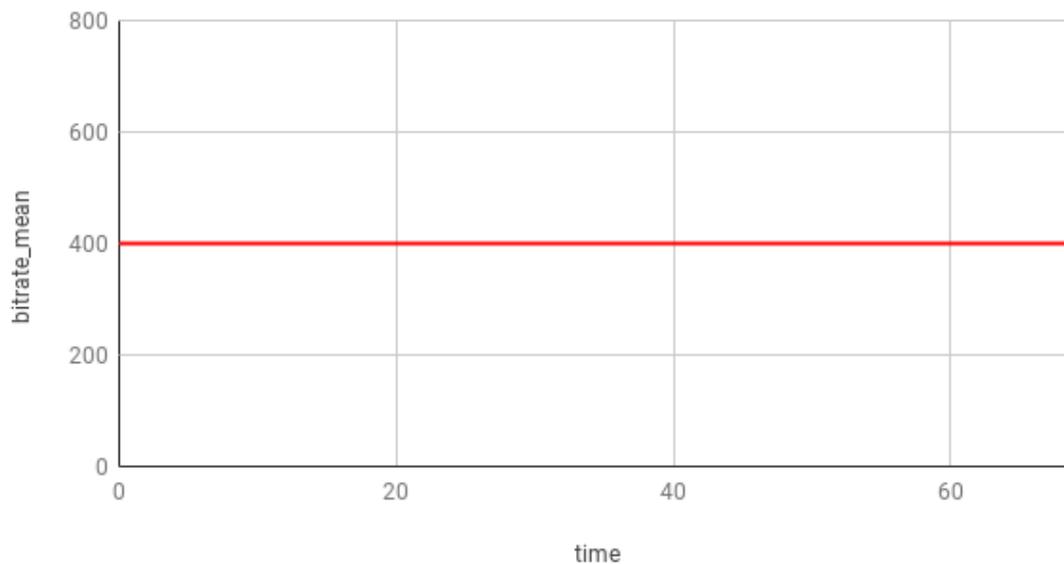


Figura 4.9: *QoE - Resultados do experimento de tamanho médio do buffer x tempo, com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário da cidade universitária.*

Tamanho médio do Buffer vs. Tempo - 10 carros x Cenário 2 x Vídeo 2

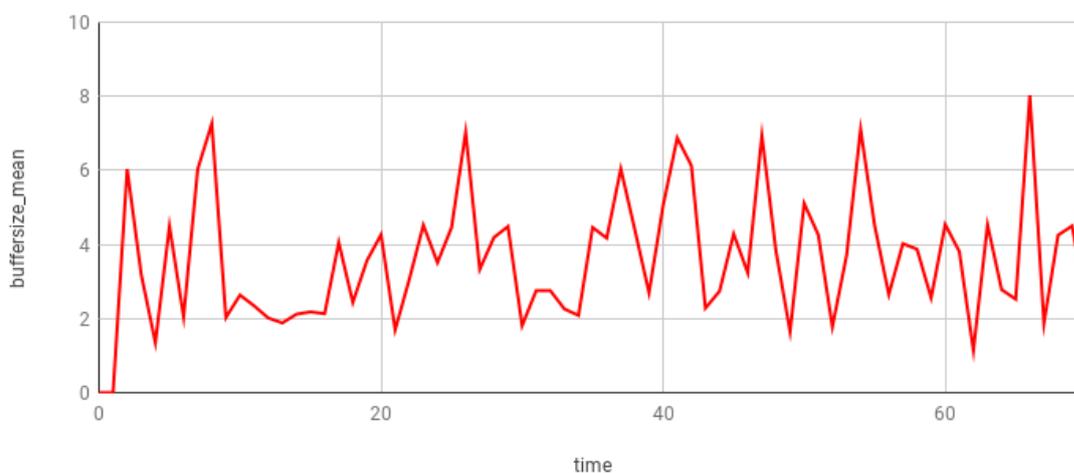


Figura 4.10: *QoE - Resultados do experimento da qualidade média do vídeo x tempo, com 10 carros com a aplicação, reproduzindo o segundo vídeo no cenário da cidade universitária.*

 QoE - Atraso de Reprodução

Média	1.339 segundos
Desvio Padrão	1.898 segundos

Tabela 4.14: *Média e Desvio Padrão do atraso de reprodução dos segmentos levando em consideração a análise dos dois vídeos.*

de recarregamento acontecem, porém, mantém o mínimo de segmentos para a execução do vídeo. Esse ponto pode ser uma vantagem para a aplicação em cenários como esse, pois mesmo em cenário altamente densos como o da USP a aplicação mantém o buffer sempre adequado para a reprodução do vídeo.

No caso da métrica de qualidade média dos vídeos temos algo muito semelhante ao cenário 1, como mostram as figuras 4.8 e 4.9, onde podemos analisar que no primeiro vídeo a qualidade chega a variar em alguns pontos saindo de 400 kbps, porém, no vídeo 2 a qualidade se mantém estável na qualidade mais baixa, o que as tabelas 4.11 e 4.13 corroboram quanto a média e desvio padrão das qualidades. O que pode refletir em uma qualidade ruim de experiência para o usuário.

A última métrica de QoE deste cenário o atraso na reprodução demonstra algo parecido com o atraso nos pacotes, porém é atenuada pelo efeito de bufferização da aplicação, onde podemos ver uma média de 1.339 segundos e um desvio padrão de 1.898 segundos, como demonstra a tabela 4.14. Portanto, temos que o usuário pode experimentar travamentos longos o suficiente para que afete sua qualidade de experiência quanto a aplicação em cenários urbanos e altamente densos, como o da USP.

Capítulo 5

Conclusões

Considerando as métricas analisadas no capítulo quatro deste trabalho, podemos avaliar que o algoritmo do DASH se comporta melhor em cenários menos densos, mesmo que estes apresentem um dinamismo de movimento maior, como pode ser observado no cenário das auto estradas. Porém, pode ter seu desempenho prejudicado quando tratamos de ambientes mais densos e com um dinamismo maior de nós na rede, como o cenário da cidade universitária.

Esses dados nos levam a crer que a técnica é implementável em cenários como o das rodovias que são grande parte dos cenários onde no futuro, com grandes viagens e carros autônomos, pode ser aproveitado para a reprodução de mídias digitais durante os trajetos. Porém, em casos urbanos como o de grandes cidade a técnica deve ser adaptada para obter um melhor desempenho, se aproveitando das partes mais eficientes que ela demonstra, como a sua alta taxa de adaptação e seu carregamento em buffer eficiente. Outro ponto a se ressaltar é o melhor desempenho com vídeos maiores, mesmo que tendo alguns segundos de diferença o vídeo 1 com 13 segundos obteve um melhor resultado da técnica, o que nos leva a concluir que se levarmos em conta vídeos maiores que permitam a ferramenta a agir de forma mais duradora podemos obter resultados melhores.

Temos então que a técnica do DASH pode ser uma opção para transmissões em redes como as VANETs no futuro e se pensarmos em adaptações mais inteligentes, que tirem proveito dos cenários mais densos, como carregamentos prévios em nós das FNC, podemos ter um desempenho maior durante a reprodução de vídeos.

Bibliografia

- Adobe Digital Index(2017)** Adobe Digital Index. A look at the 2017 adobe mobile maturity survey. Relatório técnico, Adobe. Citado na pág. 3
- Bill(2018a)** Barnacle Bill. Busy street scene with shoppers and commuters hd file stock footage - <https://www.youtube.com/watch?v=lgfamifxmuc>, 2018a. Citado na pág. 18
- Bill(2018b)** Barnacle Bill. Cars stop at traffic light manchester uk circa 2013 stock footage - <https://www.youtube.com/watch?v=9eid2bh8mo>, 2018b. Citado na pág. 18
- Cisco(2017)** Cisco. Cisco visual networking index: Global mobile data traffic forecast update, 2016–2021, 2017. Citado na pág. 3
- Denis Rosário(2013)** Claudio Silva Eduardo Cerqueira Torsten Braun Denis Rosário, Zhongliang Zhao. An omnet++ framework to evaluate video transmission in mobile wireless multimedia sensor networks. Citado na pág. 10, 16
- E. Setton(2018)** Xiaoqing Zhu A. Goldsmith B. Girod E. Setton, Taesang Yoo. Cross-layer design of ad hoc networks for real-time video streaming. Citado na pág. 15
- Git-patrickpa(2018)** Git-patrickpa. <https://github.com/patrickpa/vanet-dash-tcc>, 2018. Citado na pág. 17, 18, 19
- Immich(2017)** Roger Kreutz Immich. Mechanisms for resilient video transmission. Citado na pág. 5, 15
- INET(2018)** INET. <https://inet.omnetpp.org/>, 2018. Citado na pág. 11
- Jair Jose Ferronato(2015)** Marco Antonio Sandini Trentin Jair Jose Ferronato. Simulação de vanets utilizando ferramentas omnet++, sumo e veins. Citado na pág. 10, 16
- Mohsen Nader Tehrani e Yanikomeroğlu()** Murat Uysal Mohsen Nader Tehrani e Halim Yanikomeroğlu. Citado na pág. 16
- MPEG-DASH(2018)** Bitmovin MPEG-DASH. <https://bitmovin.com/dynamic-adaptive-streaming-http-mpeg-dash/>, 2018. Citado na pág. 6
- Noori(2012)** Hamed Noori. Realistic urban traffic simulation as vehicular ad-hoc network (vanet) via veins framework. Citado na pág. 8, 16
- OMNeT++(2018)** OMNeT++. <https://www.omnetpp.org/>, 2018. Citado na pág. 9, 10, 11, 16
- OpenStreetMap(2018)** OpenStreetMap. <https://www.openstreetmap.org/traces>, 2018. Citado na pág. 17

- Rong Yu e Stein Gjessing(2013)** Yan Zhang Rong Yu e Kun Yang Stein Gjessing, Wenlong Xia. Toward cloud-based vehicular networks with efficient resource management. Citado na pág. [3](#), [8](#), [15](#), [16](#)
- Rupendra Nath Mitra(2015)** Dharma P. Agrawal Rupendra Nath Mitra. 5g mobile technology: A survey. Citado na pág. [16](#)
- SALIM BITAM e ZEADALLY(2015)** ABDELHAMID MELLOUK SALIM BITAM e SHERALI ZEADALLY. Vanet-cloud: A generic cloud computing model for vehicular ad hoc networks. Citado na pág. [3](#), [15](#)
- Shanhe Yi(2015)** Qun Li Shanhe Yi, Cheng Li. A survey of fog computing: Concepts, applications and issues. Citado na pág. [3](#), [7](#), [8](#), [13](#), [15](#), [17](#)
- Shiwen Mao(2001)** S.S. Panwar Yao Wang Shiwen Mao, Shunan Lin. Reliable transmission of video over ad-hoc networks using automatic repeat request and multipath transport. Citado na pág. [15](#)
- SimuLTE(2018)** SimuLTE. <http://simulte.com/index.html>, 2018. Citado na pág. [12](#)
- Sumo(2018)** Sumo. <http://sumo.dlr.de/index.html>, 2018. Citado na pág. [12](#), [17](#)
- Tesnim Mekki(2017)** Abderrezak Rachedi Maher ben Jemaa Tesnim Mekki, Issam Jabri. Vehicular cloud networks: Challenges, architectures, and future directions. Citado na pág. [3](#), [8](#), [15](#), [16](#), [17](#)
- Veins(2018)** Veins. <http://veins.car2x.org/>, 2018. Citado na pág. [12](#), [13](#)

Parte II
Parte Subjetiva

Desafios e Frustrações

O trabalho de conclusão de curso como um todo foi um grande desafio, e como todo grande desafio essa jornada de um ano de trabalho teve seus altos e baixos gerando algumas frustrações. O trabalho em si é a continuação de uma iniciação científica (IC) que realizei com meu professor orientador Alfredo Goldman e com um colega de curso Anderson Andrei, dela tivemos a ideia de expandir um pouco as simulações e criar um trabalho mais próximo do real em relação as redes veiculares, pois durante a IC elaboramos um cenário de redes de celulares, por ser mais fácil de entender e construir, dado o nosso conhecimento com os simuladores.

Todo esse trabalho de expansão do que já havia sido feito na IC me tomou muito tempo para compreender e elaborar um novo cenário de simulação e dessa vez que tivesse todos os elementos o mais realistas possíveis. Todo o aprendizado com os simuladores, que até então não eram o suficiente para o desenvolvimento deste trabalho, com as técnicas de transmissão de vídeo, que no início eu não tinha muito conhecimento, as tentativas de análises de QoE e QoS, com as frustrações de algumas análises que nós queríamos realizar e não foram possíveis, o fato de ainda cursar matérias da graduação, que me tomavam tempo, e estar estagiando os cinco dias da semana. Todos esses pontos foram grandes desafios, porém, com tempo e dedicação todas as frustrações e desafios foram sendo superados e esse trabalho foi tomando forma e agora escrevo isso feliz por ter aceitado todos os desafios e ultrapassado todos os momentos ruins.

A contribuição do curso

O curso de ciência da computação me ajudou muito em diversos pontos do desenvolvimento deste trabalho de conclusão, aqui vou citar algumas matérias que me ajudaram a decorrer do desenvolvimento:

- MAC0121 Algoritmos e Estruturas de Dados I - Essa matéria me iniciou no mundo dos algoritmos e das estruturas de dados, o que foi fundamental para o entendimento dos simuladores e as estruturas de dados que são utilizadas para a construção das simulações. Ao me ensinar sobre algoritmos eu pude entender e utilizar os algoritmos de codificação e de transmissão de vídeo;
- MAC0323 Algoritmos e Estruturas de Dados II - Da mesma forma que a matéria anterior, essa matéria me introduziu ao segundo nível dos algoritmos e estruturas de dados, me propiciando uma visão ainda melhor, mais construtiva e mais crítica acerca das soluções e formas de implementação dos algoritmos;
- MAC0219 Programação Concorrente e Paralela - Essa disciplina foi a primeira a qual eu tive contato com algoritmos paralelos, o que facilitaram muito o entendimento para execução de simulações em paralelo durante o desenvolvimento deste trabalho. Ela também foi onde eu tive contato com o professor Alfredo, que me iniciou no mundo acadêmico e o responsável por este trabalho ser como é hoje;
- MAC0338 Análise de Algoritmos - Durante o curso de computação essa matéria foi um dos maiores desafios, mas o mesmo tempo uma das matérias que tiveram uma fundamental importância na forma de enxergar os algoritmos, me propiciando assim uma forma nova de analisar e desenvolver programas.

Certamente todas as matérias deste curso foram, de alguma forma, importantes para o meu crescimento acadêmico, profissional e intelectual, porém deixei listado aqui aquelas que mais me marcaram para a realização deste trabalho.

Próximos Passos

Da mesma forma que minha IC foi um primeiro passo para esse trabalho espero que ele seja um passo para trabalhos futuros relacionados a este tema, que tem despertado minha curiosidade e obstinação. Ao finalizar este trabalho pretendo continuar meus estudos em relação ao tema de transmissão de dados em *VANETs* e explorar as possibilidades que esses estudos podem me trazer, possivelmente com a produção de algum artigo acadêmico referente ao tema.

Ainda tenho muito a explorar no âmbito das redes veiculares, como o teste delas em ambientes de cidades inteligentes com outros módulos conectados as redes e trocando informações entre eles, para simular uma cidade realmente conectada, interativa e o impacto tecnológico destas transformações em nossa sociedade. Além de existirem outras aplicações a serem analisadas nestas redes, como controle inteligente de tráfego, prevenção de acidentes, uso eficiente dos veículos, etc. Espero que esse trabalho me abra portas para aprender mais e desenvolver mais em relação a temas do âmbito de cidades inteligentes.

Agradecimentos

Eu gostaria de agradecer a todos aqueles que me ajudaram durante a jornada deste último ano. Foi um ano complicado, cheio de tarefas e responsabilidades, porém um ano de finalização de um ciclo que é a minha graduação.

Gostaria de agradecer especialmente aos meus orientadores que estiveram comigo em toda a jornada de produção deste trabalho e que me auxiliaram mesmo tendo toda a demanda normal que já possuem em suas funções profissionais.

Gostaria de agradecer minha família por ter me apoiado desde o início da graduação e nunca terem me deixado desistir do curso de computação. E por fim, gostaria de agradecer a pessoa que esteve do meu lado durante todos os momentos da produção deste trabalho e que se não fosse por ela eu nunca conseguiria ter terminado ele, minha namorada, que me aguentou morando comigo durante a confecção de todo o trabalho, aguentou minhas inseguranças em relação a ele e aguentou as minhas enlouquecidas por conta de as vezes as coisas não saírem como o planejado.

A todos vocês meu muito obrigado e sem vocês eu nunca conseguiria ter terminado este trabalho!