

INSTITUTO FEDERAL DO PARÁ - CAMPUS BELÉM  
CURSO DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE  
SISTEMAS

JOSÉ RENAN GAIA GALHARDO

**UMA FERRAMENTA DE DISTRIBUIÇÃO DE VÍDEO EM REDES  
VEICULARES**

BELÉM  
2017

JOSÉ RENAN GAIA GALHARDO

**UMA FERRAMENTA DE DISTRIBUIÇÃO DE VÍDEO EM REDES  
VEICULARES**

Trabalho de conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia do Pará - IFPA - Campus Belém. Como requisito para obtenção de Grau em Tecnologia em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Paulo Henrique Gonçalves Bezerra

BELÉM

2017

Dados para catalogação na fonte  
Setor de Processamento Técnico  
Biblioteca IFPA - Campus Belém

---

G155f Galhardo, José Renan Gaia.  
Uma Ferramenta de distribuição de vídeo em redes veiculares. /  
José Renan Gaia Galhardo. — Belém, 2017.  
79 f. : il.

Impresso por computador (fotocópia).  
Orientador: Paulo Henrique Gonçalves Bezerra.  
Trabalho de Conclusão de Curso (Tecnologia em Análise e  
Desenvolvimento de Sistemas) — Instituto Federal de Educação,  
Ciência e Tecnologia do Pará – IFPA, 2017.

1. Rede veicular. 2. Indicador de qualidade. 3. Aplicação. I.  
Título.

CDD: 004.68

---

JOSÉ RENAN GAIA GALHARDO

**UMA FERRAMENTA DE DISTRIBUIÇÃO DE VÍDEO EM REDES  
VEICULARES**

Trabalho de conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia do Pará - IFPA - Campus Belém. Como requisito para obtenção de Grau em Tecnologia em Análise e Desenvolvimento de Sistemas.

Data da defesa: 04/12/2017

Conceito: 9,0

---

Orientador: Prof. Me. Paulo Henrique Gonçalves Bezerra  
Instituto Federal do Pará – Campus Belém

---

Prof. Me. Adalberto Cavalcante de Melo  
Universidade da Amazônia

---

Prof. Me. Allan Douglas Bento da Costa  
Universidade da Amazônia

Dedico este trabalho a minha família e a minha namorada Amanda Fagundes, por todo o incentivo e ajuda para que isso se tornasse possível.

## **AGRADECIMENTOS**

Agradeço a todos que de alguma forma me ajudaram nessa caminhada.

## RESUMO

Nos últimos anos as redes sem fio estão passando por atualizações importantes, que originam domínios de aplicações. No caso das redes móveis surgiram as redes não estruturadas (MANETs), e posteriormente as VANETs que são as redes sem fio para veículos automotores. As VANETs tentam minimizar os problemas gerados nos grandes congestionamentos e melhorar a segurança nas rodovias, como por exemplo, a distribuição de conteúdo de vídeo em tempo real. Essas VANETs multimídia conseguem capturar e compartilhar, com isso, pode fornecer às autoridades de segurança pública diversas informações. Além disso, os veículos também podem contribuir entre si para a divulgação dos vídeos em quaisquer situações. Sendo assim, o objetivo deste trabalho foi desenvolver uma ferramenta na plataforma Android para indicar a qualidade do gateway (GQI) para distribuição de vídeo através da comunicação do veículo para infraestrutura (V2I), onde levou-se em consideração os requisitos de velocidade, localização e ângulo de gravação. Onde cada requisito fornecia a prioridade de acordo com o seu valor. A partir dessas prioridades obtidas realizou-se o cálculo do GQI, relacionando-os com os pesos de cada parâmetro. E o usuário poderia enviar ou não o seu vídeo.

Palavras - chave: Rede Veicular. Indicador de Qualidade. Aplicação.

## **ABSTRACT**

In recent years, wireless networks have been undergoing major updates, leading to application domains. In the case of mobile networks came the unstructured networks (MANETs), and later the VANETs which are the wireless networks for motor vehicles. VANETs try to minimize the problems generated by major congestion and improve road safety, such as the distribution of video content in real time. These multimedia VANETs can capture and share, thereby providing public safety authorities with a wealth of information. In addition, vehicles can also contribute to each other for the dissemination of videos in any situation. Therefore, the objective of this work was to develop a tool in the Android platform to indicate the quality of the gateway (GQI) for video distribution through the communication of the vehicle to infrastructure (V2I), where the requirements of speed, location and recording angle. Where each requirement provided priority according to its value. From these priorities were obtained the calculation of the GQI, relating them with the weights of each parameter. And the user could send or not your video.

Key - words: Vehicular network. Quality Indicator. Application.

## LISTA DE ILUSTRAÇÕES

Figura 1 - Comunicação V2V.....	22
Figura 2 - Comunicação V2I.....	23
Figura 3 - Comunicação V2V2I.....	24
Figura 4 - Comunicação V2I2V.....	24
Figura 5 - Arquitetura IEEE 802.11p WAVE.....	26
Figura 6 - Arquitetura de Acesso das Redes LTE.....	31
Figura 7 - Elementos da NodeB.....	32
Figura 8 - Cooperação entre NodeB.....	33
Figura 9 - Interface X2.....	34
Figura 10 - Degradações do Vídeo.....	36
Figura 11 - Estados de um <i>buffer</i> .....	37
Figura 12 - Degradação do Vídeo com o QoE.....	40
Figura 13 - Arquitetura do Android.....	51
Figura 14 - Modelagem do Caso de Uso da Aplicação.....	55
Figura 15 - Transmissão de Vídeo na Comunicação V2I.....	60
Figura 16 - Permissões para acesso dos recursos no Android.....	61
Figura 17 - Permissão solicitada pela aplicação.....	63
Figura 18 - Tela de Informações e NodeB próximas.....	64
Figura 19 - Menu com opções.....	65
Figura 20 - Alteração do Mapa.....	66
Figura 21 - Tela de Gravação e Envio.....	67

## LISTA DE TABELAS

Tabela 1 - Principais características dos protocolos de roteamento baseados em cluster ....	49
Tabela 2 - Distriuição das Versões do Android .....	54
Tabela 3 - Parâmetro com as regiões de interesse e as prioridades do ângulo .....	56
Tabela 4 - Setores e Prioridades da localização do veículo e das NodeB. ....	57
Tabela 5 - Parâmetro de Prioridade da Velocidade.....	58
Tabela 6 - Parâmetro de Pesos .....	58

## LISTA DE SIGLAS

3GPP - 3rd Generation Partnership Project  
4G - 4 Geração  
API - Application Programming Interface  
ART - Runtime Android  
BER - Bit Error Rate  
CH - Cluster-heads  
DSRC - Dedicated Short-Range Communication  
EPC - Evolved Packet Core  
EPS - Evolved Packet system  
E-UTRAN - Evolved-Terrestrial Radio Access Network  
FCC Comissão Federal de Comunicações (Federal Communications Commission)  
FER - Frame Error Rate  
GPRS - General Packet Radio Services  
GPS - Sistema de Posicionamento Global (Global Positioning System)  
GQI - Indicador de Qualidade do Gateway  
HSPA+ - High Speed Packet Access  
HTTP - Hyper Text Transfer Protocol  
HTTPS - Hyper Tetx Transfer Protocol Secure  
IDE - Ambiente de Desenvolvimento Integrado (Integrated Development Environment)  
IEEE - Instituto de Engenheiros Eletricistas e Eletrônicos (Institute of Eletrical and Eletronics Engineers)  
INS - Sistema de Navegação Inercial (Inertial Navigation System)  
IP - Internet Protocol  
ISM - Industrial, Scientific and Medical  
LET - Tempo de Validade de Ligação  
LTE - Long Term Evolution  
MAC - Media Access Control  
MANET - Rede Ad-Hoc Móvel (Mobile Ad-Hoc Network)  
MDDC - Multiagent Driven Dynamic Clustering  
MME - Mobility Management Entity  
NFC - Near Field Communication  
OBU - Unidades de Bordo (On Board Unit)  
OFDMA - Orthogonal Frequency Division Multiple Access  
OHA - Open Handset Alliance  
OICA - Organização Internacional dos Construtores Automobilísticos (Organization Internacionale des Constructeurs d'Automobiles)  
P-GW - Packet Data Network Gateway  
PSOBUs - OBUs de Segurança Pública  
QoE - Qualidade de Experiência  
QoS - Qualidade dos Serviços  
RCP - Processador de Comandos de Recursos (Resource Command Processor)  
RM - Gerente de Recursos (Resource Manager)  
RNC - Radio Network Controller  
RSU - Unidade de Acostamento (Road Side Units)

RTSP - Real Time Streaming Protocol  
SAE - System Architecture Evolution  
SAE-GW - System Architecture Evolution Gateway  
SDK - Software Development Kit  
S-GW - Serving Gateway  
S.O - Sistema Operacional  
SRB - Selection Protocol for Limitation of Broadcast Storm Problem  
SSIM - Structural Similarity Index  
TCP - Transmission Control Protocol  
UC - Unidade de Computação  
UDP - User Datagram Protocol  
UMTS - Universal Mobile Telecommunication System  
VANET - Redes Veiculares (Vehicular Ad-Hoc Network)  
V2I - Veículo para Infraestrutura (Vehicle to Infrastructure)  
V2I2V - Veículo para Infraestrutura para Veículo (Vehicle to Infrastructure to Vehicle)  
V2V - Veículo para Veículo (Vehicle to Vehicle)  
V2V2I - Veículo para Veículo para Infraestrutura (Vehicle to Vehicle to Infrastructure)  
VQM - Video Quality Metric  
WAVE - Wireless Access in a Vehicular Environment

## LISTA DE ALGORITMOS

Algoritmo 1 - Algoritmo da Aplicação .....	56
--	----

## SUMÁRIO

<b>1 INTRODUÇÃO</b>	<b>15</b>
<b>1.1 Motivações e Desafios</b>	<b>17</b>
<b>1.2 Objetivos</b>	<b>18</b>
1.2.1 Objetivo Geral	18
1.2.2 Objetivos Específicos	18
<b>1.3 Delimitação do Trabalho</b>	<b>19</b>
<b>1.4 Estrutura do Trabalho</b>	<b>19</b>
<b>2. REFERENCIAL TEÓRICO</b>	<b>20</b>
<b>2.1 Redes Veiculares</b>	<b>20</b>
2.1.1 Comunicação V2X	22
2.1.2 Arquitetura IEEE 802.11p (WAVE)	25
2.1.3 Aplicações	27
2.1.3.1 Segurança	28
2.1.3.2 Entretenimento	28
2.1.3.3 Assistência ao Motorista	29
<b>2.2 Redes LTE 4G</b>	<b>29</b>
2.2.1 Arquitetura	30
2.2.2 Node B	31
2.2.3 Interface X2	33
<b>2.3 Transmissões de Vídeos</b>	<b>35</b>
2.3.1 Protocolo de Transporte	35
2.3.2 Degradação do QoS	37
2.3.3 Qualidade de Experiência (QoE)	39
2.3.3.1 Structural Similarity Index (SSIM)	41
2.3.3.2 Video Quality Metric (VQM)	42
<b>3 TRABALHOS RELACIONADOS</b>	<b>43</b>
<b>3.1 Técnica de Formação de Cluster</b>	<b>43</b>
3.1.1 Multiagent Driven Dynamic Clustering of Vehicles in Vanets (MDDC)	43
3.1.2 Engineering a Cooperative Content Distribution Framework for Highway Vehicular Communications (CHAINCLUSTER)	44
<b>3.2 Protocolos de Roteamento Baseados em Cluster</b>	<b>46</b>
3.2.1 A Passive Clustering Aided Routing Protocol for Vehicular Ad-Hoc Networks (PASSCAR)	46
3.2.1.1 Grau do Nó	47
3.2.1.2 Tempo de vida do Link	47
3.2.2 Forwarder Smart Selection Protocol for Limitation of Broadcast Storm Problem (SRB)	47
3.2.2.1 Transmissão Seletiva Confiável	48
3.2.3 Conclusão do Capítulo	49
<b>4 Materiais e Métodos</b>	<b>50</b>
<b>4.1 Android</b>	<b>50</b>
4.1.1 Arquiterura	50

4.1.1.1 Kernel	51
4.1.1.2 Bibliotecas	52
4.1.1.3 Android Runtime - ART	52
4.1.1.4 Android SDK	53
<b>4.2 Aplicação Android</b>	<b>53</b>
4.2.1 Hardwares	53
4.2.2 Software	53
4.2.3 Desenvolvimento	55
4.2.3.1 Implementação do Algoritmo	56
4.2.3.2 Obter o ângulo de Gravação	56
4.2.3.3 Obter a Localização	57
4.2.3.4 Obter Velocidade do Veículo	58
4.2.3.5 Calcular o Indicador de Qualidade do Gateway	58
4.2.3.6 Transmissão do Vídeo	59
4.2.4 Permissões de Acesso aos Recursos da Aplicação	60
<b>4.3 Conclusão do capítulo</b>	<b>61</b>
<b>5 Aplicação Desenvolvida</b>	<b>62</b>
<b>5.1 Funcionamento da Aplicação Proposta</b>	<b>62</b>
5.1.1 Primeiro Acesso	62
5.1.2 Valor do GQI e Mapa	63
5.1.3 Gravar e Enviar Vídeo	67
<b>6 CONSIDERAÇÕES FINAIS</b>	<b>68</b>
<b>6.1 Trabalhos Futuros</b>	<b>68</b>
<b>REFERÊNCIAS</b>	<b>69</b>
<b>APÊNDICE</b>	<b>73</b>

# 1 INTRODUÇÃO

As redes sem fio vêm passando por atualizações nas últimas décadas, dando origem a diversos domínios de aplicações, desde segurança, entretenimento e assistência ao motorista, no cenário de cidades inteligentes (BARBA *et. al.*, 2012). No domínio das redes móveis, surgem as redes não estruturadas, chamadas também de MANETs (Rede Ad-Hoc Móvel, do inglês *Mobile Ad-Hoc Network*), onde cada nó da rede pode assumir funções de sistema terminal ou de encaminhador, o que permite que a comunicação direta entre nós seja possível (CHO; SWAMI; CHEN, 2011).

Com o surgimento das redes sem fio para os veículos automotores, também chamadas de VANETs (Redes Veiculares, do inglês *Vehicular Ad-Hoc Network*), aliados aos sistemas embarcados e multimídias disponíveis atualmente nos veículos estão sendo propostas e, dessa forma, o conceito tradicional de viajar de carro está sendo modificado (MOHAMMAD; RASHEED; QAYYUM, 2011).

As VANETs vem chamando a atenção de pesquisadores e indústrias ao redor do mundo, a fim de minimizar os problemas causados pelos grandes congestionamentos e melhorar a segurança nas rodovias das grandes cidades (BRADAI; AHMED; BENSLIMANE, 2014). De acordo com Faouzi; Leung; Kurian (2011), o tráfego de vídeo vai representar mais de 90% dos dados IP em poucos anos, onde milhares de usuários irão produzir, compartilhar e consumir serviços multimídias de forma ubíqua, inclusive em seus veículos.

A título de exemplo, a distribuição de conteúdo multimídia em tempo real em VANETs está se tornando uma tendência, e concede aos motoristas e passageiros, novas experiências com vídeos (FAOUZI; LEUNG; KURIAN, 2011). Por consequência, diversas aplicações multimídias estão sendo desenvolvidas para as VANETs, tais como entretenimento, segurança e condução cooperativa (DIMITRAKOPOULOS, 2011).

As VANETs multimídias são adaptadas para capturar e compartilhar monitoramento ambiental, *vídeo surveillance*, acidentes de trânsito e aplicações de vídeo para cidades inteligentes. Um fluxo de *streaming* de vídeo pode fornecer às autoridades de segurança pública (por exemplo, a polícia rodoviária, paramédicos e bombeiros), as informações com mais precisão do que as mensagens de texto e,

desse modo, permitir a ação mais adequada para cada tipo de situação, reduzindo o tempo de atendimento e/ou resgates (DIMITRAKOPOULOS, 2011).

Os veículos também podem colaborar entre si para divulgar vídeos de situações perigosas, para informar visualmente pilotos e equipes de resgates, tanto na cidade ou em uma rodovia. Os vídeos gerados a partir dos veículos que trafegam nas rodovias podem ser transmitidos para usuários fixos – fora da rede veicular.

Diversas pesquisas na área de sistemas inteligentes de transporte, como a comunicação de Veículo para Veículo V2V (*Vehicle to Vehicle*), Veículo para Infraestrutura V2I (*Vehicle to Infrastructure*), Veículo para Veículo para Infraestrutura V2V2I (*Vehicle to Vehicle to Infrastructure*) e Veículo para Infraestrutura para Veículo V2I2V (*Vehicle to Infrastructure to Vehicle*) estão sendo desenvolvidas e testadas (NOORI; VALKAMA, 2013) e diferentes aplicações vão surgindo para os cenários urbanos e de rodovias.

De acordo com a OICA (Organização Internacional dos Construtores Automobilísticos, do inglês *Organization Internationale des Constructeurs d'Automobiles*), assim como a tecnologia apóia a comunicação entre pessoas, também haverá uma mudança na forma como os veículos interagem, aquela que é conhecida como comunicação V2V. Por meio dela, os automóveis serão capazes de viajar com mais proximidades uns dos outros, e redirecionar automaticamente os veículos para evitar condições climáticas perigosas ou estradas congestionadas. O erro humano será quase removido do volante, tornando a experiência de dirigir mais segura e agradável (BARFIELD; DINGUS, 2014).

Segundo os especialistas da área de transportes inteligentes, é admirável pensar que há poucos anos os *smartphones* não existiam e que atualmente estão incorporados à vida das pessoas e que alguns sistemas embarcados já estão presentes nos automóveis (BARFIELD; DINGUS, 2014), espera-se que os veículos autônomos irão reduzir os acidentes, que frequentemente são causados por erro dos motoristas.

Na atualidade já temos tecnologia que informa uma unidade sobre a saída da pista de um veículo, e corrigir e o manter dentro da pista (BARFIELD; DINGUS, 2014). Essa tecnologia está disponível em veículos como Mercedes e Lexus. Conforme passa o tempo, mais e mais veículos serão equipados com esse tipo de tecnologia, alterações graduais vão fornecer *drivers* com mais confiabilidade e maior aceitação do usuário.

Para alguns pesquisadores, o tempo para que esses veículos se tornem uma realidade para as pessoas, depende de como se dará a padronização. Ela é fundamental e os governos dos países terão que exigir que os fabricantes contenham tecnologias V2X nos veículos. Dentro de cerca de 10 anos veremos um veículo automatizado chegar ao mercado a preços acessíveis para muitos. E dentro de 20 anos, a maioria dos veículos vendidos serão automatizados e autônomos (NOORI; VALKAMA, 2013).

## **1.1 Motivações e Desafios**

E As limitações nas VANETs multimídias são severas. Por causa do tráfego multimídia, os desafios aumentam, a transmissão de fluxo contínuo de vídeo gera um grande volume de dados que devem ser transmitidos com baixa tolerância a atrasos, o que, para as VANETs, é um desafio, considerando os problemas causados pela alta mobilidade dos veículos.

As transmissões de vídeo ao longo das VANETs podem ser usadas para melhorar a eficácia da resposta dos agentes de segurança, em caso de acidentes. Um veículo em uma rodovia pode transmitir um vídeo ao vivo, do local do acidente. Esse fluxo estaria disponível para os paramédicos em ambulâncias que poderiam se preparar para o bom atendimento antes mesmo de chegar ao local do evento ocorrido. O mesmo fluxo pode ser encaminhado para os hospitais onde os paramédicos possam iniciar os preparativos para um pronto atendimento. O tratamento inicial dado pelos paramédicos também podem ser transmitidos aos hospitais, assim, auxiliando ainda mais os médicos a tomar decisões.

Diante das limitações e possibilidades, uma ferramenta para distribuição de vídeo que utiliza qualquer protocolo de roteamento, deve suprir os desafios impostos pela alta mobilidade. Em uma rodovia, a comunicação veicular se depara com grandes desafios como, por exemplo, no tempo que os veículos podem iniciar uma conexão entre si, por outro lado numa comunicação em que o veículo precisam se comunicar com a central pode haver a falta de conectividade ou atrasos nas entregas das mensagens.

Do ponto de vista da camada de rede, um dos principais desafios é garantir uma comunicação eficiente e confiável entre os veículos assim como a

Qualidade dos Serviços (QoS) disponibilizados. Do ponto de vista da camada de aplicação, um dos principais desafios é assegurar que o conteúdo multimídia transmitido tenha uma qualidade mínima assegurada, considerando os requisitos de Qualidade de Experiência (QoE), do ponto de vista do usuário.

Para as VANETs, devido à velocidade, localização, quebra dos enlaces e a falta de uma comunicação fim-a-fim entre os veículos, soluções baseadas em protocolos de Roteamento vêm sendo estudadas.

## 1.2 Objetivos

Este trabalho tem como objetivo a distribuição de vídeo gravado em tempo real a partir dos veículos, com uma qualidade mínima assegurada com o propósito de comunicar aos agentes de segurança pública os eventos ocorridos nas estradas.

Para Gerla *et. al.* (2014), a distribuição de conteúdos em VANETs de uma forma mais eficiente, utiliza a colaboração entre tecnologia de acesso. Neste trabalho, utilizaremos o *Long Term Evolution* (LTE), também conhecida como tecnologia móvel de quarta geração (4G), para uma comunicação V2I.

### 1.2.1 Objetivo Geral

O objetivo principal deste trabalho é desenvolver uma ferramenta para indicar a qualidade do *gateway* (GQI) para a distribuição do vídeo através da comunicação V2I.

### 1.2.2 Objetivos Específicos

- Desenvolver uma ferramenta para transmissão de vídeo;
- Implementar um algoritmo que considera a velocidade, localização e o vídeo gravado no veículo;
- Gerar o Indicador de qualidade do gateway;

### 1.3 Delimitação do Trabalho

A proposta do trabalho é para um cenário em uma rodovia com aplicações para VANETs multimídia que, devido aos problemas na comunicação V2V durante o processo de compartilhamento e transmissão de vídeos, encontram-se restrições causadas pelas limitações da arquitetura e protocolos. Considerando a aplicação para o cenário proposto, algumas características foram delimitadas inicialmente para o escopo deste trabalho:

- Aplicação de segurança pública;
- Cenário de rodovia;
- veículo possui interfaces LTE;
- Todos os veículos estão em sentido único;
- veículo possui câmera de gravação ou dispositivo de gravação com S.O (Android)
- ângulo será imposto em três sentidos (lado esquerdo, central e lado direito)
- veículo possui velocidades caracterizadas por intervalos (< 20, 20 a 39, 40 a 59, 60 a 79, 80 a 100 e > 100 Km/h);
- Na comunicação V2I será usada a rede LTE.

### 1.4 Estrutura do Trabalho

Este trabalho está estruturado da seguinte forma: o capítulo 1 é realizado uma introdução sobre os problemas e motivações, a cerca das redes veiculares e os objetivos da aplicação. No capítulo 2 é exposto o referencial teórico sobre as redes veiculares e os meios de comunicação. No capítulo 3 são apresentados os trabalhos relacionados, que tem como objetivo justificar na escolha do algoritmo da aplicação. No capítulo 4 são descritos o Sistema Operacional Android e sua arquitetura, e também o desenvolvimento da aplicação. No capítulo 5 são mostradas as telas da aplicação e suas funcionalidades. Por fim, no capítulo 6 é exposta a conclusão sobre o trabalho e os trabalhos futuros a serem desenvolvidos.

## 2. REFERENCIAL TEÓRICO

### 2.1 Redes Veiculares

As VANETs são formadas por veículos que podem comunicar-se diretamente entre si, através de tecnologias de comunicação sem fio e de protocolos de encaminhamento adequado. O elevado dinamismo desse tipo de rede faz com que essa seja uma área de investigação com enormes desafios tecnológicos. Um dos principais fatores que alavancaram seu desenvolvimento foi à necessidade de aumentar a segurança e eficiência do tráfego por meio de comunicação entre veículos.

Apesar de semelhante a uma MANET, as VANETs possuem características singulares, principalmente relacionadas ao padrão de mobilidade. A primeira, em que os nós permanecem estáveis durante um período relativamente longo, e a segunda, na qual a velocidade em que os automóveis se movimentam é bastante rápida e as ligações entre estes permanecem ativas por um período menor. Por conta dessas características, protocolos desenvolvidos para MANET não são adequados para funcionarem em VANETs (ALASMARY; ZHUANG, 2012).

As VANETs podem ser divididas em três categorias de aplicação: as voltadas para segurança, que será o eixo deste trabalho e, entretenimento e assistência ao motorista. Na categoria de aplicações para segurança, os veículos são adaptados para capturar e compartilhar recursos como *video surveillance*, acidentes de trânsito e aplicações de vídeos para cidades inteligentes. Um vídeo pode fornecer às autoridades de segurança pública informações com mais precisão de que uma mensagem de texto e, dessa forma, permitir a ação mais adequada para cada tipo de situação, reduzindo o tempo de atendimento e/ou resgates (DIMITRAKOPOULOS, 2011).

A FCC (Comissão Federal de Comunicações, do inglês, *Federal Communications Commission*), que é o órgão regulador da área de telecomunicações e radiodifusão dos Estados Unidos, demonstrou interesse nessa área ao alocar 75MHz do espectro de frequência na faixa de 5,9 GHz, para aplicações DSRC (*Dedicated Short-Range Communication*). Uma DSRC nada mais é que um serviço de comunicação para suportar aplicações de segurança pública e

operações privadas, além de fornecer altas taxas de transmissão de dados com baixa latência em um alcance de transmissão de até 1000m (KENNEY, 2011).

O DSRC ainda não é um padrão, embora existam tentativas de padronizá-lo. Além do DSRC americano, temos o DSRC europeu, que utiliza o intervalo de frequência de 5725MHz a 5875MHz, que corresponde à banda de frequência ISM (*Industrial, Scientific and Medical*), e o DSRC japonês denominado ARIB T75, que ainda não está formalizado, mas já está sendo usado para aplicações que usam DSRC, na faixa de 5.8GHz.

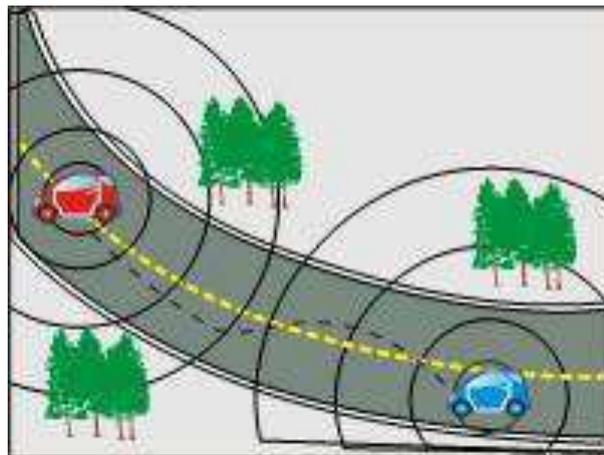
Por outro lado, o IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos, do inglês *Institute of Electrical and Electronics Engineers*) vem desenvolvendo o padrão WAVE (*Wireless Access in a Vehicular Environment*) para a camada física de dispositivos móveis que implementam VANETs. O padrão 802.11a é usado como base para o WAVE (802.11p) (CAMPOLO *et. al.*, 2011).

O WAVE é construído sobre o DSRC americano dividindo o espectro de frequência em sete canais de 10MHz cada (CAMPOLO *et. al.*, 2011). Um desses canais é usado apenas para gerenciamento da rede e serve para troca de mensagens de controle da rede. O grande desafio desse protocolo é a construção de uma camada MAC (*Media Access Control*) que suporte a alta velocidade dos nós, que precisa ter latência minimizada.

### 2.1.1 Comunicação V2X

Para VANETs, podemos citar quatro tipos de comunicação: a primeira na qual, a estrutura de comunicação pode ser realizada de V2V, conforme é apresentado na Figura 1. Na Comunicação V2V, cada OBU (Unidades de Bordo, do inglês *On Board Unit*) funciona em modo *Ad-Hoc*, podendo, assim, encaminhar mensagens através de múltiplos saltos. Nesse tipo de configuração, não é necessário, uma infraestrutura para seu funcionamento, porém, a conectividade da rede é altamente dependente da densidade veicular e do padrão de mobilidade (velocidade, traça da estrada, entre outros).

Figura 1 - Comunicação V2V



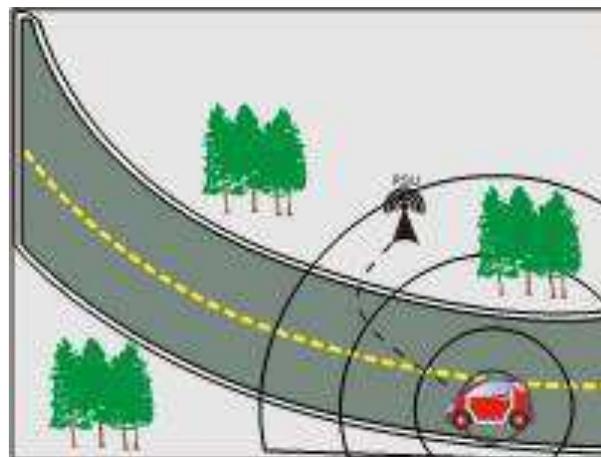
Fonte: Adaptada de Campolo *et. al.*, 2011.

Nesta arquitetura, o tráfego de dados é passado de veículo para veículo, desse modo, os automóveis funcionam como sendo nós da rede. Apesar dessa arquitetura ser implementada de forma mais acessível que as demais, surgem entraves quanto a sua conectividade entre os nós da rede, posto que a sua quantidade se dá aleatória, e seus atributos (movimento, velocidade, entre outros), não são totalmente padronizados.

A segunda, em que a estrutura de comunicação pode ser realizada de Veículo para Infraestrutura (V2I), conforme ilustrado na figura 2. Já na comunicação V2I, a conectividade da rede veicular aumenta e também pode ser integrada, através de gateways, a outras redes.

Ao contrário da arquitetura anterior, a infraestruturada possui nós fixos ao longo da rodovia. Possibilitando, o tráfego em pontos fixos denominados de RSU (Unidade de Acostamento, do inglês *Road Side Units*), reduzindo assim o transtorno da conectividade (GOZÁLVEZ; SEPULCRE; BAUZA, 2012). Mas para que possa utilizar total usabilidade nessa arquitetura, é necessário contar com várias RSUs, consequentemente, elevando o preço de sua instalação.

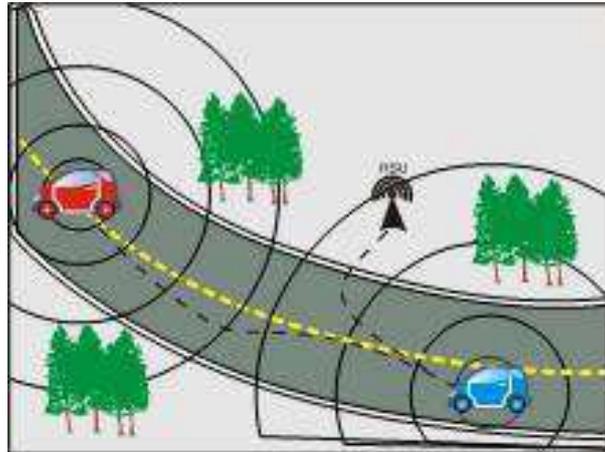
Figura 2 - Comunicação V2I



Fonte: Adaptada de Campolo *et. al.*, 2011

A Terceira, cuja estrutura de comunicação para uma VANET pode ser realizada de Veículo para Veículo para Infraestrutura V2V2I, conforme a figura 3. Tendo em vista sanar os entraves de cada arquitetura uniu-se seus benefícios, atentando para uma arquitetura híbrida, onde é possível que tenha nós fixos e nós estáticos na rede, os veículos e as RSUs.

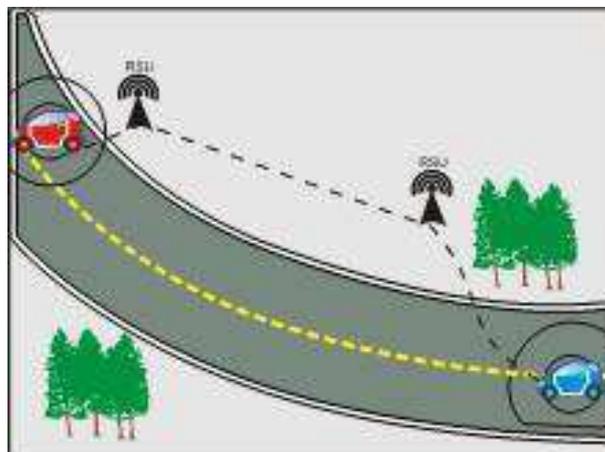
Figura 3 - Comunicação V2V2I



Fonte: Adaptada de Campolo *et. al.*, 2011.

A quarta, em que a estrutura de comunicação para uma rede veicular pode ser implementada é de Veículo para Infraestrutura para Veículo V2I2V, conforme a figura 4.

Figura 4 - Comunicação V2I2V



Fonte: Adaptada de Campolo *et. al.*, 2011.

A comunicação entre os veículos afastados é possível quando as RSUs estão interligadas por fios ou outros links de comunicação (Internet), e cada uma está conectada a um ou mais veículos. Em vista disso, a conexão entre as RSUs é uma ponte para a comunicação entre dispositivos de bordo dos veículos.

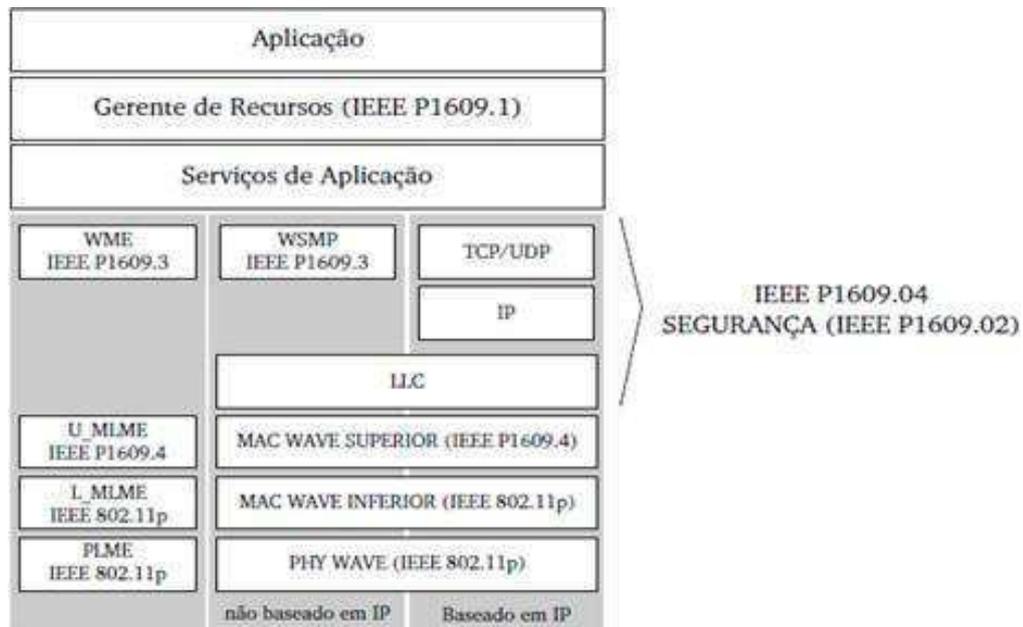
Além das estruturas de comunicação (V2V, V2I, V2V2I, V2I2V), também fazem parte de uma rede veicular, os elementos abaixo:

- Estação-Base: são dispositivos instalados ao lado das vias de tráfego, com o objetivo de fornecer a comunicação entre a VANET e as outras redes (internet). Essas estações podem pertencer a órgãos governamentais ou provedores de serviços com fins comerciais;
- Veículos: são os principais membros das redes veiculares, podendo ser veículos particulares (carros, caminhões, entre outros), públicos (ônibus) ou oficiais (viaturas de polícia, ambulâncias). Esses “veículos inteligentes” podem ser equipados com diferentes componentes que possibilitam a comunicação e a execução dos aplicativos. Alguns deles são descritos abaixo:
  - Sensores: responsáveis pelo monitoramento dos eventos ocorridos com o veículo e ao seu redor;
  - Unidade de Computação (UC): Unidade Central de Processamento do veículo, responsável por transformar os dados obtidos através dos sensores em informações relevantes para os motoristas e para a rede;
  - Unidade de Armazenamento: guarda os dados obtidos pela UC para futuras consultas;
  - Unidade de Comunicação: permite a comunicação sem fio com outros veículos e com as estações-base;
  - Sistema de posicionamento (GPS): ajuda o veículo a informar melhor a sua posição para o resto da rede e a localização de eventos ocorridos nas estradas (acidentes, por exemplo).

### 2.1.2 Arquitetura IEEE 802.11p (WAVE)

Nesta seção, serão abordados os protocolos da arquitetura WAVE que são utilizados como padrão nas comunicações entre as redes veiculares, por intermédio do IEEE 802.11p. Com o objetivo de compreender melhor a estrutura dos protocolos, utilizaremos a figura 5 para descrever as camadas da arquitetura.

Figura 5 - Arquitetura IEEE 802.11p WAVE



Fonte: Adaptada de Alasmay; Zuang, 2012.

O padrão 802.11p para as VANETs definem as propriedades das camadas físicas e controle de acesso ao meio (ALASMARY, ZHUANG, 2012). O IEEE 1609 foi designado para disponibilizar recursos para uma grande quantidade de aplicações no contexto veicular, incluindo aplicações de segurança (KARAGIANNIS *et. al.*, 2011). No padrão IEEE 1609, cada documento trata das características específicas das VANETs. O IEEE 1609 (1609.1, 1609.2, 1609.3, 1609.4) define a arquitetura de comunicação complementar para padronizar vários serviços e interfaces que coletivamente permitem, de forma segura, as comunicações sem fio do tipo V2V e V2I. O IEEE 1609.1 (Gerência de Recursos) descreve como serão os serviços de dados e define os serviços de aplicação dentro da arquitetura WAVE. No padrão WAVE, existem duas unidades importantes, as RSUs que guardam aplicações e provêem serviços, e as OBUs que executam aplicações que utilizam um serviço específico.

Em uma RSU, é executada uma aplicação de RM (Gerente de Recursos, do inglês *Resource Manager*) que se comunica com um RCP (Processador de Comandos de Recursos, do inglês *Resource Command Processor*) situado em uma OBU. O RM tem como função multiplexar pedidos vindos de um dispositivo de acostamento, que podem se comunicar com as RCPs.

O IEEE 1609.2 (Gerente de Segurança) define os formatos de informações seguras e seu processamento. Define os momentos que devem ser utilizadas essas informações e de que modo de ser feito o processamento delas conforme seu propósito.

É estabelecido um subconjunto das OBUs, utilizados em veículos de segurança pública, chamados de OBUs de Segurança Pública (PSOBUs). Esse conjunto é implementado para gestão de sinais de trânsito e outras atividades semelhantes. Similar ao padrão anterior, o IEEE 1609.2 define um gerente de segurança que atua no gerenciamento do certificado principal (raiz) e no armazenamento das listas de certificados que foram revogados por uma autoridade de certificação.

O IEEE 1609.3 define as atividades das camadas de rede e de transporte, abrangendo roteamento e endereçamento. Para dar suporte segura à troca de dados da arquitetura WAVE é determinado um plano de dados e um de gerenciamento.

- Plano de dados: especifica quais os protocolos responsáveis pela transmissão de dados resultantes das aplicações WAVE e também pelo tráfego que as entidades do plano de gerenciamento geram na rede.
- Plano de gerenciamento: configura e faz a manutenção do sistema. Implementa serviços (por exemplo: registro de aplicações).

O IEEE 1609.4 fornece aprimoramentos para o IEEE 802.11 para apoiar as operações WAVE.

### 2.1.3 Aplicações

As redes veiculares podem ser distribuídas em três categorias: segurança, entretenimento e assistência ao motorista. Em segurança, as aplicações têm por objetivo divulgar de maneira rápida as informações, sendo elas preventiva ou emergencial, dessa forma, o condutor que recebe essas informações tenha tempo de reagir. As aplicações de respeito ao entretenimento, contam com recursos adaptativos para aplicação de internet para as VANETs. Finalmente, as aplicações que envolvem o recebimento de dados para que possa utilizar como auxílio para buscas e automatização de serviços, são as de assistência ao motorista.

### 2.1.3.1 Segurança

Um dos principais incentivos nas pesquisas e desenvolvimento das redes VANETs, é a expectativa no aumento de segurança no trânsito (CHU; HUANG, 2012). Geralmente, as aplicações voltadas para a classe de segurança têm por finalidade diminuir os casos de acidentes, por meio de trocas de dados entre os veículos. Dados esses que são mostrados aos motoristas ou são utilizados para acionar sistemas de segurança. As aplicações de segurança determinam requisitos restritos de confiabilidade e latência, requerendo propriedades diferenciadas dos protocolos das camadas inferiores.

Neste contexto, conteúdo multimídia fornece informações mais precisas do que uma simples mensagem de texto, permitindo que especialistas ou software específicos de segurança pública possam verificar visualmente o impacto real do evento e, desse modo, evitar alarmes falsos, tomar consciência do que está acontecendo no ambiente, planejar ações, detectar objetos e analisar, de forma mais segura, as informações recebidas.

### 2.1.3.2 Entretenimento

Informações sobre restaurantes locais, postos de gasolina e demais pontos de interesse podem ser carregadas no veículo por uma RSU e compartilhadas com os demais veículos em uma rede *Ad-Hoc* (TAL; MUNTEAN, 2012). O mesmo pode ser feito com arquivos multimídias (músicas, filmes, notícias, entre outros).

Um anúncio é transmitido da mesma maneira, e sua exibição ainda pode ser condicionada às informações coletadas do veículo. Por exemplo, postos de gasolinas podem ser sugeridos a veículos com pouco combustível, oficinas mecânicas a veículos com problemas diagnosticados eletronicamente. Grande parte das aplicações de entretenimento desenvolvidas para as VANETs, no entanto, estão associadas com a necessidade de internet. Por esse motivo, é fundamental adaptar os recursos mais utilizados na internet para as propriedades das VANETs.

Várias aplicações para as VANETs sustentam a utilização da arquitetura *Ad-Hoc*, por não dispor de dispositivos centralizadores e a comunicação entre os veículos não necessita de pontos de acessos. Além de que, na comunicação V2I, o custo da instalação e manutenção da rede é alto. Diante disso, as aplicações para

entretenimento, optam a utilização de sistemas ponto-a-ponto ao modelo cliente-servidor, que é de forma centralizada.

### 2.1.3.3 Assistência ao Motorista

As aplicações de assistência aos motoristas têm como objetivo auxiliar o motorista na condução do veículo com base no monitoramento e na disponibilidade de informações (AISSAOUI *et. al.*, 2014). Essas informações são adquiridas a partir de serviços que podem ser oferecidos ao condutor em momentos oportunos ou podem ser de fácil acesso através de procedimentos de busca. Dentre alguns exemplos, podem-se citar: aviso de estacionamentos, disseminação de informações de vias, controle de tráfego, condução conjunta de veículos, localização em mapas, aumento da visibilidade e veículos sem condutor humano. Dentre as aplicações desta classe, as aplicações de indicação de vagas de estacionamentos vêm recebendo bastante atenção. Um dos argumentos utilizados é que, além de conveniente, ela pode reduzir os problemas de congestionamento nas cidades.

Outra aplicação importante de assistência ao motorista é a disseminação de informações sobre as condições das vias. Essas aplicações buscam reduzir o tempo de espera dos motoristas em congestionamentos, apresentando rotas alternativas que evitam áreas com tráfego lento. Essas aplicações possuem ainda o efeito indireto de redução da poluição ambiental e podem ser utilizadas para evitar áreas de risco.

## 2.2 Redes LTE 4G

A tecnologia móvel, também conhecida quarta geração (4G), irá desempenhar um papel central para o futuro das comunicações nos cenários das redes móveis (CHEN; PANDE; MOHAPATRA 2014). O 3GPP (*3rd Generation Partnership Project*) é o órgão responsável pela padronização e especificação do LTE (SHEN *et. al.*, 2012).

O objetivo do LTE é apresentar uma arquitetura evoluída baseada totalmente em IP (*Internet Protocol*), onde na SAE (*System Architecture Evolution*), que se trata de uma evolução das redes GPRS (*General Packet Radio Services*) com algumas

evoluções caracterizadas como: arquitetura simplificada, conceito de redes All-IP, suporte a redes de acesso via rádio e suporte à mobilidade entre redes de acesso (HOLMA; TOSKALA, 2011).

Nos últimos anos, as organizações têm definido as arquiteturas de rede e funções para suportar as características únicas de comunicações. As redes de banda larga móvel mostram algumas diferenças importantes em relação aos seus equivalentes com fio, criando a necessidade de administrar o tráfego.

### 2.2.1 Arquitetura

O LTE foi proposto pelo 3GPP e apresentado nas releases 8 e 9, tratando-se de um sistema totalmente novo de comunicações móveis (DAHLMAN; PARKVALL; SKOLD, 2013).

A release 8 do 3GPP foi marcada pela introdução de uma nova rede de *core*, a SAE. A SAE também é denominada EPC (*Evolved Packet Core*) que, operando em conjunto, o LTE/SAE, abrange o EPS (*Evolved Packet System*) (DAHLMAN; PARKVALL; SKOLD, 2013).

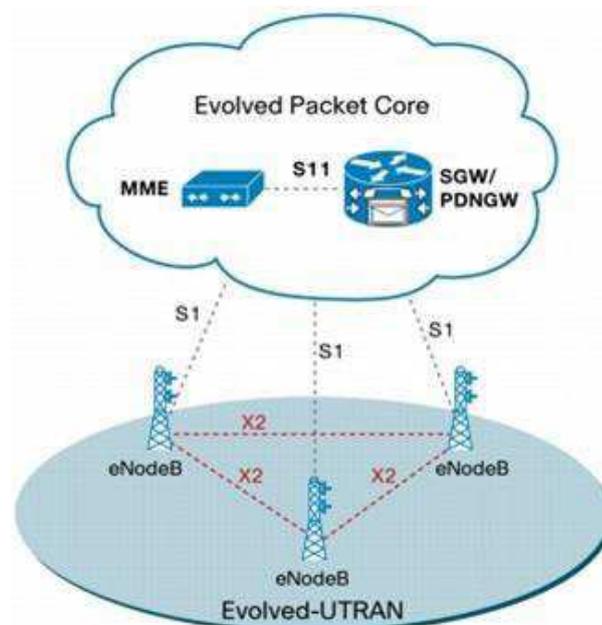
O E-UTRAN (*Evolved-Terrestrial Radio Access Network*) é uma interface de rádio inovadora e evoluída, baseada na tecnologia OFDMA (*Orthogonal Frequency Division Multiple Access*), que associa diferentes sub-canais para usuários diferentes, evitando problemas causados por reflexões multipercursos, enviando *bits* de um dado a baixas velocidades, combinados no receptor para formar uma mensagem de alta velocidade. O SAE é uma rede All-IP desenvolvida para suportar o E-UTRAN; ou seja, a rede básica e o acesso rádio são completamente comutadas por pacotes (DAHLMAN; PARKVALL; SKOLD, 2013).

A release 9 foi divulgada no final do ano de 2009 e definiu, de forma mais precisa, algumas integrações do LTE e aprimorou ainda mais alguns componentes das redes 3G, promovendo melhorias nas taxas de transmissão de dados em dispositivos baseados em HSPA+ (*High Speed Packet Access*). Assim, foi obtida a interoperabilidade entre as tecnologias, não havendo a necessidade de migrar do UMTS (*Universal Mobile Telecommunication System*) (3G/3.5G) para o LTE (3.9G/4G) por algum tempo.

A arquitetura é composta principalmente por dois componentes, E-UTRAN e EPC, como é apresentada na figura 6.

O E-UTRAN consiste apenas em um elemento, o eNodeB. O EPC é composto pelo MME (*Mobility Management Entity*), o S-GW (*Serving Gateway*) e o P-GW (*Packet Data Network Gateway*), sendo SAE-GW (*System Architecture Evolution Gateway*) (COX, 2012).

Figura 6 - Arquitetura de Acesso das Redes LTE



Fonte: Cox, 2012.

### 2.2.2 Node B

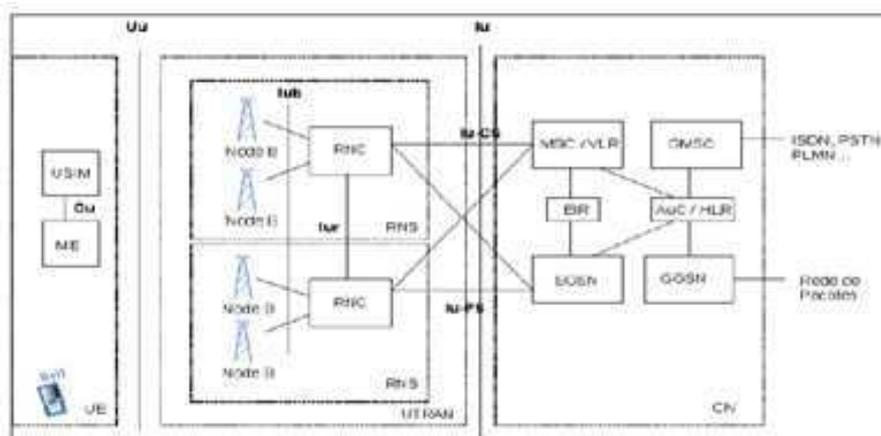
A Node B é a unidade física de transmissão e recepção de rádio. Ela tem a responsabilidade pela conversão de dados, enviados pela RNC (*Radio Network Controller*), em sinais eletromagnéticos e o envio destes sinais para o meio de transmissão. Outra funcionalidade da Node B é a gestão da potência dos terminais móveis e a captura de dados para o processo de gerenciamento da rede, tais como qualidade e robustez da conexão, que está relacionada à taxa de erro de bit BER

(*Bit Error Rate*), e à taxa de erro de quadro FER (*Frame Error Rate*). Os envios dessas informações para a RNC estão em um relatório de medidas para que se possa tomar decisões como handover, por exemplo (REZAEI; HEMPEL; SHARIF, 2015).

Do ponto de vista da rede de rádio, cada Node B é formada por diversas unidades lógicas denominadas células. Uma célula é a menor entidade da rede e possui um número de identificação próprio, chamado CI (*Cell ID*), que constitui um dos métodos de rádio-localização baseado na identidade da célula, porém, esse método possui baixa precisão comparada a outros métodos mais complexos. Toda célula está associada a um scrambling code que, em conjunto com o CI, permite que o UE reconheça aquela célula (REZAEI; HEMPEL; SHARIF, 2015).

Como ilustra a figura 7, a Node B está localizada entre as interfaces UU e IUB. Ela é conectada ao UE através da interface UU, que utiliza o WCDMA como tecnologia de acesso a rádio, enquanto a comunicação com a RNC é feita via interface IUB (SESIA; BAKER; TOUFIK, 2011).

Figura 7 - Elementos da NodeB



Fonte: Rezaei; Hempel; Sharif, 2015.

Embora as principais funções da Node B estejam relacionadas com a transmissão e recepção do sinal de rádio, existem também outras funções como: geração de código e o controle de potência (SESIA; BAKER; TOUFIK, 2011).

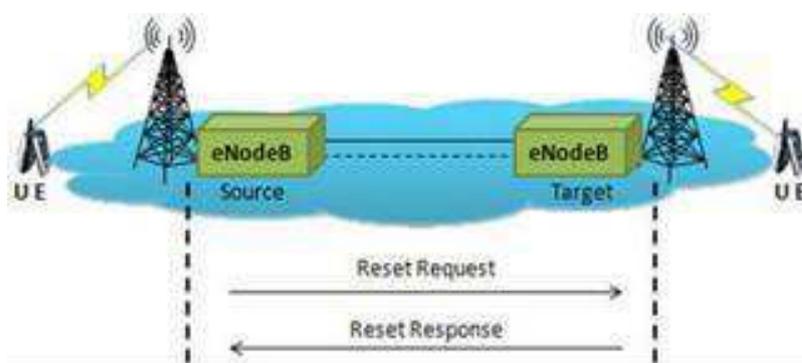
### 2.2.3 Interface X2

As NodeB cooperam entre si para fornecer determinados serviços aos assinantes. Quando estamos trafegando por uma rodovia, iremos passar pela área de cobertura da estação base atual rapidamente, devido à velocidade do veículo. Para não perder a conexão, às NodeB cooperam entre si e com o resto da rede celular, para encontrar automaticamente a próxima estação de base que poderá garantir a conexão.

Na rede LTE, as NodeB também possuem inteligência e com isso não necessitam de comunicação constante com o núcleo da rede LTE para algumas funções. Esse método é muito mais eficiente e favorável ao assinante, pois reduz consideravelmente o fluxo de tráfego na rede e latência, consequentemente. Com o protocolo X2 e o seu funcionamento entre as NodeBs, a entrega de informações leva menos tempo e exige menos tráfego de rede (PACHECO-PARAMO; AKYILDIZ; CASARES-GINER, 2013).

Os benefícios da interface X2 não estão limitados apenas à inteligência, durante a comunicação e cooperação é possível executar o gerenciamento de carga entre elas. Elas podem relatar status e erros, e executar outras funções de cooperação, como ilustra a figura 8 (PACHECO-PARAMO; AKYILDIZ; CASARES-GINER, 2013).

Figura 8 - Cooperação entre NodeB



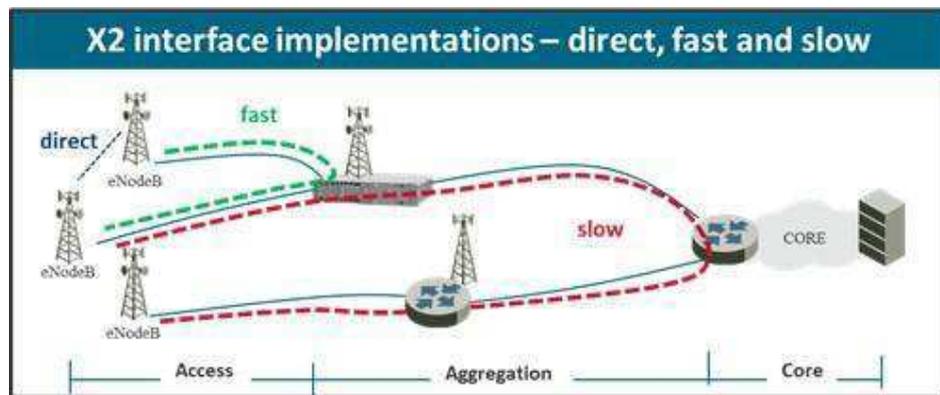
Fonte: Pacheco-Paramo; Akyildiz; Casares-Giner (2013)

Existem três métodos para a implementação da interface X2. O primeiro é chamado de método direto, onde uma NodeB pode se comunicar diretamente com

outra NodeB, por fibra, sem fio ou banda licenciada (PACHECO-PARAMO; AKYILDIZ; CASARES-GINER, 2013).

O método direto é o melhor, considerando custos e distâncias alcançadas, uma vez que é o menos sujeito a latência. Muitas operadoras de telefonia móvel, até agora, estão evitando a implementação das ligações diretas entre NodeB, como ilustra a figura 9.

Figura 9 - Interface X2



Fonte: Pacheco-Paramo; Akyildiz; Casares-Giner, 2013.

Há duas outras maneiras de implementar X2. Em uma implementação mais rápida para aplicação X2, as NodeBs podem se comunicar umas com as outras através do seu RNC, que atua como um *router* para essa finalidade. Isso reduz o custo de implantação consideravelmente, mas introduz mais latência no arranjo (OUELLETTE; MARCHAND; PIERRE, 2011).

Em uma implementação mais lenta, o serviço de mensagens X2 pode viajar por todo o caminho de volta para um ponto de agregação ou até mesmo o núcleo. Implementação X2 lenta é relativamente simples, uma vez que esta é semelhante à maneira como ele funciona em redes 3G. No entanto, como o próprio nome indica, a implementação lenta é preocupante com a mais alta latência (OUELLETTE; MARCHAND; PIERRE, 2011).

## 2.3 Transmissões de Vídeos

Atualmente, existem diversas aplicações para VANETs multimídia e, para cada uma dessas aplicações, existem alguns requisitos a ser seguidos para obter uma qualidade de serviço aceitável para o usuário final. Algumas aplicações são altamente sensíveis a atraso, se esse atraso passa de certo limite de tolerância, os pacotes atrasados já são inúteis ao receptor. Por outro lado, algumas aplicações são mais tolerantes a perdas de pacotes, pois só causam pequenas perturbações na recepção do vídeo (PHAM; PIAMRAT, VIHO, 2014).

As aplicações para transmissão de vídeo ao vivo são análogas a uma transmissão de televisão, só que nesse caso a transmissão é realizada a partir de um veículo por uma VANET. Atrasos de até dezenas de segundos, desde a requisição de um vídeo até o início da reprodução, podem ser tolerados, considerando o tipo de vídeo que está sendo transmitido ou compartilhado (FELICE *et. al.*, 2014).

### 2.3.1 Protocolo de Transporte

Como explicado anteriormente, no serviço de transmissão de vídeo, o cliente solicita arquivos de vídeos comprimidos que residem em servidores. Como resposta a essa requisição, o servidor envia ao cliente o arquivo de vídeo por uma porta. Essa porta pode corresponder tanto a protocolos orientados à conexão ou a protocolos não orientados à conexão. Usaremos como referência o TCP (*Transmission Control Protocol*) e UDP (*User Datagram Protocol*), respectivamente, por serem os mais utilizados (KUROSE; ROSS, 2012).

O protocolo TCP é orientado à conexão assegurando que toda a informação chegue ao ponto destino, na sequência correta. Por esse motivo, a maior parte dos serviços de transmissão de vídeo utiliza esse protocolo, já que os quadros de vídeos são apresentados na sua totalidade (KUROSE; ROSS, 2012). Por outro lado, o protocolo UDP é um protocolo não orientado à conexão e, conseqüentemente, não garante a entrega da informação no destino. Por tal motivo, há uma falta de informação nos quadros da sequência de vídeo (KUROSE; ROSS, 2012).

O uso de um determinado protocolo de transporte para o envio do vídeo nesse tipo de serviço origina diferentes degradações do vídeo no receptor, quando a rede apresenta algum problema, por exemplo, de capacidade (largura de banda) ou perda de pacotes. No caso do TCP, a degradação principal no receptor é o congelamento de uma imagem por um período de tempo (pausa), o usuário não recebe novos quadros do vídeo durante o período da pausa. Para o caso do protocolo UDP, os problemas da rede se manifestam como um tipo de ruído na sequência do vídeo, mas essas são apresentadas continuamente ao cliente (KUROSE; ROSS, 2012).

Assim, pode-se dizer que a transmissão de vídeo via UDP está mais relacionada às degradações espaciais, ou seja, existe uma degradação na imagem (quadros); e à transmissão que usa protocolo TCP é mais afetada por degradação no domínio do tempo (pausas) (KUROSE; ROSS, 2012). A figura 10 ilustra esses dois tipos de degradações.

Figura 10 - Degradações do Vídeo

(A) Protocolo TCP (B) Protocolo UDP em um cenário com perdas de pacotes.



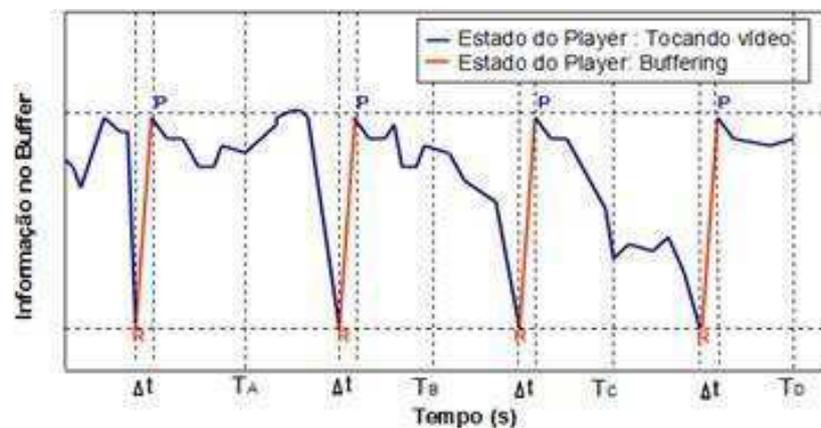
Fonte: Kurose; Ross, 2012.

No caso do protocolo TCP, o *buffer* do transdutor pode passar por diferentes estados, dependendo da relação da taxa da rede e a taxa com o qual o vídeo é apresentado ao usuário. Em seguida é descrito o comportamento do *buffer* e a relação com as pausas.

As pausas são originadas por diversos problemas na rede IP, como: capacidade de largura de banda, perdas de pacotes, atrasos, congestionamento nos

elementos da rede. Quando o TCP detecta perdas de pacotes, o algoritmo de congestionamento do TCP, reduz o tamanho da janela e, conseqüentemente, o TCP diminui a taxa de transferência e se essa nova taxa é menor do que a taxa de reprodução do *player* será utilizada toda a informação do *buffer* e depois disso ele irá entrar em um processo de *rebuffering*. A figura 11 apresenta os diferentes estados do transdutor de vídeo (*player*), os quais são descritos em seguida.

Figura 11 - Estados de um *buffer*



Fonte: Song *et. al.*, 2012.

O *buffer* inicial é o período de tempo para armazenamento mínimo da informação a ser mostrada no início do vídeo. Esse parâmetro é configurável, assim, para redes com boa taxa de transmissão, podendo ser períodos curtos, por exemplo, dois segundos. Para redes baixa transmissão, como as VANETs, devem ser tempos maiores, como de cinco segundos (SONG *et. al.*, 2012). Período de reprodução do vídeo, para algumas aplicações, deve ser apresentado de forma contínua, sem interrupções e sem perda de informação. O período de *rebuffering* é o momento no qual o *buffer* não tem um mínimo de informação para ser apresentado.

### 2.3.2 Degradação do QoS

Otimização de QoS são muito importantes para as aplicações de emergência para as VANETs multimídia. As VANETs requerem transmissão de mensagens em tempo real, e que seja capaz de fornecer dados em tempo hábil e preciso para um socorro imediato. Por exemplo, considerando-se o caso de aplicações de segurança

no trânsito, qualquer atraso na entrega da mensagem pode ocasionar grandes acidentes. Da mesma forma, o envio de arquivos multimídia e fluxo de vídeo requerem um alto nível de QoS (JOTHI; JEYAKUMAR, 2013). Os principais fatores de degradação de qualidade dos serviços nas VANETs multimídia estão associadas a fatores, como:

- Taxa de perda de pacotes: está representada pelo número de pacotes que foram transmitidos pela rede, mas não alcançaram seu destino em um determinado tempo.
- Atraso ponto a ponto: é definido pelo tempo que uma mensagem demora em chegar desde o ponto de origem até o ponto destino, esse atraso é devido a diversos fatores como: algoritmo de codificação, o meio de transmissão e o atraso de filas em roteadores.
- Variação de atrasos: chamado de *jitter*, é definido por (ITU-T 1540), como o intervalo entre a chegada de dois pacotes consecutivos em relação ao intervalo de sua transmissão. Ele é baseado na medição do atraso fim a fim e é definido para pares consecutivos de pacotes

Em uma rede de transmissão de pacotes é necessário programar algoritmos de controle para diminuir a variação de atrasos e, conseqüentemente, melhorar o nível de satisfação do usuário final (JAGADISH; MANIVASAKAN, 2011).

A transmissão de vídeo é crucialmente necessária e valiosa para um alto nível de satisfação dos usuários para a maioria das aplicações e serviços para VANETs. Dispositivos computacionais e telas de exibição disponíveis dentro de veículos não sofrem as mesmas restrições impostas aos dispositivos usuais para MANETs (QUAN *et. al.*, 2014). Portanto, eles são capazes de reproduzir vídeos de alta qualidade.

A transmissão do vídeo ao longo das rodovias a partir de uma VANETs apresenta grandes desafios (QUAN *et. al.*, 2014). A transmissão do conteúdo do vídeo exige preocupações com requisitos rigorosos de atraso e taxa de entrega. Além disso, o *jitter* e o controle de buffer tem de ser considerados quanto ao seu impacto sobre QoE dos usuários.

O vídeo está desempenhando um papel cada vez mais importante no futuro da comunicação. Para VANETs é um meio viável para compartilhar informações de segurança, entretenimento, entre outras, com sua alta capacidade, coerência e influência sobre os seres humanos (QUAN, *et. al.*, 2014). A transmissão

de vídeos em tempo real para VANETs ainda é um desafio devido à sua natureza dinâmica. Esse tipo de aplicação exige muito em termos de largura de banda e atraso.

As VANETs irão desempenhar um papel importante no cenário das cidades inteligentes e apoiarão o desenvolvimento, não apenas de aplicações de segurança, mas também de serviços de transportes inteligentes. As recentes melhorias em VANETs multimídia irão permitir aos condutores, passageiros e equipes de resgate, capturar e compartilhar vídeos. Os veículos podem cooperar entre si para transmitir ao vivo os fluxos de acidentes ou desastres de trânsito e proporcionar aos condutores informações visuais sobre uma área monitorada (QUADROS *et. al.*, 2014).

### 2.3.3 Qualidade de Experiência (QoE)

As tradicionais técnicas que visam maximizar o nível de qualidade de aplicações multimídias estão focadas apenas no desempenho da rede ou em métricas de QoS. Porém, os parâmetros de QoS falham na captura de aspectos subjetivos associados com a percepção humana, no controle de rede e nas operações de otimização (SERRAL-GARCIA *et. al.*, 2010) (MU *et. al.*, 2009). Em seguida, na figura 12 são apresentados os principais serviços de vídeo, indicando-se os possíveis problemas que causam as degradações, os parâmetros que podem ser medidos e como o QoE do usuário é afetado.

Figura 12 - Degradação do Vídeo com o QoE

Serviços	Problemas na rede	Parâmetros mesuráveis	Degradação e QoE
Streaming de Vídeo (protocolo não orientado a conexão; exemplo UDP)	Congestão na rede (falta de capacidade), sobrecarga nos roteadores, perda no canal de transmissão.	Taxa de transmissão, perda de pacotes, atraso fim a fim, e variação de atraso.	Degradação espacial (perda de informação nos quadros)
Streaming de Vídeo (protocolo orientado a conexão; exemplo TCP)	Congestão na rede (falta de capacidade), sobrecarga nos roteadores, perda no canal de transmissão.	Pausa, número de pausas, localização temporal das pausas. (No transdutor do dispositivo do usuário).	Degradação temporal (congelamento de imagem)
IPTV (protocolo não orientado a conexão; exemplo UDP)	Congestão na rede (falta de capacidade), sobrecarga nos roteadores, perda no canal de transmissão.	Taxa de transmissão, perda de pacotes, atraso fim a fim, e variação de atraso.	Degradação espacial (perda de informação nos quadros), chaveamento do canal.
Vídeo Conferência (protocolo não orientado a conexão; exemplo UDP)	Congestão na rede (falta de capacidade), sobrecarga nos roteadores, perda no canal de transmissão.	Taxa de transmissão, perda de pacotes, atraso fim a fim, e variação de atraso	Degradação espacial (perda de informação nos quadros); latência

Fonte: Song *et. al.*, 2012.

Portanto, QoE ajuda a avaliar as aplicações multimídia sob o ponto de vista da percepção do usuário. As métricas podem ser classificadas como objetivas ou subjetivas, tendo como retorno um valor quantitativo, que é mapeado para uma faixa de valores qualitativos. As métricas objetivas possuem como característica a realização da estimativa da qualidade do vídeo recebido pelo usuário através de modelos matemáticos, tendo seus valores mapeados para valores subjetivos. Já as subjetivas são caracterizadas como métricas que necessitam que o conteúdo multimídia seja analisado por um grupo de pessoas, e os resultados dessa análise são tratados estatisticamente para antecipar as percepções dos usuários.

Várias métricas objetivas foram desenvolvidas com a proposta de estimar/prever o nível da qualidade de serviços multimídia, conforme a percepção do usuário. As principais métricas objetivas são descritas a seguir:

### 2.3.3.1 Structural Similarity Index (SSIM)

O SSIM é uma métrica para medir a similaridade entre duas imagens. O índice SSIM pode ser visto como uma medalha de qualidade de uma das imagens que estão sendo comparadas, desde que a outra imagem seja perfeita.

O SSIM compara três parâmetros do vídeo original e o vídeo degradado, são eles: luminosidade, contraste e estrutura dos objetos apresentados em cada frame do vídeo. Denominado o vídeo original de  $x$  e o vídeo degradado de  $y$ , são obtidas a média de cada vetor e o indicativo da qualidade do vídeo é gerado pela combinação das três médias. Segundo Gao *et. al.* (2009), o valor de SSIM é obtido através da Equação 1.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\delta_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\delta_x^2 + \delta_y^2 + c_2)} \quad (1)$$

Onde:

- $\mu_x$  é a média de  $x$ ;
- $\mu_y$  é a média de  $y$ ;
- $\delta_x$  é o desvio padrão de  $x$ ;
- $\delta_y$  é o desvio padrão de  $y$ ;
- $\mu_{xy}$  é a covariância de  $y$ ;
- $c_1 = (K_1L)^2$ ,  $c_2 = (K_2L)^2$  são duas constantes;
- $L$  = o valor máximo que pode ser atribuído a cada *pixel*;
- $K_1 = 0,01$  e  $K_2 = 0,03$  por padrão;

O valor de SSIM é representado como um valor decimal entre 0 e 1. Quanto mais próximo de 0, pior é a qualidade do vídeo. Quanto mais próximo de 1, melhor a qualidade do vídeo. No caso em que o valor seja igual 1, o vídeo recebido pelo usuário é equivalente em 100% do vídeo original (KIM; KIM, 2013).

### 2.3.3.2 Video Quality Metric (VQM)

A métrica de VQM foi desenvolvida pelo The Institute for Telecommunication Science para fornecer uma avaliação objetiva da qualidade do vídeo. Ela mede os efeitos da percepção das deficiências do vídeo, incluindo desfoque, movimentação irregular e artificial, ruído global e distorção de cor, para posteriormente os combinar em uma simples métrica.

A métrica é realizada em quatro etapas onde, na primeira, denominada de “calibração”, ocorre a comparação do contraste e brilho entre o vídeo original e o recebido. Posteriormente, ocorre a comparação do alinhamento espacial, onde se verifica a posição no eixo horizontal e vertical de ambos os vídeos. Já a terceira etapa é caracterizada pela realização da análise da quantidade de pixels perdidos em relação ao vídeo original. Finalizando, essa etapa se ao alinhamento temporal, comparando o tempo no qual cada frame foi recebido, em relação ao tempo no qual o mesmo deveria ter sido recebido. O VQM é dado por um número real entre 0 e 5. Quanto mais próximo o valor for de 0, melhor será a qualidade do vídeo, indicando uma menor distorção em relação ao vídeo original (MOORTHY *et. al.*, 2012).

### 3 TRABALHOS RELACIONADOS

Este capítulo tem como objetivo apresentar o levantamento do estado da arte em relação ao roteamento colaborativo, o qual tem o potencial de ser utilizado no cenário de segurança pública apresentado no Capítulo 1. Esse capítulo divide-se em duas seções; na seção 3.1, serão apresentadas algumas técnicas de agrupamentos em clusters (VODOPIVEC; BESTER; KOS, 2012) que define a colaboração utilizando cluster. Na seção 3.2, serão apresentados alguns protocolos de roteamento baseados em clusters projetados para VANETs (SHAREF; ALSAQOUR; ISMAIL, 2014), que define o roteamento colaborativo.

Para ambos os casos serão apresentadas as principais vantagens e desvantagens, incluindo os problemas relacionados à escalabilidade, confiabilidade, distribuição justa dos recursos e a manutenção no nível de qualidade mínima para entrega de vídeo.

#### 3.1 Técnica de Formação de *Cluster*

##### 3.1.1 Multiagent Driven Dynamic Clustering of Vehicles in Vanets (MDDC)

Segundo Kakkasageri; Manvi (2012), a formação de *cluster* em redes veiculares *Ad-Hoc* é um problema desafiador, devido às constantes mudanças da topologia da rede e quedas frequentes dos enlaces de comunicação V2V. As técnicas para formação de *cluster* em cenários de rodovias facilitam o agrupamento de veículos. Neste trabalho, os autores Kakkasageri; Manvi (2012) apresentaram um software agente que impulsiona o esquema de agrupamento dinâmico para VANETS. O software agente é composto por agentes estáticos *heavy-weight* (peso-pesado) e agentes móveis *light-weight* (peso-leve).

Inicialmente, os membros do *cluster* são identificados com base na velocidade relativa do veículo e na direção para agrupamento dinâmico. O CH (*Cluster-heads*) é selecionado entre os membros do *cluster*, com base na métrica de estabilidade derivada do grau de conectividade, velocidade média e tempo para deixar a

interseção da estrada. O CH prevê futura associação de membros do cluster com base em padrões de mobilidade. O anúncio do padrão de mobilidade do cluster para todos os membros do *cluster* é feito pelo CH. Os membros do *cluster* com semelhante padrão de mobilidade podem se reconectar com o CH depois de passar em um cruzamento da pista.

Consideramos uma VANET em que um número de veículos é separado por certa distância entre si, ou seja, são baseados em arquitetura de V2V, os autores assumem que os veículos se deslocam no mesmo sentido, em um cenário urbano.

Nesse ambiente de rede, todos os veículos são equipados com GPS, onde o INS (Sistema de Navegação Inercial do inglês, *Inertial Navigation System*), permite obter um plano de referência estabilizado e facilita o cálculo das coordenadas em VANETs transceiver/router (OBUv). Nesse cenário, cada veículo é carregado com um mapa digital do local e está preocupado com a informação de estrada à sua frente.

Cada veículo tem informações do seu comprimento em relação à pista e se comunica com outro veículo dentro de sua área de comunicação, onde um segmento de pista termina em um cruzamento, dando início à formação do cluster. Ainda segundo Kakkasageri; Manvi (2012), o fabricante do equipamento de borda fornece uma plataforma (agente) que apóiam o agrupamento dos veículos durante a formação do *cluster*.

Essa proposta depende exclusivamente do *software* do fabricante de OBU, o que compromete a escalabilidade, confiabilidade e distribuição justa dos recursos considerando que, em locais que não se tenha o mapa com as informações, compromete-se o sistema como um todo.

### 3.1.2 Engineering a Cooperative Content Distribution Framework for Highway Vehicular Communications (CHAINCLUSTER)

Segundo Zhou *et. al.* (2014), os recentes avanços em técnicas de comunicação sem fio tornaram possível para os veículos, baixar dados a partir de uma infraestrutura de comunicações de beira de estrada, através da comunicação V2I, ou seja, um *drive-thru* de serviço de Internet de banda larga, o que vem reforçar a necessidade da comunicação V2I, no cenário para VANETs . No entanto, devido à alta mobilidade, os canais sem fio intermitente e o volume de *download* de dados de

veículo individual pelo *drive-thru* são bastante limitados. O que restringe severamente a qualidade das aplicações multimídia. Para resolver esse problema, os autores propõem uma cadeia de *cluster* (*ChainCluster*).

O *ChainCluster* seleciona veículos para formar um *cluster* na rodovia. Os membros do *cluster*, em seguida, realizam uma cooperação para realização do *download* do mesmo arquivo (mesmo conteúdo), cada membro recuperar uma parte do arquivo a partir da infraestrutura de comunicação de beira de estrada. O *download* de um conteúdo que antes era realizado individualmente, agora é estendido a um conjunto de veículos, o que, conseqüentemente aumenta a probabilidade de sucesso de *download* de arquivos de forma significativa. Um RSU está instalado na beira da estrada, que serve como porta de entrada para fornecer serviços de Internet para a condução de veículos por meio de sua cobertura.

Nesse modelo proposto pelos autores, um dos veículos do *cluster* é identificado, ou seja, que assina o *download* do arquivo a partir do RSU. O veículo com etiquetas e outros veículos não identificados podem cooperar para realizarem em conjunto o *download* e distribuição do conteúdo.

O *cluster* aparece como uma topologia linear nas rodovias, e cada membro do *cluster* são responsáveis por baixar e enviar uma parte separada do arquivo para o veículo com etiquetas, a cooperação se estende praticamente o tempo do *download*, onde ocorrem alguns ajustes, dependendo do número de membros no *cluster*.

A cooperação consiste em três fases: fase de formação do *cluster*, fase de *download* de conteúdo e a fase de encaminhamento de conteúdo. Especificamente, se o veículo com etiqueta deseja baixar um conteúdo de grande capacidade, ele convida outros veículos para formar um *cluster*.

Uma vez que o veículo com a etiqueta entrar na cobertura do RSU, esse veículo pode avaliar a atual taxa de transferência de *download*. Vários veículos são escolhidos para formar um aglomerado em uma cadeia linear cooperativa, chamada de conjunto de formação de fases.

Dentro da cobertura de RSU, cada veículo no *cluster* cooperativo transfere uma parte sem sobreposição do arquivo sequencialmente, que é a fase de *download* do conteúdo. Do lado de fora da cobertura do RSU, o veículo com etiquetas recolhe as partes do arquivo baixado pelos membros do *cluster*, para recuperar o arquivo e completar o processo de transferência final.

Essa proposta também apresenta problemas que comprometem a escalabilidade, confiabilidade e distribuição justa dos recursos, considerando que um veículo com etiqueta, fazendo um *download*, sai do *cluster*, o sistema, é comprometido.

## 3.2 Protocolos de Roteamento Baseados em *Cluster*

### 3.2.1 A Passive Clustering Aided Routing Protocol for Vehicular Ad-Hoc Networks (PASSCAR)

Segundo Wang; Lin (2013), o protocolo de roteamento baseado em *clusters* denominado de (Pass-CAR), tem como objetivo determinar participantes para a estrutura de *clusters* com características estáveis (mesma direção) e confiáveis, durante o processo de descoberta de rotas, para melhorar o desempenho do roteamento em cenários de rodovias com mão única e com várias pistas.

Como objetivos específicos da proposta, cada veículo candidato determina a sua própria prioridade no *cluster*, usando métricas, como: grau do nó e tempo de vida do link. Os resultados obtidos pelos autores foram comparados com o mecanismo de cluster passivo proposto por Hashim; Shariff; Fadilah (2013) para MANETs.

Segundo os autores, o PassCAR não só aumenta a probabilidade de sucesso de descoberta de rotas, mas também seleciona os nós mais adequados para participar do cluster criado. Essa estrutura bem definida melhora significativamente a taxa de entrega de pacotes e atinge uma maior taxa de transferência na rede, devido à sua preferência por caminhos de roteamento confiáveis, estáveis e duráveis.

As métricas de roteamento também analisam a probabilidade de encaminhamento de pacotes RREQ por hop ao selecionar o próximo nó de encaminhamento, para construir uma estrutura de cluster eficiente para roteamento confiável e durável. A seguir, iremos conhecer as métricas do PassCAR e o cálculo da prioridade.

### 3.2.1.1 Grau do Nó

Estudos anteriores mostram que o grau do nó é um fator crucial que deve ser considerado na formação de *clusters* (HASHIM; SHARIFF; FADILAH, 2013). O grau de um nó é o número de nós da sua área de comunicação. Sem considerar o grau nó, a abordagem seria definida de forma aleatória e poderia selecionar um nó que não tem vizinhos com o *cluster head* ou *gateway*.

### 3.2.1.2 Tempo de vida do Link

Redes sem fio altamente dinâmicas são importante para reduzir a probabilidade de queda do link. Se a rota descoberta consiste em ligações mais estáveis, ele vai sobreviver por mais tempo. Como resultado, a relação na entrega dos dados e no aumento na sobrecarga para manter a rota diminui. Ayaida *et. al.* (2011) propôs uma medição, chamado de tempo de validade da ligação (LET), para selecionar o caminho de roteamento.

### 3.2.2 Forwarder Smart Selection Protocol for Limitation of Broadcast Storm Problem (SRB)

O protocolo SRB de Vegni; Natalizio (2015), propõe roteamento baseado em *clusters*. O objetivo da proposta destina-se a limitar o número de transmissões por pacotes. Através de uma escolha oportunista de veículo. Os pacotes são retransmitidos apenas pelos veículos selecionados, de modo a reduzir o número de encaminhadores, preservando um nível aceitável de QoS.

O SRB explora o comportamento dos veículos em uma partição das redes *Ad-Hoc*, a fim de detectar automaticamente os *clusters* de veículos por "zonas de interesse". Os pacotes são encaminhados apenas para veículos selecionados, de forma oportunista, eleitos como CH. Segundo os autores, o SRB foi avaliado em diferentes cenários de veículos, principalmente ambientes realistas, como cenários urbanos e de rodovias. A limitação do problema de congestionamento de transmissões, na forma da SRB, é expressa em termos de uma redução do número de *forwarders-next-hop*.

Ainda segundo os autores, a eficácia de SRB foi também comparada com (i) protocolos de transmissão tradicional, assim como (ii) uma abordagem de divulgação baseada em conteúdo, devido à sua capacidade para determinar os conjuntos de veículos com interesses comuns.

### 3.2.2.1 Transmissão Seletiva Confiável

O objetivo de SRB é: (i) evitar mensagens de retransmissão (isto é, uma limitação do problema de *broadcast*), e (ii) detectar aglomerados em forma automática e rápida. A abordagem proposta considera o processo de retransmissão mensagem dentro de um VANET, selecionando um número limitado de veículos, na qualidade de potenciais *forwarders-next-hop*. Longe de roteamento de *broadcast* tradicional, SRB detecta grupos de veículos de uma forma rápida e eficiente, e elege o CH para cada *cluster* detectado. O CH é, então, escolhido como o próximo encaminhador mensagem.

O SRB aproveita dois pressupostos principais (i) que a área dos veículos está particionada em setores adjacentes e (ii) que todos os veículos estão equipados com GPS e são capazes de estimar a sua própria posição. Cada partição transmite livremente mensagens a partir do veículo origem, e são identificados como partes de um círculo. O tamanho de cada setor é um processo dinâmico e pode variar conforme o sentido de transmissão de cada veículo.

Depois de detectar vários *clusters*, o veículo fonte elege o veículo mais distante dentro de cada *cluster*, onde o CH transmite uma mensagem apenas para esse veículo. Ao receber a mensagem, cada um CH irá tornar-se a origem da mensagem para a próxima fase de contenção e, assim, o algoritmo de SRB é repetido para os próximos saltos.

Esse protocolo apresenta problemas que comprometem a escalabilidade, confiabilidade, considerando que possui limitações nas retransmissões de pacotes e ainda limita o número de veículos que podem transmitir o que compromete o sistema.

### 3.2.3 Conclusão do Capítulo

Este capítulo teve como ênfase apresentar o funcionamento de algumas técnicas de agrupamentos para clusters e o funcionamento de alguns protocolos de roteamento baseados em clusters para VANETs que possam ser usados em aplicações de segurança pública.

A Tabela 1 apresenta as principais características dos protocolos de roteamento apresentados neste capítulo. Sendo possível concluir que nenhum dos protocolos pode atender todos os requisitos citados anteriormente como forma de fornecer uma comunicação confiável, escalável entre os veículos, para garantir que os vídeos transmitidos tenham níveis de qualidade assegurados.

Tabela 1 - Principais características dos protocolos de roteamento baseados em cluster

<b>Protocolos</b>	<b>Colaborativo</b>	<b>Link</b>	<b>Velocidade</b>	<b>Localização</b>	<b>Vídeo</b>
PASSCAR	Sim	Sim	Sim	Não	Não
SRB	Sim	Não	Não	Sim	Não

Fonte: Autoria própria.

Por fim, vale ressaltar que as soluções de roteamento baseadas em cluster é uma das melhores opções para VANETs, devido à alta mobilidade dos veículos e aos problemas de queda dos links, onde as soluções colaborativas minimizam esses problemas.

## 4 Materiais e Métodos

Este capítulo está dividido em duas seções; na seção 4.1 será apresentado o S.O (Sistema Operacional) *Android* e sua arquitetura. Na seção 4.2 serão apresentadas as ferramentas de desenvolvimento bem como as bibliotecas utilizadas no desenvolvimento da aplicação.

### 4.1 Android

O *Android* é uma plataforma para dispositivos móveis, que utiliza como linguagem de programação o Java, seu sistema operacional é baseado no *kernel* do Linux. Seu desenvolvimento contou com a ajuda de um grupo de gigantes empresas, denominado OHA (*Open Handset Alliance*) liderada pelo Google.

Na atualidade o *Android* pode ser encontrado em diversos dispositivos, como *smartphones e tablets*, TV (Google TV), Óculos (Google Gless), relógios (*Android Wear*), carros (*Android Auto*), e é o sistema mais utilizado no mundo (LECHETA, 2010).

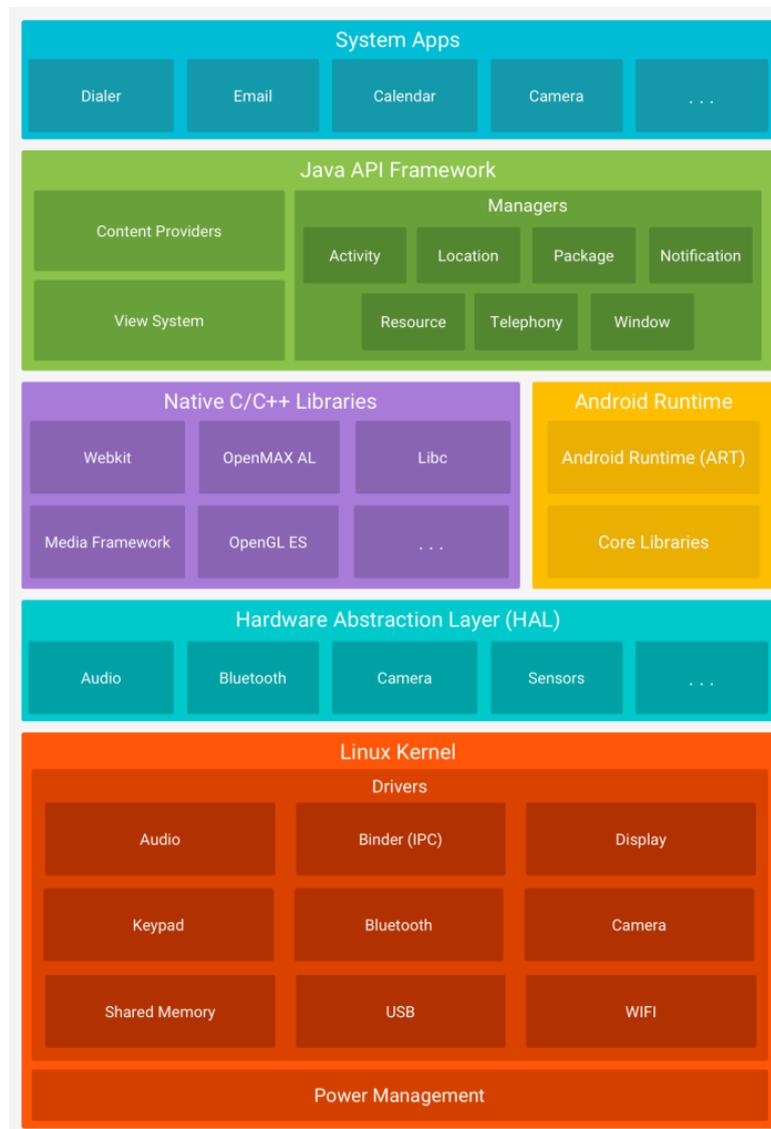
O *Android* oferece uma alternativa aberta para desenvolvimento de aplicativos móveis. Os desenvolvedores são livres para escrever aplicativos que aproveitam cada vez mais o poder de *hardware* e distribuí-los em um mercado aberto (MEIER, 2012).

Diversos dispositivos elegantes e versáteis contam com GPS (*Global Positioning System*), acelerômetros, NFC (*Near Field Communication*) e telas sensíveis ao toque, que juntos oferecem uma plataforma atrativa para criar aplicativos inovadores (MEIER, 2012).

#### 4.1.1 Arquiterura

A plataforma do *Android* é composta por uma pilha de *Softwares*, divididos nas seguintes camadas: *Kernel Linux*, Camada de Abstração de *Hardware*, *Android Runtime*, Bibliotecas, Estrutura da Java API e Aplicativos do Sistema, conforme é apresentada na figura 13.

Figura 13 - Arquitetura do Android



Fonte: [developer.com.br](http://developer.com.br)

#### 4.1.1.1 Kernel

Assim como os computadores, as arquiteturas dos dispositivos móveis são diferentes entre si. Para que haja a operação de um sistema em vários tipos de *hardware*, é necessário o sistema operacional. Para Tanenbaum (2005), o sistema operacional é responsável por gerenciar memória, processador, dispositivos de entrada e saída, e fornecer ao usuário uma interface mais simples com o *hardware*.

A tendência da criação de sistemas operacionais nos dispositivos móveis é crescente pela necessidade da gestão de recursos que os dispositivos oferecem como memória e processamento. Além disso, a criação de software não deve ser desenvolvida para um *hardware*, mas para um sistema operacional específico, dessa forma, o sistema fica responsável pelo gerenciamento de cada *hardware*.

O *Android* é baseado no *kernel Linux*, responsável pelo gerenciamento de memória, processos, *threads*, segurança de arquivos e pastas, redes e *drivers* de dispositivos. A segurança do Android é baseada na segurança do Linux. Para cada aplicação que é instalada no dispositivo é criado um usuário para ter acesso a sua estrutura de pastas, não permitindo que outros usuários tenham acesso a essa aplicação (LECHETA, 2010).

#### 4.1.1.2 Bibliotecas

Diferentes componentes e serviços essenciais do sistema Android, são implementados por código nativo que necessitam de bibliotecas nativas programadas em C e C++. O Android fornece a Java Framework APIs para expor as funcionalidades de algumas dessas bibliotecas nativas aos aplicativos (GOOGLE DEVELOPER, 2017).

Um exemplo é a utilização da biblioteca OpenCv, utilizada neste trabalho, para o funcionamento da detecção das partes de interesse da rodovia, faz-se necessário o uso do Android NDK (Kit de Desenvolvimento Nativo Android) para que seja acessada de forma direta as bibliotecas necessárias diretamente do código nativo.

#### 4.1.1.3 Android Runtime - ART

Nos dispositivos com a versão 5.0 (API nível 21) do Android, ou as mais atuais, as aplicações executam seu próprio processo com uma instância do Android Runtime. O ART foi projetado para processar várias máquinas virtuais nos dispositivos que apresentam baixa memória, executando arquivos (.dex), formato de *bytecode* específico para a plataforma Android.

#### 4.1.1.4 Android SDK

O *Android* SDK (Kit de Desenvolvimento de Software para Android), é o software utilizado para o desenvolvimento de aplicações na plataforma Android de modo nativo, possui um emulador que simula os dispositivos reais, apresenta ferramentas utilitárias e uma API completa para a linguagem Java (LECHETA, 2010).

## 4.2 Aplicação *Android*

Esta seção tem como objetivo apresentar as ferramentas utilizadas no desenvolvimento da aplicação proposta deste trabalho. Apresentando o *Hardware* e o *Software* utilizados e o desenvolvimento da aplicação.

### 4.2.1 Hardwares

Foi utilizado o *Smartphone* da SAMSUNG *Galaxy J5 Metal*, com Sistema Operacional *Android* 6.0, Processador 1.2 GHz *Quad Core*, 2GB de RAM, Tela de 5.2" com resolução de 720x1280 *pixel*, para a realização dos testes e envio dos vídeos.

Para a recepção dos vídeos, foi utilizado um Computador Servidor, com o sistema operacional Linux (Ubuntu), o vídeo recebido será salvo com as coordenadas do seu envio.

### 4.2.2 Software

A ferramenta foi projetada na plataforma *Android*, dessa forma foi utilizado para seu desenvolvimento a IDE (Ambiente de Desenvolvimento Integrado, do inglês, *Integrated Development Environment*) *Android Studio*, IDE oficial para *Android*.

O *Android Studio* apresenta um editor visual com várias opções, conta com um sistema de *build* moderno que se baseia no Gradle, é integrado ao Android SDK,

disponibilizando várias utilidades e funcionalidades para o desenvolvimento, contém *layout* para *smartphones*, *tablets*, Tv, relógios (LECHETA, 2010).

Para o desenvolvimento da ferramenta, foi definida a versão 4.1, API 16 do SDK, denominada de *Jelly Bean*. Como ilustrado na tabela 2, é a versão que atinge o maior número de usuário (GOOGLE DEVELOPER, 2017) e foi o motivo da utilização de determinada distribuição. Vale ressaltar que os aplicativos que utilizam essa versão, são compatíveis com as versões mais atuais, dessa forma, a ferramenta poderá ser utilizada nos dispositivos atuais.

Tabela 2 - Distriuição das Versões do Android

Android Version	Codename	API	Distribuição
4.0	Ice Cream Sandwich	15	-
4.1	Jelly Bean	16	99,20%
4.2	Jelly Bean	17	96%
4.3	Jelly Bean	18	91,40%
4.4	KitKat	19	90,10%
5.0	Lollipop	21	71,30%
5.1	Lollipop	22	62,60%
6.0	Marshmallow	23	39,30%
7.0	Nougat	24	8,10%
7.1	Nougat	25	1,50%

Fonte: developer.com.br

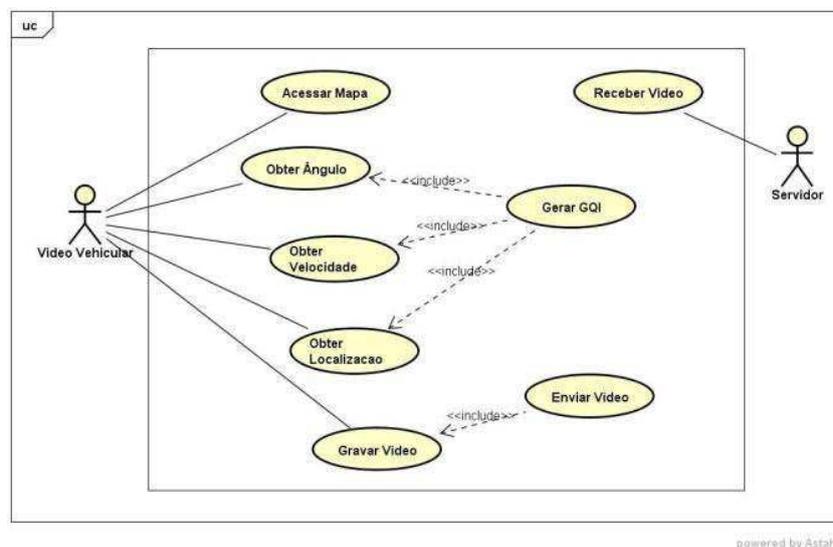
Além de contar com a biblioteca OpenCv 2.4.9 para a utilização da detecção da parte de interesse da rodovia e gravação do vídeo. A biblioteca OpenCV, tem como objetivo tornar a visão computacional aproximável a desenvolvedores e usuários nas áreas de interação humano-computador em tempo real.

A biblioteca se encontra dividida em cinco grupos: Análise estrutural, Análise de movimento e rastreamento de objetos, Processamento de imagens, Reconhecimento de imagens, Reconhecimento de padrões e Calibração de câmera e reconstrução 3D.

### 4.2.3 Desenvolvimento

Para que a aplicação fosse desenvolvida, foi necessário primeiramente coletar alguns parâmetros (ângulo do vídeo, velocidade e localização) para calcular o Indicador de Qualidade do Gateway e assim transmitir o vídeo. A figura 14 apresenta a modelagem da aplicação.

Figura 14 - Modelagem do Caso de Uso da Aplicação.



Fonte: Autoria própria.

No desenvolvimento utilizou-se a API (*Application Programming Interface*) Google Maps Android API V2 (Versão 2), com o objetivo de construção de mapas e que possui recursos de controle de zoom, tipo de visualização, desenho de imagens (marcadores), entre outros. Além disso, também foi utilizada a API de Localização (*Fused Location Provider*).

Essas APIs foram conectadas ao Google Play Services, aplicativo distribuído pelo Google que apresenta funcionalidades essenciais para a comunicação dos serviços do Google, com a finalidade de detectar de forma rápida a última localização do veículo.

#### 4.2.3.1 Implementação do Algoritmo

Baseando-se nos trabalhos relacionados percebeu-se que os protocolos de roteamento mencionados não atendem a todos os requisitos para fornecer uma comunicação confiável. Com isso, foi implementado um algoritmo para realizar o cálculo do Indicador de Qualidade do Gateway, sendo utilizados os parâmetros: ângulo de gravação, velocidade do veículo e localização do veículo em relação à NodeB, conforme é mostrado no algoritmo 1.

---

Algoritmo 1 - Algoritmo da Aplicação

---

Início

Identificar o ângulo de gravação  
 Classificar a localização do veículo em relação à NodeB  
 Classificar a velocidade do veículo  
 Gerar o cálculo do Indicador de Qualidade  
 Transmitir o vídeo

Fim

---

Fonte: Autoria Própria

Abaixo, são apresentados as formas para obter os requisitos necessários para a implementação do algoritmo.

#### 4.2.3.2 Obter o ângulo de Gravação

Considerando o ângulo da câmera na horizontal e definindo como maior valor o ângulo de 180 graus, foram definidas três regiões de gravação, sendo elas: lado esquerdo, central e lado direito, conforme é mostrado na tabela 3.

Tabela 3 - Parâmetro com as regiões de interesse e as prioridades do ângulo

<b>Região de Gravação</b>	<b>Esquerdo</b>	<b>Centro</b>	<b>Direito</b>
Ângulo	( $\geq 0$ e $< 80$ )	( $\geq 80$ e $\leq 90$ )	( $91 \leq 180$ )
Prioridade	1	2	1

Fonte: Autoria própria.

Foram utilizadas duas formas para análise e testes do ângulo de gravação, e as duas tiveram resultados diferentes, conforme será descrito abaixo.

No primeiro teste, foi utilizada a biblioteca OpenCv, para detectar a região de interesse e retornar esse valor para o cálculo do GQI, porém, é apresentada uma dificuldade para gravação do vídeo a ser transmitido.

No segundo teste, utilizou-se a classe SurfaceView, onde o mesmo realiza a gravação do vídeo e permite o envio, contudo, não tem função de detectar região a interesse.

No apêndice A é possível verificar como foi determinada o ângulo de interesse.

#### 4.2.3.3 Obter a Localização

Para obter o local onde se encontra o veículo, foi recuperada a latitude e longitude a partir do GPS do dispositivo. Neste trabalho as torres de celular de uma determinada operadora será utilizada como referência para as NodeBs, as quais utilizam LTE e encaminharam os vídeos enviados do veículo ao servidor.

Foi realizado um levantamento das localidades dessas torres na região metropolitana de Belém e realizado um mapeamento e adicionados ao Mapa da aplicação, através do aplicativo *NetMonitor*.

Dessa forma, foi possível determinar a distância que o veículo está da NodeB ou mesmo se ele não está dentro da faixa de recepção de sinal. Foram definidos três setores de interesse e suas prioridades que pode ser visualizado na tabela 4.

Tabela 4 - Setores e Prioridades da localização do veículo e das NodeB.

<b>Localização</b>	<b>Setor 1</b>	<b>Setor 2</b>	<b>Setor 3</b>
Distância da NodeB	1000m	500m	250m
Prioridade	1	2	3

Fonte: Autoria própria.

O apêndice C é mostrado o método de obter a prioridade conforme a sua localização.

#### 4.2.3.4 Obter Velocidade do Veículo

Não foi possível detectar a velocidade real do veículo, em relação a velocidade identificada no velocímetro, como alternativa foi utilizado o método `getSpeed()` da API de Localização, que retorna a velocidade aproximada. A partir da coleta dessa velocidade, é possível comparar aos intervalos de velocidades determinada pelo algoritmo, conforme é apresentado na tabela 5.

Tabela 5 - Parâmetro de Prioridade da Velocidade

<b>Km/h</b>	<b>Menor 20</b>	<b>20 a 39</b>	<b>40 a 59</b>	<b>60 a 79</b>	<b>80 a 99</b>	<b>Maior 100</b>
<b>Prioridade</b>	6	5	4	3	2	1

Fonte: Autoria própria.

De acordo com a tabela 5, conclui-se que o veículo que possuir menor velocidade, estará mais tempo conectado a uma NodeB, dessa forma, sua prioridade é maior do que um veículo que apresenta velocidade maior.

No apêndice B é exibido o código de prioridade obtida da velocidade.

#### 4.2.3.5 Calcular o Indicador de Qualidade do Gateway

Após obter as prioridades dos parâmetros necessários, e os pesos definidos pelo algoritmo, pode-se calcular o Indicador de Qualidade de Gateway. Na tabela 6 são apresentados os parâmetros de peso de cada item obtido.

Tabela 6 - Parâmetro de Pesos

<b>Parâmetro</b>	<b>Vídeo</b>	<b>Velocidade</b>	<b>Localização</b>
<b>Prioridade</b>	0.5	0.3	0.2

Fonte: Autoria própria.

A definição dos pesos dos parâmetros (vídeo, velocidade e localização) foram baseados em testes de *upload* dos vídeos.

Abaixo é descrito a motivação dos valores de peso para cada parâmetros.

- **Vídeo:** a qualidade do vídeo é o fator que mais implica no envio, com isso sua prioridade foi definida como a maior.
- **Velocidade:** o veículo pode obter baixas e altas velocidades, porém, não é um fator que implica tanto na transmissão do vídeo.

- Localização: mesmo que o veículo não esteja ao alcance de uma NodeB, ele pode assim que obter sinal, enviar a informação que coletou.

Diante das prioridades e os pesos determinados, o GQI retorna um valor ao usuário, esse valor define a qualidade do sinal para a distribuição do vídeo. A equação 2, apresenta o cálculo do GQI.

$$GQI = \alpha(P_{vídeo}) + \beta(P_{velocidade}) + \gamma(P_{localização}) \quad (2)$$

Onde:

- $P_{vídeo}$  = Prioridade do ângulo do vídeo;
- $P_{velocidade}$  = Prioridade da velocidade do veículo;
- $P_{localização}$  = Prioridade da localização.

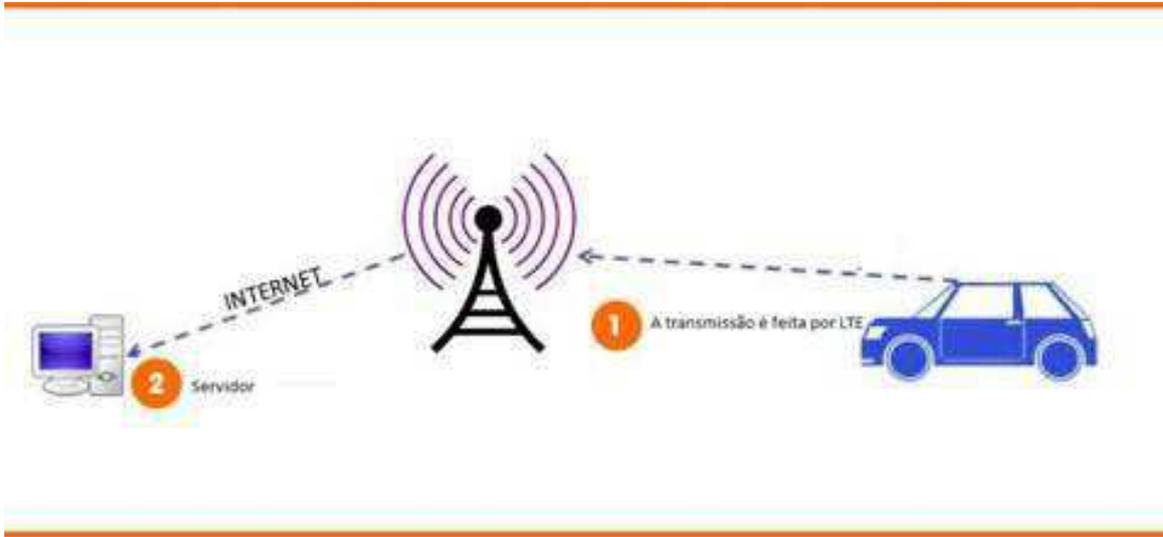
O Cálculo do GQI (no apêndice D) foi chamado no método `onLocationChanged(Location location)`, conforme é apresentado no apêndice E e, dessa forma, a cada atualização dos parâmetros obtidos, o valor do GQI será atualizado, além, dos valores da velocidade, localização e ângulo de gravação.

#### 4.2.3.6 Transmissão do Vídeo

Segundo a documentação do Android, os seguintes formatos de vídeos são suportados para desenvolver aplicações: mp4, formato escolhido como padrão neste trabalho, e 3gp. E os protocolos de comunicação são: HTTP (*Hyper Text Transfer Protocol*), protocolo utilizado neste trabalho, HTTPS *Hyper Text Transfer Protocol Secure* e RTSP (*Real Time Streaming Protocol*), sendo este último recomendado para streaming.

Após a gravação do vídeo, o usuário poderá ou não realizar a transmissão para o servidor responsável pelo armazenamento dos vídeos. A figura 15 ilustra o funcionamento da transmissão de vídeo na comunicação V2I.

Figura 15 - Transmissão de Vídeo na Comunicação V2I.



Fonte: Adaptada da ADEASP, 2017.

Na transmissão do vídeo, não foi levando em consideração os métodos de compressão ou tratamento do vídeo, o mesmo é transmitido de forma bruta e enviado ao servidor. São gravados vídeos de um minuto e armazenado na memória do dispositivo, entretanto não ocorre gravação full time ainda.

#### 4.2.4 Permissões de Acesso aos Recursos da Aplicação

O *Android* é um sistema operacional de privilégio separado em que cada aplicativo executa com uma identidade de sistema distinta (GOOGLE, 2017). A partir do *Android* 6.0, a versão do dispositivo que foi utilizado no projeto, é necessário que os aplicativos solicitem determinadas permissões em tempo de execução, para que se possam utilizar os recursos de localização, câmera, internet, entre outros (LECHETA, 2010).

As permissões são separadas em duas categorias: normais e perigosas. As normais são aquelas acessadas de forma direta, sem a necessidade de confirmação do usuário, por exemplo, o uso da internet. Por outro lado, as perigosas, são aquelas em que o usuário precisa confirmar a permissão para a utilização do recurso, por exemplo, localização/GPS, câmera, escrita em memória, e outras.

Neste trabalho, foram utilizadas algumas permissões para o funcionamento da ferramenta. Para obter a localização do veículo, permitir a gravação do vídeo que será enviado ao servidor, escrita e leitura na memória e internet, conforme apresenta A figura 16 apresenta as permissões declaradas no arquivo AndroidManifest.xml que foram utilizadas no projeto.

Figura 16 - Permissões para acesso dos recursos no Android.

```
<!-- Permissões -->
<uses-permission android:name="android.permission.CAMERA"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<!-- Gps por Hardware -->
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<!-- Obter a localização por wifi ou triangulação de Antenas -->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```

Fonte: Autoria própria.

### 4.3 Conclusão do capítulo

Este capítulo teve como objetivo apresentar as principais camadas do sistema operacional Android e o desenvolvimento da aplicação destacando os parâmetros necessários para a implementação do algoritmo e gerar o Indicador de Qualidade do Gateway e transmitir o vídeo gravado. Além de apresentar as dificuldades encontradas no momento de classificar o ângulo de gravação e as permissões necessárias para a utilização dos recursos da aplicação.

## **5 Aplicação Desenvolvida**

Neste capítulo são apresentadas as telas de funcionamento da aplicação desenvolvida neste trabalho.

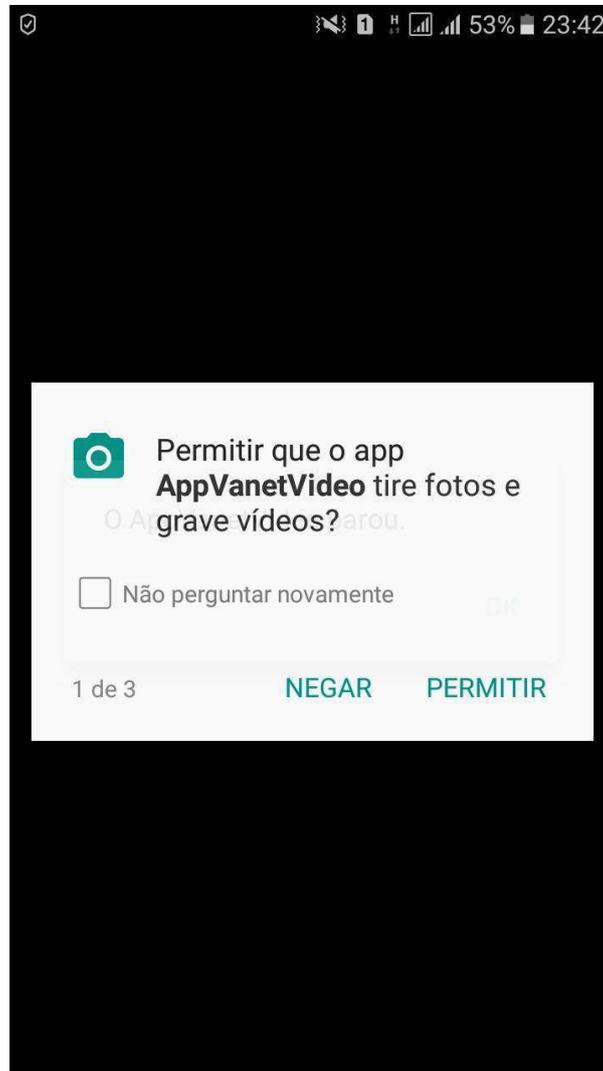
### **5.1 Funcionamento da Aplicação Proposta**

As funcionalidades da ferramenta AppVanetVideo a princípio foram projetadas de forma simples e com poucos recursos a serem alterados pelos usuários. A seguir são mostradas as telas da aplicação e suas finalidades.

#### **5.1.1 Primeiro Acesso**

Ao iniciar a aplicação pela primeira vez serão solicitadas as permissões para utilizar os recursos de localização, acesso a câmera e ao Microfone, conforme a figura 17, onde é preciso que o usuário aceite a permissão de tirar fotos e gravar vídeos.

Figura 17 - Permissão solicitada pela aplicação.



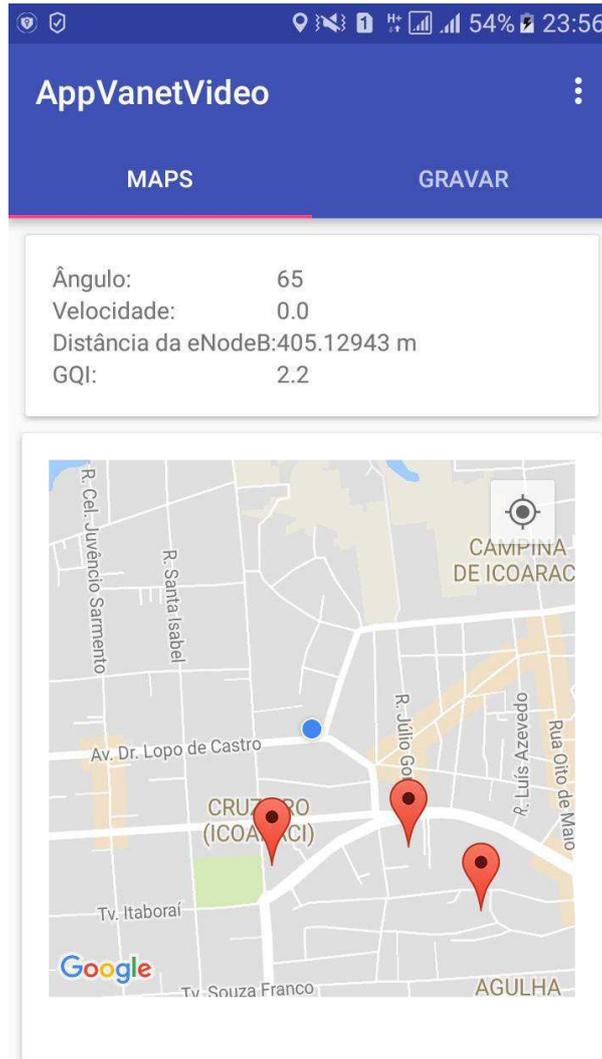
Fonte: Autoria própria.

### 5.1.2 Valor do GQI e Mapa

Após as permissões solicitadas, a tela principal é exibida ao usuário, conforme apresentada na figura 18, são exibidas as informações dos parâmetros apresentados no capítulo 4 e o mapa com as NodeB próximas do veículo.

Os valores de ângulo, velocidade, distância para a NodeB, são coletados pela aplicação, e a partir desses valores é retornado o valor do GQI.

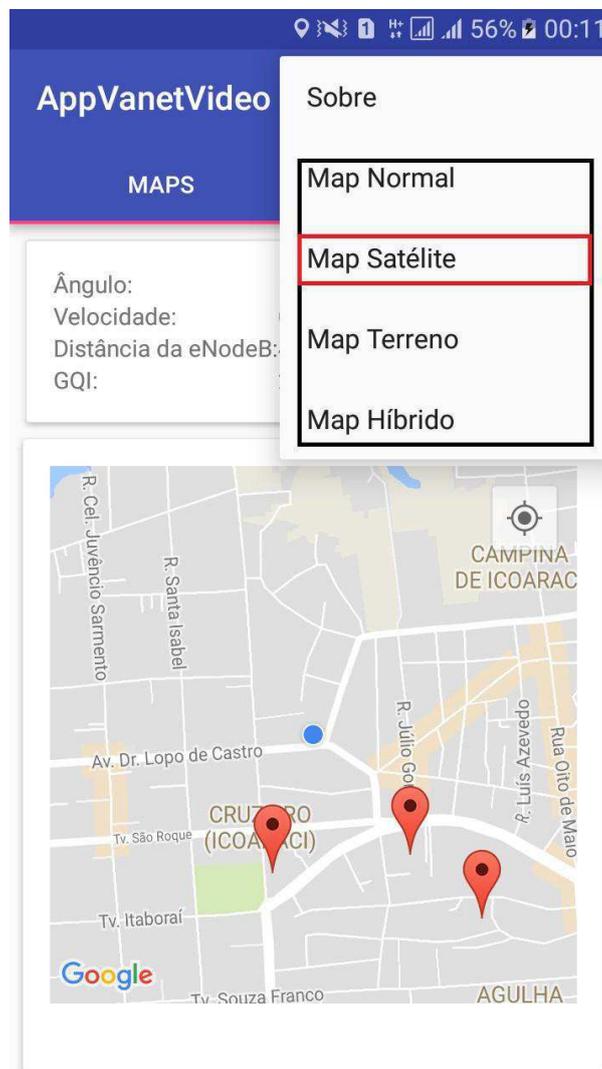
Figura 18 - Tela de Informações e NodeB próximas.



Fonte: Autoria própria.

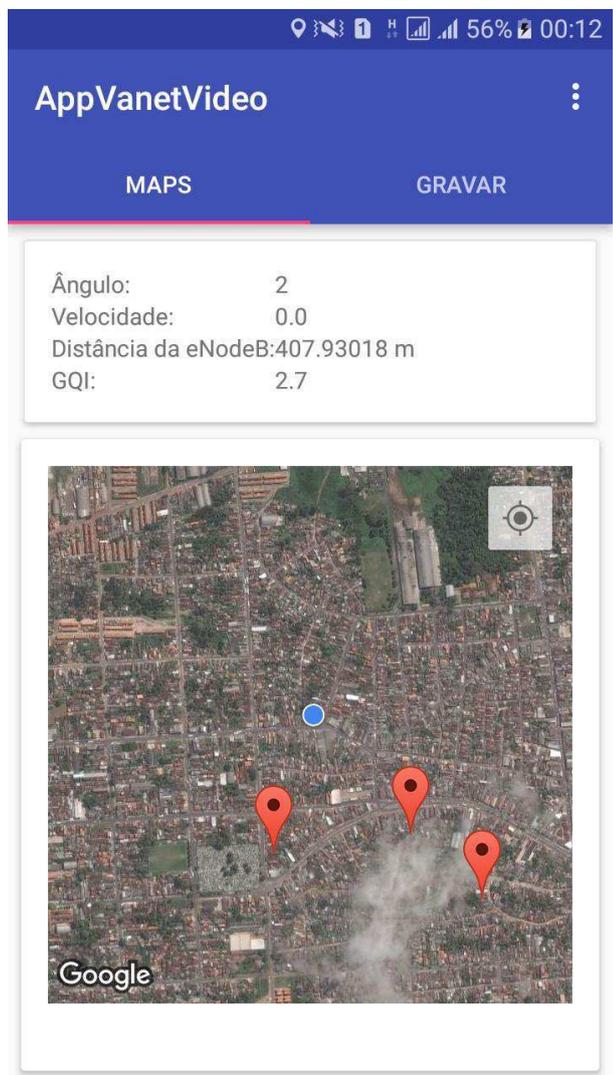
Nesta mesma tela, o usuário tem a opção de visualizar o mapa de outras forma, conforme é apresetada na figura 19 e a figura 20 exhibe a alteração da visualização do mapa. Nesse caso, foi selecionado o Modo Satélite.

Figura 19 - Menu com opções.



Fonte: Autoria própria.

Figura 20 - Alteração do Mapa.



Fonte: Autoria própria.

### 5.1.3 Gravar e Enviar Vídeo

A partir das figuras anteriores foi possível observar que se são utilizadas abas para as telas de Mapas e Gravar. No momento que o usuário desliza ou clica na aba Gravar, é mostrada o fragmento da tela que permite a gravação e transmissão do vídeo, conforme a figura 21.

Figura 21 - Tela de Gravação e Envio.



Fonte: Autoria própria.

## 6 CONSIDERAÇÕES FINAIS

Através do trabalho realizado é possível observar vários protocolos de roteamento para as redes veiculares, porém, são poucas as aplicações voltadas para a finalidade de transmissão de vídeos utilizando algoritmos que identificam parâmetros para determinar a qualidade dessa transmissão.

A ferramenta desenvolvida neste trabalho propôs um algoritmo para identificar um Indicador de Qualidade, onde foi levado em consideração três requisitos (ângulo de gravação, velocidade e localização). A partir do valor do Indicador, o usuário tomaria a decisão ou saberia o momento oportuno para realizar a transmissão do seu vídeo.

Durante a etapa de desenvolvimento, surgiram alguns obstáculos, como a identificação do ângulo de gravação, que foi um dos fatores mais complicados de solucionar, dessa forma, o indicador de qualidade não ficou completamente funcional. Apesar desse obstáculo, a ferramenta obteve as funcionalidades definidas neste trabalho.

### 6.1 Trabalhos Futuros

Existem diversas melhorias na aplicação que ainda podem ser desenvolvidas para ficar ainda mais robusta, e melhorar a experiência do usuário, como por exemplo:

- Identificar o ângulo de gravação de uma forma precisa e ao mesmo tempo poder gravar o vídeo;
- Alertas de sinal, informando quando o sinal estiver bom para enviar o vídeo, sem que o usuário tenha que ficar acompanhando;
- As interfaces gráficas, podem ser melhoradas e aplicados novos recursos.

Outro objetivo da aplicação será implementar em protocolos de roteamento colaborativos, com isso, vários veículos poderão gerar seus indicadores de qualidade, e avaliar quais veículos terão o melhor momento para a distribuição do vídeo.

## REFERÊNCIAS

AISSAOUI, R. *et al.* Advanced real-time traffic monitoring system based on v2x communications. In: IEEE. **Communications (ICC), 2014 IEEE International Conference on**. [S.l.], 2014. p. 2713–2718.

ALASMARY, W.; ZHUANG, W. Mobility impact in ieee 802.11 p infrastructureless vehicular networks. **Ad Hoc Networks**, Elsevier, v. 10, n. 2, p. 222–230, 2012.

AYAIDA, M. *et al.* Improving the link lifetime in vanets. In: IEEE. **Local Computer Networks (LCN), 2011 IEEE 36th Conference on**. [S.l.], 2011. p. 905–912.

BARBA, C. T. *et al.* Smart city for vanets using warning messages, traffic statistics and intelligent traffic lights. In: IEEE. **Intelligent Vehicles Symposium (IV), 2012 IEEE**. [S.l.], 2012. p. 902–907.

BARFIELD, W.; DINGUS, T. A. **Human factors in intelligent transportation systems**. [S.l.]: Psychology Press, 2014.

BRADAI, A.; AHMED, T.; BENSLIMANE, A. Vehicular communications. 2014.

CAMPOLO, C. *et al.* Modeling broadcasting in ieee 802.11 p/wave vehicular networks. **IEEE Communications Letters**, IEEE, v. 15, n. 2, p. 199–201, 2011.

CHEN, S.; PANDE, A.; MOHAPATRA, P. Sensor-assisted facial recognition: an enhanced biometric authentication system for smartphones. In: ACM. **Proceedings of the 12th annual international conference on Mobile systems, applications, and services**. [S.l.], 2014. p. 109–122.

CHO, J.-H.; SWAMI, A.; CHEN, R. A survey on trust management for mobile ad hoc networks. **IEEE Communications Surveys & Tutorials**, IEEE, v. 13, n. 4, p. 562–583, 2011.

CHU, Y.-C.; HUANG, N.-F. An efficient traffic information forwarding solution for vehicle safety communications on highways. **IEEE Transactions on Intelligent Transportation Systems**, IEEE, v. 13, n. 2, p. 631–643, 2012.

COX, C. **An introduction to LTE: LTE, LTE-advanced, SAE and 4G mobile communications**. [S.l.]: John Wiley & Sons, 2012.

DAHLMAN, E.; PARKVALL, S.; SKOLD, J. **4G: LTE/LTE-advanced for mobile broadband**. [S.l.]: Academic press, 2013.

DIMITRAKOPOULOS, G. Intelligent transportation systems based on internet-connected vehicles: Fundamental research areas and challenges. In: IEEE. **ITS**

**Telecommunications (ITST), 2011 11th International Conference on.** [S.l.], 2011. p. 145–151.

AOUZI, N.-E. E.; LEUNG, H.; KURIAN, A. Data fusion in intelligent transportation systems: Progress and challenges—a survey. **Information Fusion**, Elsevier, v. 12, n. 1, p. 4–10, 2011.

FELICE, M. D. et al. Traffic monitoring and incident detection through vanets. In: IEEE. **Wireless On-demand Network Systems and Services (WONS), 2014 11th Annual Conference on.** [S.l.], 2014. p. 122–129.

GAO, X. et al. Image quality assessment based on multiscale geometric analysis. **IEEE Transactions on Image Processing**, IEEE, v. 18, n. 7, p. 1409–1423, 2009.

GERLA, M. et al. Content distribution in vanets. **Vehicular Communications**, Elsevier, v. 1, n. 1, p. 3–12, 2014.

GOOGLE. **Developer Android.** 2017. Acesso 03 de Setembro de 2017. Disponível em: <https://developer.android.com>.

GOZALVEZ, J.; SEPULCRE, M.; BAUZA, R. IEEE 802.11 p vehicle to infrastructure communications in urban environments. **IEEE Communications Magazine**, IEEE, v. 50, n. 5, 2012.

HASHIM, W. W.; SHARIFF, A. M.; FADILAH, S. I. A new hop-count and node-degree based routing protocol for vehicular ad-hoc networks (vanets). In: IEEE. **Communications (MICC), 2013 IEEE Malaysia International Conference on.** [S.l.], 2013. p. 310–315.

HOLMA, H.; TOSKALA, A. **LTE for UMTS: Evolution to LTE-advanced.** [S.l.]: John Wiley & Sons, 2011.

JAGADISH, S.; MANIVASAKAN, R. Analysis of jitter control algorithms in qos networks. In: IEEE. **Internet (AH-ICI), 2011 Second Asian Himalayas International Conference on.** [S.l.], 2011. p. 1–5.

JOTHI, K.; JEYAKUMAR, A. E. A survey on broadcasting protocols in vanets. Citeseer, 2013.

KAKKASAGERI, M.; MANVI, S. Multiagent driven dynamic clustering of vehicles in vanets. **Journal of Network and Computer Applications**, Elsevier, v. 35, n. 6, p. 1771–1780, 2012.

KARAGIANNIS, G. et al. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. **IEEE communications surveys & tutorials**, IEEE, v. 13, n. 4, p. 584–616, 2011.

KENNEY, J. B. Dedicated short-range communications (dsrc) standards in the united states. **Proceedings of the IEEE**, IEEE, v. 99, n. 7, p. 1162–1182, 2011.

KIM, Y.; KIM, I. Security issues in vehicular networks. In: IEEE. **Information Networking (ICOIN), 2013 International Conference on**. [S.l.], 2013. p. 468–472.

KUROSE, J. F.; ROSS, K. W. **Computer networking: A top-down approach featuring the internet, 6/E**. [S.l.]: Pearson Education India, 2012.

LECHETA, R. R. Google android. **S~ao Paulo: Editora Novatec**, 2010.

MEIER, R. **Professional Android 4 application development**. [S.l.]: John Wiley & Sons, 2012.

MOHAMMAD, S. A.; RASHEED, A.; QAYYUM, A. Vanet architectures and protocol stacks: a survey. In: SPRINGER. **International Workshop on Communication Technologies for Vehicles**. [S.l.], 2011. p. 95–105.

MOORTHY, A. K. et al. Subjective analysis of video quality on mobile devices. In: **Sixth International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM), Scottsdale, Arizona**. [S.l.: s.n.], 2012.

MU, M. et al. Quality of experience management framework for real-time multimedia applications. **International Journal of Internet Protocol Technology**, v. 4, p. 54–64, 2009.

NOORI, H.; VALKAMA, M. Impact of vanet-based v2x communication using ieee 802.11 p on reducing vehicles traveling time in realistic large scale urban area. In: IEEE. **Connected Vehicles and Expo (ICCVE), 2013 International Conference on**. [S.l.], 2013. p. 654–661.

OUELLETTE, S.; MARCHAND, L.; PIERRE, S. A potential evolution of the policy and charging control/qos architecture for the 3gpp ietf-based evolved packet core. **IEEE Communications Magazine**, IEEE, v. 49, n. 5, 2011.

PACHECO-PARAMO, D.; AKYILDIZ, I. F.; CASARES-GINER, V. X2-interface-based location management for small cell networks. In: IEEE. **Globecom Workshops (GC Wkshps), 2013 IEEE**. [S.l.], 2013. p. 4976–4981.

PHAM, T. A. Q.; PIAMRAT, K.; VIHO, C. Qoe-aware routing for video streaming over vanets. In: IEEE. **Vehicular Technology Conference (VTC Fall), 2014 IEEE 80th**. [S.l.], 2014. p. 1–5.

QUADROS, C. et al. A multi-flow-driven mechanism to support live video streaming on vanets. In: IEEE. **Computer Networks and Distributed Systems (SBRC), 2014 Brazilian Symposium on**. [S.l.], 2014. p. 468–476.

QUAN, W. et al. Tb2f: Tree-bitmap and bloom-filter for a scalable and efficient name lookup in content-centric networking. In: IEEE. **Networking conference, 2014 IFIP**. [S.l.], 2014. p. 1–9.

REZAEI, F.; HEMPEL, M.; SHARIF, H. Towards the performance evaluation of 4th generation wireless communication standards: Lte versus mobile wimax. **Wireless**

**Communications and Mobile Computing**, Wiley Online Library, v. 15, n. 8, p. 1290–1307, 2015.

SERRAL-GRACIA, R. et al. An overview of quality of experience measurement challenges for video applications in ip networks. In: SPRINGER. **International Conference on Wired/Wireless Internet Communications**. [S.l.], 2010. p. 252–263.

SESTIA, S.; BAKER, M.; TOUFIK, I. **LTE-the UMTS long term evolution: from theory to practice**. [S.l.]: John Wiley & Sons, 2011.

SHAREF, B. T.; ALSAQOUR, R. A.; ISMAIL, M. Vehicular communication ad hoc routing protocols: A survey. **Journal of network and computer applications**, Elsevier, v. 40, p. 363–396, 2014.

SHEN, Z. et al. Overview of 3gpp lte-advanced carrier aggregation for 4g wireless communications. **IEEE Communications Magazine**, IEEE, v. 50, n. 2, 2012.

SONG, T. et al. Si/ge double-layered nanotube array as a lithium ion battery anode. **Acs Nano**, ACS Publications, v. 6, n. 1, p. 303–309, 2012.

TAL, I.; MUNTEAN, G.-M. User-oriented cluster-based solution for multimedia content delivery over vanets. In: IEEE. **Broadband Multimedia Systems and Broadcasting (BMSB), 2012 IEEE International Symposium on**. [S.l.], 2012. p. 1–5.

TANENBAUM, A. S. **Sistemas operacionais modernos**. [S.l.]: Pearson Education Do, 2005. v. 2.

VEGNI, A. M.; NATALIZIO, E. Forwarder smart selection protocol for limitation of broadcast storm problem. **Journal of network and computer applications**, Elsevier, v. 47, p. 61–71, 2015.

VODOPIVEC, S.; BESTER, J.; KOS, A. A survey on clustering algorithms for vehicular ad-hoc networks. In: IEEE. **Telecommunications and Signal Processing (TSP), 2012 35th International Conference on**. [S.l.], 2012. p. 52–56.

WANG, S.-S.; LIN, Y.-S. Passcar: A passive clustering aided routing protocol for vehicular ad hoc networks. **Computer Communications**, Elsevier, v. 36, n. 2, p. 170–179, 2013.

ZHOU, Y. et al. A cluster-based method to map urban area from dmsp/ols nightlights. **Remote Sensing of Environment**, Elsevier, v. 147, p. 173–185, 2014.

## APÉNDICE

## APÊNDICE A – CÓDIGO DE PRIORIDADE DO ÂNGULO

```
1 public static float prioridadeAngulo()
2 {
3     if (anguloAleatorio() >= 80 && anguloAleatorio() <= 90)
4         anguloPrioridade = 2;
5     else
6         anguloPrioridade = 1;
7
8     return anguloPrioridade;
9 }
10
11 //Ate o momento o angulo esta sendo de forma aleatoria
12 public static int anguloAleatorio()
13 {
14     Random random = new Random();
15     int anguloMaximo = 180;
16     int anguloMinimo = 0;
17     angulo = random.nextInt((anguloMaximo - anguloMinimo) + 1)
18         + anguloMinimo;
19     return angulo;
20 }
```

## APÊNDICE B – CÓDIGO DE PRIORIDADE DA VELOCIDADE

```
1
2     public static float prioridadeVelocidade(float velocidade)
3     {
4
5         if (velocidade >= 100)
6             velocidadePrioridade = 1;
7         else if (velocidade <= 99)
8             velocidadePrioridade = 2;
9         else if (velocidade <= 79)
10            velocidadePrioridade = 3;
11        else if (velocidade <= 59)
12            velocidadePrioridade = 4;
13        else if (velocidade <= 39)
14            velocidadePrioridade = 5;
15        else if (velocidade <= 20)
16            velocidadePrioridade = 6;
17
18        return velocidadePrioridade;
19    }
```

## APÊNDICE C – CÓDIGO DE PRIORIDADE DA LOCALIZAÇÃO

```
1 public static float prioridadeLocalizacao(float distancia)
2 {
3     //Adotando que a eNodeB envia sinal ate 1000m.
4     if (distancia <= 250)
5         localizacaoPrioridade = 3;
6     else if (distancia <= 500)
7         localizacaoPrioridade = 2;
8     else if (distancia <= 1000)
9         localizacaoPrioridade = 1;
10
11     return localizacaoPrioridade;
12 }
```

## APÊNDICE D – CÁLCULO DO GQI

```
1 //Calcular o GQI
2 public static float calcularGqi(float vel, float dist)
3 {
4     float resGQI;
5     resGQI = (float) ((pesoVideo * prioridadeAngulo() +
6         pesoVelocidade * prioridadeVelocidade(vel) +
7         pesoLocalizacao * prioridadeLocalizacao(dist))/10.0);
8
9     return resGQI;
}
```

## APÊNDICE E – VALORES DOS PARÂMETROS E DO GQI ATUALIZADO

```
1      @Override
2      public void onLocationChanged(Location location) {
3
4          ArrayList<LatLng> posicao = ListaLocais.listarLocais();
5          Location target = new Location("eNodeB");
6
7          for (int i = 0; i < posicao.size(); i++)
8          {
9              for (LatLng ponto : new LatLng[]{posicao.get(i)})
10             {
11                 target.setLatitude(ponto.latitude);
12                 target.setLongitude(ponto.longitude);
13
14                 if (location.distanceTo(target) <= 600)
15                 {
16                     distancia = location.distanceTo(target);
17                 }
18             }
19         }
20
21         resDistancia = Gqi.prioridadeLocalizacao(distancia);
22         velocidadeAtual = (location.getSpeed() * 3600 / 1000);
23         resVelocidade = Gqi.prioridadeVelocidade(velocidadeAtual);
24
25         gqi = Gqi.calcularGqi(resVelocidade, resDistancia);
26         resGqi = gqi;
27
28         tAngulo.setText(Gqi.anguloAleatorio()+"");
29         tDistancia.setText(distancia+" m");
30         tVelocidade.setText(velocidadeAtual+"");
31         tGqi.setText(resGqi+"");
32     }
```