

Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia

Departamento de Computação

Programa de Pós-Graduação em Ciência da Computação

**Rede Overlay de Suporte a Ambientes
Virtuais Colaborativos em Redes Veiculares**

Rodolfo Ipolito Meneguette

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação, do Departamento de Computação, da Universidade Federal de São Carlos, como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Processamento de Imagens e Sinais - PIS.

SÃO CARLOS - SP

JULHO – 2009

**Ficha catalográfica elaborada pelo DePT da
Biblioteca Comunitária da UFSCar**

M541ro

Meneguette, Rodolfo Ipolito.

Rede overlay de suporte a ambientes virtuais
colaborativos em redes veiculares / Rodolfo Ipolito
Meneguette. -- São Carlos : UFSCar, 2009.
93 f.

Dissertação (Mestrado) -- Universidade Federal de São
Carlos, 2009.

1. Redes locais de computação. 2. Peer-to-peer. 3.
Ambiente virtual. I. Título.

CDD: 004.68 (20ª)


Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Ciência da Computação

“Rede Overlay de Suporte a Ambientes Virtuais
Colaborativos em Redes Ad Hoc Veiculares”

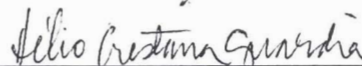
RODOLFO IPÓLITO MENEGUETTE

Dissertação de Mestrado apresentada ao
Programa de Pós-Graduação em Ciência da
Computação da Universidade Federal de São
Carlos, como parte dos requisitos para a
obtenção do título de Mestre em Ciência da
Computação


Membros da Banca:



Profa. Dra. Regina Borges de Araujo
(Orientadora - DC/UFSCar)



Prof. Dr. Hélio Crestana Guardia
(DC/UFSCar)



Prof. Dr. Edmundo Roberto Mauro Madeira
(IC/UNICAMP)

São Carlos
Julho/2009

Diz a sabedoria popular chinesa que toda marcha, por mais longa e importante que seja, começa com o primeiro passo. Dar esse primeiro passo, às vezes, exige grande determinação, esforço monumental e muita coragem. Principalmente se o passo foi em direção a um caminho desconhecido, com o qual não estamos acostumados a lidar, por conta dos vícios adquiridos.

Dedico este trabalho:

À minha querida mãe, Maria Aparecida Ipolito Meneguette, pelo amor, incentivo e dedicação, sempre acreditando no meu sucesso.

Ao meu pai, Cícero Meneguette, certamente orgulhoso por mais um importante passo na minha vida.

Aos meus irmãos, que sempre torceram para que tudo desse certo.

À Tania, minha namorada, pela paciência nos momentos mais difíceis.

Ao meu querido amigo Adinovam Pimenta, ao qual tenho um grande apreço.

Agradecimentos

A Deus, acima de tudo e todos, por todas as oportunidades que tive e ainda terei na minha vida;

A meus pais (Cícero e Cidinha), meus irmãos, pessoas sem as quais eu não seria o que sou hoje;

À minha namorada, Tania, pela paciência durante os inúmeros finais de semana e feriados os quais passei a frente do computador;

À prof^a. Dr^a. Regina Borges de Araújo, pela motivação, encorajamento, amizade e direção durante os meus dois anos de estudos na pós-graduação.

A todos os meus familiares pela confiança irrestrita;

Aos meus amigos verdadeiros, que sempre estiveram ao meu lado, mesmo quando distantes;

A todos meus colegas do *LRVNet*, pelas discussões que contribuíram para a realização de nossos trabalhos.

Aos professores do Departamento de Computação da UFSCar, principalmente aos que tive a oportunidade de conhecer durante meus dois anos de estudos;

Resumo

Os ambientes virtuais colaborativos (AVCs) são ambientes sintéticos 3D compartilhados por múltiplos usuários localizados remotamente, com um objetivo comum, que acessam o ambiente por meio de avatares, representações gráficas do usuário. Os AVCs vêm ganhando campo em dispositivos móveis, como dispositivos de mão, celular, entre outros, através de jogos e simulações, visando o entretenimento e o conforto do usuário participante. Esses ambientes virtuais requerem uma baixa latência e uma alta escalabilidade para que um grande número de participantes possa interagir em um ambiente consistente e seus usuários não percam a sensação de imersão. Com isso, exige-se uma preocupação maior na estratégia de conexão entre os participantes. Esse cuidado aumenta quando esses ambientes são executados em veículos em movimento, já que as características singulares de uma rede ad hoc veicular (VANETs), como alta mobilidade dos nós, alta velocidade de transmissão e uma topologia altamente dinâmica, requerem uma estrutura complexa que possa superar essas limitações. Essas características nos levam aos desafios da comunicação entre os participantes e à disseminação do resultado das ações de seus *avatares* no AVC e/ou mudança de estado de um objeto para todos os participantes. A rede de suporte à comunicação deve suportar a entrada de um número crescente de participantes no AVC sem que isso afete a sensação de presença desses usuários.

De modo a superar os desafios de escalabilidade e garantir uma visualização uniforme do mundo virtual para todos os usuários em redes VANETs, esse trabalho apresenta o projeto e avaliação de uma rede *overlay*, mais precisamente uma rede *peer-to-peer* (p2p), que visa prover conectividade entre os usuários participantes de um mesmo AVC, além da atualização de estado de seus *avatares* e de objetos no AVC para todos os participantes, garantindo que essas informações sejam entregues no menor tempo possível.

Abstract

Collaborative Virtual Environments (CVEs) are synthetic 3D environments shared by multiple remotely located users with a common goal: the access to the environment through avatars (graphical representations of the user). CVEs are becoming popular on mobile devices such as handheld devices and cell phones, through games and simulations for users entertainment and comfort. These virtual environments require low latency and high scalability so that a large number of participants can interact in a consistent way without losing the immersion feeling. Therefore it entails a major concern on participants' connection strategies, especially when these environments are implemented on moving vehicles, because the unique characteristics of a vehicular ad hoc network (VANETs) like high mobility of nodes, high speed transmission and a highly dynamic topology demand a complex structure to overcome these limitations. These characteristics lead us to communication challenges among participants and to the dissemination of actions results of their avatars in the CVE and / or the state change of an object for all participants. The communication support network should be able to cope with the entrance of an increasing number of participants in the CVE without affecting these users feeling of presence. In order to overcoming the challenges of scalability and ensuring a uniform view of the virtual world for all users in VANETs networks, this paper presents a design and evaluation of an overlay network, specifically a peer-to-peer (p2p) network, which aims at providing connectivity between users of a CVE, in addition to updating the status of avatars and objects in the CVE for all participants, ensuring that information is delivered in the shortest time possible.

Lista de Figuras

FIGURA 2-1 - AMBIENTE EDUCACIONAL EM AVC.....	6
FIGURA 2-2 - TREINAMENTO PARA EMERGÊNCIA USANDO AVC.....	7
FIGURA 2-3 - SHOPPING VIRTUAL	8
FIGURA 2-4 - COMUNIDADE VIRTUAL	9
FIGURA 3-1 - EXEMPLO DE UMA REDE <i>OVERLAY</i> (ADAPTADO DE [ANARG, 2009]) (A) REDE VIRTUAL FORMADA DE UMA REDE FÍSICA (REDE <i>OVERLAY</i>) (B) REDE FÍSICA.	18
FIGURA 3-2 - UMA ABSTRAÇÃO DA ARQUITETURA DE REDE P2P (ADAPTADO DE [LUO ET AL., 2004]).....	19
FIGURA 3-3 - ROTA DE BUSCA PARA UM MODELO CAN	22
FIGURA 3-4 - P2P MODELO CENTRALIZADO	23
FIGURA 3-5 - P2P MODELO DESCENTRALIZADO	24
FIGURA 3-6 - REDE GNUTELLA	25
FIGURA 3-7 - MODELO P2P NÃO ESTRUTURADO HÍBRIDO	26
FIGURA 3-8 - TOPOLOGIA DA REDE GNUTELLA VERSÃO 0.6 (ADOTADO DE [XIE ET AL. 2008]).....	27
FIGURA 4-1 - COMUNICAÇÃO DE REDES AD HOC VEICULARES (ADAPTADO DE [STERLING, 2009])	30
FIGURA 4-2 - BANDA DSRC NORTE AMERICANA ADAPTADO DE [ARMSTRONG, 2009]	36
FIGURA 4-3 - ESTRUTURA DE DADOS DA CAMADA FÍSICA DSRC (ADOTADO DE[YIN, AT ALL, 2004]).....	38
FIGURA 4-4 - FUNCIONAMENTO DO PROTOCOLO GPSR.....	44
FIGURA 4-5 ESTRATÉGIAS DO PROTOCOLO GPCR (ADOTADO DE [LIE E WANG,2007]).....	45
FIGURA 4-6 – MODELO DE ROTEAMENTO EM <i>CLUSTER</i> (ADOTADO DE [LIE E WANG,2007])	47
FIGURA 4-7 – SOLUÇÃO DADA POR SANTOS E COLABORADORES (ADOTADO DE [SICHITIU E KIHIL, 2008]).....	48
FIGURA 4-8 - DIFERENTES PROTOCOLOS DE ROTEAMENTO (ADAPTADO DE [LI E WANG, 2007])	53
FIGURA 5-1 - DIVISÃO DO AMBIENTE VIRTUAL EM HEXÁGONOS IGUAIS	57
FIGURA 5-2 - INTERAÇÃO ENTRE A APLICAÇÃO (SIMULAÇÃO) A REDE <i>OVERLAY</i> E AS CAMADAS DE REDE	59
FIGURA 5-3 -CENA DE JOGO DUKEBEANEM	60
FIGURA 5-4 - DIAGRAMA DE SEQÜÊNCIA DO FUNCIONAMENTO DA REDE <i>OVERLAY</i>	62
FIGURA 5-5 - DIAGRAMA DE SEQÜÊNCIA PARA A VERIFICAÇÃO DE NÓS ATIVOS E INATIVOS NA REDE.	63
FIGURA 5-6 - (A) MENSAGENS ORIGINAIS DA REDE GNUTELLA; (B) MENSAGENS MODIFICADAS BASEADAS NA REDE GNUTELLA.....	64
FIGURA 5-7 -MENSAGEM PING_AVC.....	65
FIGURA 5-8 - MENSAGEM PONG_AVC.....	65
FIGURA 5-9 - MENSAGEM QUERY_AVC	66
FIGURA 5-10 - MENSAGEM QUERYHIT_AVC	67
FIGURA 5-11 - ROTEAMENTO GNUTELLA MODIFICADA	67
FIGURA 5-12 - GRÁFICO DA LATÊNCIA GERADA PELA SIMULAÇÃO.....	74
FIGURA 5-13 - GRÁFICO DA INCONSISTÊNCIA ESPAÇO-TEMPORAL DA VISUALIZAÇÃO DO AMBIENTE VIRTUAL.74	
FIGURA 5-14 GRÁFICO DA LATÊNCIA GERADA PELA SIMULAÇÃO EM FUNÇÃO DA DIFERENÇA DO POSICIONAMENTO REAL E SIMULADO.....	75
FIGURA 5-15 - GRÁFICO DA INCONSISTÊNCIA ESPAÇO-TEMPORAL PELA SIMULAÇÃO EM FUNÇÃO DA DIFERENÇA DO POSICIONAMENTO REAL E SIMULADO.	76

Índice de Tabelas

TABELA 4-1 - COMPARAÇÃO VANET VS REDES SEM FIO TRADICIONAL.....	33
TABELA 4-2 - COMPARAÇÃO PADRÃO 802.11 Vs ADHOC MAC	39
TABELA 4-3 - RESULTADOS DE QUE AVALIARAM OS PROTOCOLOS DE TRANSPORTE [SICHITIU E KIHL, 2008] ..	54
TABELA 5-1 - PARÂMETROS DE SIMULAÇÃO DA APLICAÇÃO.....	71
TABELA 5-2 - PARÂMETROS DA SIMULAÇÃO DE REDE VANET	72

Lista de Abreviaturas e Siglas

AODV	Ad Hoc on demand Distance Vector
AOI	Área de Interesse
A-STAR	Anchor-based Street e Tráfego Aware Routing
ASTM	American Society for Testing and Material
AVC	Ambiente Virtual Colaborativo
CA	Collision Avoidance
COIN	Clustering for Open IVC Networks
COVEN	Collaborative Virtual Environments
CSMA	Carrier Sense Multiple Access
DR	Dead Reckoning
DSDV	Destination-Sequenced Distance-Vector Routing Algorithm
DSR	Dynamic Source Routing
DSRC	Dedicated Short Range Communications
FCC	Federal Communications Commission
FDM	Frequency-Division Multiplexing

FRS	Fisheye State Routing
GPCR	Greedy Perimeter Coordinator Routing
GPSR	Greedy Perimeter Stateless Routing
GSR	Geographic Source Routing
ICI	Inter Carrier Interference
IEEE	Institute of Electrical and Electronics Engineers
LAR	Location-Aided Routing
MAC	Media Access Control
MANET	Mobile ad hoc Networks
MASSIVE	Model, Architecture and System for Spatial Interaction in Virtual Environments
OFDM	Orthogonal Frequency Division Multiplex
P2P	Peer-to-Peer
RR-ALOHA	Reliable R-ALOHA
RV	Realidade Virtual
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access

TORA	Temporally-Ordered Routing Algorithm
TTL	Time-To-Live
UDP	User Datagram Protocol
UMB	Urbano Multi-Hop Broadcast protocolo
UTRA-TDD	UMTS Terrestrial Radio Access Time Division Duplex
VANET	Vehicular Ad Hoc Network
VLNET	Virtual Life Network
VTP	Protocolo de Transporte Veicular
WAP	Wireless Access Point
WAVE	Wireless Access in Vehicular Environments
SUMO	Simulation of Urban MObility
NS2	Network Simulator
Jist	Java in Simulation Time
SWANS	Scalable Wireless Ad hoc Network Simulator
STRAW	STreetRAndom Waypoint
ZOR	Zona Relevante
ZRP	Zone Routing Protocol

Sumário

1	Introdução.....	1
1.1	Motivação e Objetivos.....	3
1.2	Organização	4
2	Ambientes Virtuais Colaborativos.....	5
2.1	Aplicações de Ambientes Virtuais	5
2.2	Características de um Ambiente Virtual.....	9
2.3	Desafios de um Ambiente Virtual	10
2.4	Técnicas para a diminuição de dados na rede.....	12
2.4.1	Área de interesse.....	13
2.4.1.1	Área de Interesse Baseada na proximidade	13
2.4.1.2	Área de Interesse Baseado na Visibilidade.....	14
2.4.2	Dead Reckoning	14
2.5	Sistemas de Ambientes Virtuais	15
2.6	Considerações Finais	17
3	Redes <i>Overlay</i>	18
3.1	Redes <i>Peer-to-Peer</i>	19
3.1.1	Redes P-2-P Estruturadas	21
3.1.2	Redes P-2-P Não Estruturadas.....	22
3.1.2.1	Centralizada	22
3.1.2.2	Descentralizada.....	24
3.1.2.3	Híbrida	26
3.1.3	Desafio do modelo Peer – to – Peer	27
3.2	Considerações Finais	29
4	Redes <i>Ad Hoc</i> Veiculares	30
4.1	Características de uma rede veicular	31
4.2	Aplicações	34
4.3	Tecnologias Envolvidas.....	35
4.3.1	Camada Física	36
4.3.2	Camada de Rede.....	40

4.3.2.1	Rotemaneto Ad Hoc	41
4.3.2.2	Roteamento Baseado no posicionamento geográfico	42
4.3.2.3	Roteamento Baseado em Cluster	47
4.3.2.4	Rotemaneto Broadcast	49
4.3.2.5	Roteamento Geocast	51
4.3.3	Camada de transporte	53
4.4	Considerações Finais	55
5	Projeto de um Suporte <i>Overlay</i> para Ambientes Virtuais Colaborativos sobre Redes Veiculares	56
5.1	Estrutura dos Ambientes Virtuais Colaborativos	57
5.1.1	Utilização do Jogo DukeBeanEm como AVC integrado à rede overlay 59	
5.2	Projeto de Rede <i>Overlay</i> de suporte a AVCs em redes Veiculares	63
5.3	Descrição do funcionamento da Rede Overlay Proposta (Gnutella modificada) 60	
5.4	Simulador da Rede	67
5.5	Experimentos de Simulação	69
5.5.1	Métricas Utilizadas	69
5.5.2	Cenário de Simulação	70
5.5.3	Resultados Obtidos	73
5.6	Considerações Finais	76
6	Conclusões	78
6.1	Contribuições Geradas	78
6.2	Trabalhos Futuros	79
6.3	Conclusões Finais	78
7	Referências Bibliográficas	80

1 Introdução

Os ambientes virtuais colaborativos (AVCs) permitem que participantes geograficamente dispersos interajam entre si em um ambiente virtual sintético, criado pelo computador, a fim de atingir um objetivo comum [Boukerche et al., 2005]. Vários ambientes virtuais colaborativos foram desenvolvidos para diferentes tipos de aplicação, tais como: combates militares [Carlsson e Hagsand, 1993][Michael et al., 1994], jogos *multiplayer* [Smallworld, 2009, EverQuest, 2009], entre outros. O objetivo desses ambientes é criar uma representação comum e consistente de um mundo virtual entre todos os participantes. Quando uma ação é realizada por um *avatar* (representação do participante no ambiente virtual) ou o estado de uma entidade do mundo virtual se altera, todos os participantes devem ser capazes de visualizar essas modificações. Entretanto, devido às características da rede, tais como largura de banda, condições de tráfego, atraso das mensagens, etc., a consistência da visualização do ambiente virtual não é automaticamente garantida para todos os participantes [Roberts e Wolfl, 2004].

Há duas situações que podem causar inconsistência em ambientes virtuais que podem impactar na percepção do mundo virtual [Zhou et al., 2004]. Uma delas é a violação de ordem das causas que está relacionada com a ordem de acontecimentos dos eventos. Um exemplo é: “*só ocorrerá uma explosão se tiver fogo no local*”, ou seja, uma bomba não pode explodir antes de haver fogo. A outra situação é a inconsistência de espaço-temporal que leva em consideração o tempo de envio da mensagem de atualização do ambiente e o seu tempo de chegada, bem como o próprio tempo da simulação [Zhou et al., 2004].

Quanto menor for o tempo de inconsistência espaço-temporal do ambiente virtual, menor será a probabilidade de o usuário perceber alguma anomalia no ambiente [Roberts e Wolff, 2004].

A inconsistência de visualização afeta a interação entre os participantes de uma simulação, por exemplo, dando vantagens para alguns participantes sobre outros em jogos de combate. Esse problema é agravado quanto há um número maior de usuários participantes, pois isso gera uma quantidade maior de mensagens que, conseqüentemente, exige maior capacidade da rede [Rueda et al., 2007].

Por outro lado, redes *ad hoc* veiculares (VANETs) tendem a se tornar populares na medida em que automóveis e infraestrutura de vias integram-se cada vez mais às novas tecnologias de sensores e redes de sensores sem fio, comunicação sem fio e crescente poder computacional [Olariu, 2007]. A execução de AVCs em redes com características singulares, tais como as redes VANETs podem propiciar aos usuários, enquanto em automóveis em movimento, diferentes e inovadoras aplicações, de jogos *multiplayers* a simulações de treinamento e educacionais. Porém, o suporte a AVCs em redes VANETs apresenta grandes desafios, já que essas redes possuem características diferentes de outras redes sem fio tradicionais, tais como [Toor et al., 2008]: velocidade muito alta de transmissão devido ao curto tempo em que os veículos tem para transmitir um dado, pois o veículo pode atingir velocidades altas e limitação da mobilidade geográfica dos nós, isto é, o nós só percorrem ruas, avenidas e rodovias (vias públicas), etc. Estas características levam a desafios na comunicação dos participantes e na disseminação do resultado das ações de seus *avatares* no AVC e/ou mudança de estado de um objeto para todos os participantes. A rede teria que suportar a entrada de um número crescente de participantes no AVC sem que isso afetasse a sensação de presença desses usuários.

De modo a superar os desafios de escalabilidade e inconsistência espaço/temporal na visualização de aplicações de AVCs em redes VANETs, este trabalho apresenta o projeto e avaliação de uma rede *overlay*, mais precisamente uma rede *peer-to-peer* (p2p), que visa a prover conectividade entre os usuários participantes de um mesmo AVC, além da atualização de estado de seus *avatares* e de objetos no AVC para todos os participantes, garantindo que essas informações sejam entregues no menor tempo possível. Um dos objetivos de uma rede *overlay* é abstrair a complexidade das camadas inferiores de rede para a aplicação [Stemm e Katz, 1998]. Redes *peer-to-peer overlay* possuem, em sua

essência, características similares às de uma rede veicular, tais como topologia dinâmica e auto-organização [Lee et al., 2007], [Lee et al., 2006], [Ahmed et al., 2006]. Diferentemente de outros modelos de rede, as redes *peer-to-peer* não degradam com a sobrecarga de mensagens como em um modelo cliente/servidor [Singh et al., 1994] [Pandžić et al., 1997], tampouco com o tempo de migração de um servidor para outro em um modelo de multi-servidor [Das et al., 1997], o que pode propiciar escalabilidade para AVCs em Vanets.

1.1 Motivação e Objetivos

As Redes *Ad Hoc* Veiculares possuem características singulares, tais como, alta mobilidade dos nós, uma alta transmissão de dados e frequente fragmentação em sua rede, agregando com as características de latência e escalabilidade de um ambiente virtual. Portanto, o desenvolvimento de um ambiente virtual sobre a Rede *Ad Hoc* Veicular, torna-se um desafio.

Ferramentas estão sendo desenvolvidas para o compartilhamento de arquivos e multimídias em redes veiculares, mas ainda há poucos trabalhos na literatura com enfoque para jogos e simulações em redes veiculares que utilizam ambientes virtuais, porém essas aplicações são bem consolidadas em redes fixas. Uma das classes de aplicações que nos inspirou para a realização deste trabalho é a de simulações de ambientes virtuais colaborativos para o treinamento de equipes no gerenciamento de emergências. AVCs para o gerenciamento de emergências poderiam ser acessados a partir de veículos de bombeiros, por exemplo, ao se dirigirem para o local alvo da emergência. A simulação da resposta a emergência em um AVC que reflete o ambiente alvo da emergência antes mesmo dos veículos chegarem a este alvo poderá aprimorar as tomadas de decisões das equipes envolvidas na simulação com o objetivo de reduzir perdas de vida e de patrimônio.

Esse trabalho tem como objetivo projetar, implementar e avaliar uma rede *overlay* que dará suporte a ambientes virtuais colaborativos em Redes *Ad Hoc* Veiculares. Como resultado desse trabalho, diminuiremos o tempo de entrega das mensagens de atualização

entre os participantes de um ambiente virtual colaborativo sem a necessidade de infraestrutura externa.

1.2 Organização

Este documento está organizado da seguinte maneira: O Capítulo 2 discute os principais conceitos de ambiente virtual colaborativo, características, desafios, aplicações e alguns ambientes colaborativos existentes. No Capítulo 3 são apresentados conceitos de redes *overlay* focando nas redes *peer-to-peer*, seus modelos, seus desafios. No capítulo 4 o conceito de Redes *Ad Hoc* Veiculares é introduzido bem como as características e tecnologias envolvidas para a sua implementação. No capítulo 5 é apresentado o projeto de uma rede *overlay* para ambientes virtuais colaborativos em Redes *Ad Hoc* Veiculares, que inclui sua especificação, implementação, simulação e avaliação de desempenho. O capítulo 6 apresenta as conclusões deste trabalho e propostas de trabalhos futuros, seguido de Referências Bibliográficas.

2 Ambientes Virtuais Colaborativos

Ambientes Virtuais Colaborativos (AVCs) são sistemas que, com interpretação de contexto realizada por computador, suportam ativamente a colaboração e a comunicação humana [Joslin et al., 2003].

2.1 Aplicações de Ambientes Virtuais

Ambientes Virtuais Colaborativos podem beneficiar uma infinidade de aplicações, como ambientes educacionais, treinamentos, lojas virtuais, comunidades entre outros. Essa seção apresenta alguns exemplos de aplicações de AVCs.

- **Ambientes Educacionais**

Nesses ambientes, o professor pode iniciar a aplicação e os estudos conectando-se ao ambiente virtual de maneira local ou remota. Tais ambientes podem oferecer livre acesso a todos os usuários ou ser restritos aos alunos da disciplina. O professor poderá guiar os alunos ao interior do ambiente, explicando detalhes importantes, ou auxiliá-los nas dificuldades encontradas. A comunicação entre professores e alunos pode ser feita via fórum, vídeo, áudio ou outros meios. Os estudantes podem gravar a aula ou realizar novas anotações para futura consulta. A Figura 2-1 mostra um exemplo de AVC para ambientes educacionais.



Figura 2-1 - Ambiente Educacional em AVC

- **Treinamentos Colaborativos**

Esses ambientes colaborativos oferecem meios para alterar o objeto no interior de um ambiente, como também para criar outros objetos. Todos os participantes têm a mesma capacidade dentro do ambiente e podem ou não ter os mesmos privilégios. Bons exemplos dessa classe de aplicação são: treinamentos para emergência e treinamentos médicos.

Os treinamentos para emergências possuem diversas aplicabilidades como treinamento do corpo de bombeiro, polícia e militares, além de ações em casos de desastres naturais, entre outros. Neste tipo de aplicação, os objetos, que não são controlados por um usuário, têm grande participação na simulação, sendo assim, são menos passivos que os mesmos tipos de objetos em aplicações de ambientes educacionais, por exemplo. Na figura 2-2 pode-se ver um ambiente de treinamento militar para controle de guerras civis



Figura 2-2 - Treinamento para Emergência usando AVC

- **Shopping Virtual**

Um usuário pode entrar em uma loja virtual e interagir com produtos que estejam à venda. Caso ele queira mais detalhes sobre o produto, poderá assistir a algum vídeo, ler um texto ou mesmo escutar as declarações dos clientes que já compraram aquele produto. Ele pode ainda chamar um vendedor, que é representado por um outro *avatar* ou um objeto simulado.

A comunicação entre o cliente e o vendedor poderá ser feita através de texto, áudio ou vídeo. Na Figura 2-3, pode-se ver um exemplo de um shopping virtual utilizando AVC.



Figura 2-3 - Shopping Virtual

- **Comunidades Virtuais**

As comunidades virtuais diferem das aplicações de ambientes virtuais colaborativos pelo fato de os ambientes existirem ao longo de um período determinado e de as aplicações de forma permanente. Pelo fato de as comunidades virtuais poderem ser habitadas por uma grande quantidade de usuários, verdadeiras cidades podem ser formadas com a existência de prefeito, polícia e centro de informações, tal quais cidades reais. Cada usuário pode ter um espaço próprio para realizar alterações, conforme as leis prescritas na “cidade virtual”. O *Second Life* é o exemplo mais difundido desse tipo de aplicação. A Figura 2-4 mostra um exemplo de uma comunidade virtual.



Figura 2-4 - Comunidade Virtual

2.2 Características de um Ambiente Virtual

AVCs apresentam as seguintes características [Singhal e Zyda, 1999]:

- **Compartilhamento de espaço:** todos os participantes têm a ilusão de estarem localizados em um mesmo espaço que, por sua vez, apresenta as mesmas características para todos os participantes;
- **Presença compartilhada:** Cada participante é representado dentro do ambiente virtual por um objeto geométrico, denominado *avatar*, e pode ver as representações dos outros usuários que estejam localizados no mesmo ambiente compartilhado;
- **Compartilhamento de tempo:** um participante deve ser capaz de ver o comportamento dos outros no momento em que isto ocorre, ou seja, o ambiente virtual deve suportar interação em tempo real;

- **Comunicação:** o sistema deve suportar a comunicação entre os participantes para manter, entre outros aspectos, a consistência do ambiente entre todos os usuários participantes.

Segundo o projeto proposto pela *Living World* [Living World, 1997], os conjuntos de ações que um usuário pode realizar em um AVC são:

- Modificações na cena, que podem ser iniciadas por qualquer cliente;
- Modificações que devem, potencialmente, ser sincronizadas em todos os clientes;
- Todas as modificações, cada objeto, ou é responsável por refletir atualização localmente e enviar mensagens de sincronização para as instâncias deste mesmo objeto em todas as máquinas de usuários participantes, ou é um receptor destas mensagens de sincronização de outros usuários participantes e deve realizar a ação apropriada;
- Nem todas as modificações devem obrigatoriamente ser realizadas no instante em que são recebidas; uma otimização de desempenho depende de saber-se qual objeto tem maior probabilidade de ser alterado posteriormente;
- Nem todas as notícias de modificação precisam ser comunicadas; a otimização depende também de ter-se o conhecimento de quem necessita saber o quê e como.

2.3 Desafios de um Ambiente Virtual

Um dos maiores desafios encontrados nos AVCs envolve a preservação da consistência do ambiente entre todos os participantes [Zhou et al., 2004]. Este requisito consiste em garantir que cada terminal compartilhando a mesma simulação seja capaz de trocar dados de atualização do mundo virtual, de modo consistente com os demais participantes, para garantir que seja apresentado exatamente o mesmo ambiente a todos os usuários.

Segundo Zhou e colaboradores [Zhou et al., 2004] a inconsistência em um ambiente virtual pode ocorrer de duas formas:

- **Violação de ordem das causas** - é a violação da ordem dos acontecimentos dentro de um ambiente virtual. Um exemplo é: “*só ocorrerá uma explosão se tiver fogo no local*”, isso é uma bomba não pode explodir antes de haver fogo.
- **Inconsistência de espaço-temporal** – é a inconsistência da visualização dos estados dos objetos em ambiente virtual, ou seja, em um determinado tempo da simulação alguns participantes visualizam um mesmo objeto de forma diferente causando uma inconsistência entre os participantes. Essa diferença está diretamente ligada ao tempo de envio de mensagem de atualização do ambiente e o seu tempo de chegada, bem como o próprio tempo da simulação [Zhou et al., 2004].

Aplicações desse tipo são mais eficientes quando podem usufruir de uma conectividade que possibilite comunicação em grupo, ou seja, que permita que um grande número de usuários possa interagir em um mesmo mundo virtual sem afetar a consistência do ambiente.

A escalabilidade é talvez o ponto mais difícil a ser atingido em AVCs, pois a cada novo participante no ambiente virtual, haverá demanda de mais recursos computacionais como o poder de processamento da máquina responsável pelo *input* e *output* do ambiente e pela leitura e escrita para a rede; dos dispositivos gráficos responsáveis por renderizar as imagens; bem como a rede que lida com a transmissão e o transporte dos dados [Faisstnauer, 2001].

Além dos requisitos de consistência, conectividade e escalabilidade apresentados anteriormente, são também requisitos de AVCs, segundo Boukerche e Araújo [Boukerche et al., 2004]:

- **Persistência:** O estado do sistema tem que ser mantido, independente se o usuário está presente no ambiente ou não;
- **Reusabilidade:** Deve ser possível que elementos de outros ambientes possam ser reutilizados/combinados;
- **Interoperabilidade:** Definições e padrões devem ser utilizados para permitir a interação entre vários sistemas heterogêneos;
- **Adaptabilidade:** Um grande número de usuários conectados a partir de redes e dispositivos heterogêneos deve ser suportado;
- **Segurança:** um usuário não se sobressair sobre outro devido às falhas do sistema.

2.4 Técnicas para a diminuição de dados na rede

Um ambiente virtual colaborativo depende da rede para a troca de informações (mediante o envio e recebimento de mensagens), mas a utilização de uma rede sempre introduz um atraso [Faisstnauer, 2001]. Esse atraso afeta diretamente a consistência da visualização do ambiente físico [Zhou et al., 2004].

Uma das formas de se diminuir o atraso na entrega das mensagens é aumentar a vazão da rede através da diminuição de informações que estão sendo trafegadas pela rede, ou seja, transmitir somente as informações altamente necessárias. Para isso, existem algumas técnicas tais como:

2.4.1 Área de interesse

A área de interesse está entre as abordagens mais populares para reduzir a carga sobre os recursos de rede. Basicamente, este recurso tenta reduzir a quantidade de informação que é gerada, processada e transmitida, filtrando a informação que não é de interesse para o destinatário (os objetos que são invisíveis, por exemplo). Como cada participante é geralmente centrado apenas em uma pequena porção do ambiente, chamada "Área de Interesse" (AOI), com isso diminuimos a quantidade de recursos exigidos da máquina, bem como a quantidade de mensagens enviada na rede, pois o envio dos dados destina-se apenas às entidades visíveis, ou seja, entidades que estão compartilhando a mesma porção do ambiente virtual, como podemos ver pelos trabalhos [Greenhalgh e Benford, 1997][Boukerche et al., 2004].

2.4.1.1 Área de Interesse Baseada na proximidade

Nessa técnica o interesse em um determinado objeto é dado pela distância em relação ao observador. O ambiente é tipicamente dividido em regiões, e cada participante recebe atualizações somente daqueles objetos que estão nas regiões imediatamente adjacentes à sua localização. A subdivisão pode ser arbitrária ou não (por exemplo, utilizando uma grade regular de células hexagonais 2D, como na NPSNET [Macedonia et al., 1995], COMUGO [Boukerche et al., 2008].

Todos os objetos subscritos em uma região transmitem a posição e as atualizações do ambiente. Isso permite que todos os participantes fiquem informados sobre as ações que estão ocorrendo em sua região de interesse.

2.4.1.2 *Área de Interesse Baseado na Visibilidade*

Diferentemente da técnica anterior, esse método leva em consideração somente a faixa de visibilidade do participante, ou seja, o participante só enviará informações para os usuários que estão em sua faixa de visibilidade. Isto leva a um estado inconsistente da cena entre os diversos usuários, porém esta inconsistência não pode ser percebida pelos usuários, pois ela afeta apenas objetos que estão fora do campo de visão.

A visibilidade pode estar pré-determinada ou estar fixada em um objeto, e ao mover-se o objeto também se move a câmera.

2.4.2 *Dead Reckoning*

Dead Reckoning é outro mecanismo que reduz a quantidade de mensagens na rede. No *Dead Reckoning* (DR) cada participante mantém um modelo de comportamento simplificado que gerenciará os objetos remotos. Esse modelo prediz o estado real desses objetos remotos, por extrapolação, a partir dos estados anteriores. Essa operação é conhecida como *Dead Reckoning*.

Esta técnica permite limitar a quantidade de mensagens de atualização do ambiente transmitidas na rede, que são utilizadas para corrigir a previsão, caso esta previsão seja imprecisa. Para detectar a imprecisão, todos participantes armazenam uma cópia adicional dos objetos simulados localmente, aos quais se aplicam as mesmas rotinas do *Dead Reckoning* também empregadas pelos outros participantes para a predição dos objetos. Sempre que as representações dos objetos locais e as cópias dos objetos diferem de um determinado limiar, essa diferença é susceptível a todos os participantes remotos, pois todos têm seu *Dead Reckoning*, e uma mensagem de atualização é transmitida para todos os participantes para corrigir os *Dead Reckoning*.

O *Dead Reckoning* é baseado na movimentação dos objetos que podem ser computacionalmente incrementados a partir de um conjunto limitado de valores (tais como a velocidade ou a aceleração), a fim de permitir que participantes remotos realizem uma previsão acurada. Portanto, como *Dead Reckoning* baseia-se numa previsão com base no comportamento recente dos objetos, os campos de aplicação em áreas com um comportamento imprevisível dos usuários, tais como jogos on-line, são muito limitados [Faisstnauer, 2001]. Além disso, a redução de mensagens da rede empregada pelo DR tem um custo no esforço computacional, pois os participantes têm de prever comportamento dos objetos remotos.

2.5 Sistemas de Ambientes Virtuais

Vários ambientes virtuais colaborativos foram desenvolvidos para diferentes tipos de aplicação, tais como: combates militares [Carlsson e Hagsand, 1993][Michael et al., 1994], comerciais, jogos *multiplayer*, shopping virtual, entre outros. Tais sistemas são desenvolvidos segundo um modelo cliente/servidor ou em *peer-to-peer*.

No modelo cliente/servidor há um computador central, ou um conjunto de computadores que recebem todas as mensagens de atualização dos participantes, processam e reenviam as atualizações para todos os usuários.

Muitos trabalhos têm sido propostos na literatura para uso comercial e acadêmico, tal como o BrickeNet [Singh et al., 1994] que proporciona uma técnica de filtragem, a fim de minimizar a quantidade de mensagens necessárias.

BrickeNet introduz uma estratégia para controlar objetos compartilhados. Em vez de compartilhar o ambiente virtual, cada nó administra uma cópia local do ambiente virtual, e seleciona um conjunto de objetos a serem compartilhados com outros usuários. Esta estratégia oferece ao usuário total privacidade mas, por outro lado, as atividades de

colaboração são afetadas, uma vez que os usuários não podem acessar todos os recursos do ambiente virtual.

VLNET [Pandžić et al.,1997] (*Virtual Life Network*) foi inicialmente distinguida por seus sofisticados gráficos 3D e seus protocolos de comunicação. No entanto, a principal limitação do sistema proposto é que com o crescente número de usuários, o servidor passa a demonstrar uma limitação devido a enorme quantidade de mensagens geradas pelo envio; a arquitetura multi-servidor foi desenvolvida como uma alternativa para superar a limitação do servidor.

NetEffect [Das et al., 1997] foi concebido para simplificar o desenvolvimento de redes baseadas em mundos virtuais. Um servidor mestre garante o balanceamento de carga entre o conjunto de servidores disponíveis por uma sobrecarga de clientes, migrando-se para outro servidor com menor carga de trabalho. No entanto, a arquitetura multi-servidor sofre um efeito colateral: a migração entre servidores pode ter um tempo maior de espera quando um usuário solicita um novo servidor.

A arquitetura *peer-to-peer* foi desenvolvida como uma solução alternativa ao modelo cliente/servidor. Nesse modelo, cada *avatar* atualiza suas informações, comunicando-se diretamente com todos os outros participantes da rede. Nesse modelo, não existe um servidor central, nem pontos de falha, uma vez que todos os nós desempenham papéis cliente e servidor. Algumas aplicações mais conhecidas que utilizam o modelo p2p são:

SCORE [Lety et al., 2004] é um protocolo de comunicação baseado em *multicast*. Foi projetado para suportar *múltiplos* grupos de *multicast* e de múltiplos agentes. O ambiente virtual é dinamicamente dividido em várias áreas ou células, a fim de oferecer um melhor gerenciamento do estado do mundo. Cada célula tem o seu próprio grupo *multicast*, e os *avatares* devem inscrever-se em um conjunto de grupos *multicast*, que se inserem, pelo

menos parcialmente, dentro de sua área de interesse. Para controlar o tráfego de rede, SCORE [Lety et al., 2004] permite diferentes tamanhos de células baseados na população da área.

COMUGO [Boukerche et al., 2008], desenvolvido essencialmente para redes MANETs (*mobile ad hoc networks*), é um sistema baseado em multicamadas *overlay*, em que a posição do nó é dada pelo perfil do usuário e de suas regras de conexão entre os participantes. Cada camada atua como um cluster de rede que abrange todos os participantes da mesma zona. COMUGO permite a criação de grupos *multicast* através da multicamada *overlay*, porém, a quantidade de cálculos para a realização da conexão entre os participantes, bem como as replicações de informações para a descoberta de nós para se conectar trazem um impacto significativo de energia nos dispositivos.

2.6 Considerações Finais

Nesse capítulo, foram apresentadas a definição de um ambiente virtual colaborativo, aplicações, características, desafios das AVCs, técnicas para diminuir o número de mensagens na rede, além de alguns ambientes de suporte existentes com suas arquiteturas de rede.

Importante ressaltar, no encerramento deste capítulo, alguns elementos referentes à escalabilidade e à latência em um ambiente virtual colaborativo: mesmo com um grande número de usuários, a latência tem que se manter aceitável para que o participante não perca a sensação de imersão.

No próximo capítulo serão apresentadas as definições de rede *overlay*, mais especificamente as redes *peer-to-peer* (p2p), suas camadas de protocolos, modelos de implementação e desafios de desenvolvimento.

3 Redes *Overlay*

As Redes *Overlay* formam uma estrutura de comunicação virtual estabelecida através de uma rede física [Doval, O'Mahony, 2003]. Em outras palavras, são redes lógicas construídas sobre uma rede física. A figura 3-1 apresenta um exemplo de Rede *Overlay*, criando uma rede lógica, onde os nós exercem enlaces lógicos com os demais através de conexões virtuais.

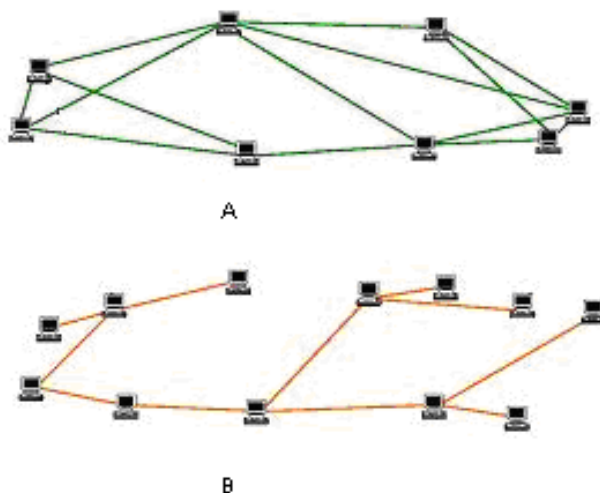


Figura 3-1 - Exemplo de uma rede *overlay* (adaptado de [ANARG, 2009]) (A) Rede virtual formada de uma rede física (rede overlay) (B) Rede física.

Aplicações dessas redes podem ser encontradas em redes *Peer-to-Peer* (P2P) [Freenet, 2009, Gnutella, 2009], serviços *multicast* [Shin & Turner, 2002], Rede de Distribuição de Conteúdo (CDN) [Gang, 2003] e *Resilient Overlay Networks* (RONs) [Andersen et. all, 2001].

Na seção a seguir serão apresentadas características, classificação, desafios e vantagens de se usar a rede *Peer-to-Peer* para se construir uma Rede *Overlay*.

3.1 Redes *Peer-to-Peer*

O *Peer-to-Peer* é um modelo de comunicação que possibilita que dois ou mais pares colaborem espontaneamente em uma rede, usando informações apropriadas e sem a necessidade de um coordenador central [Milojicic et al., 2002].

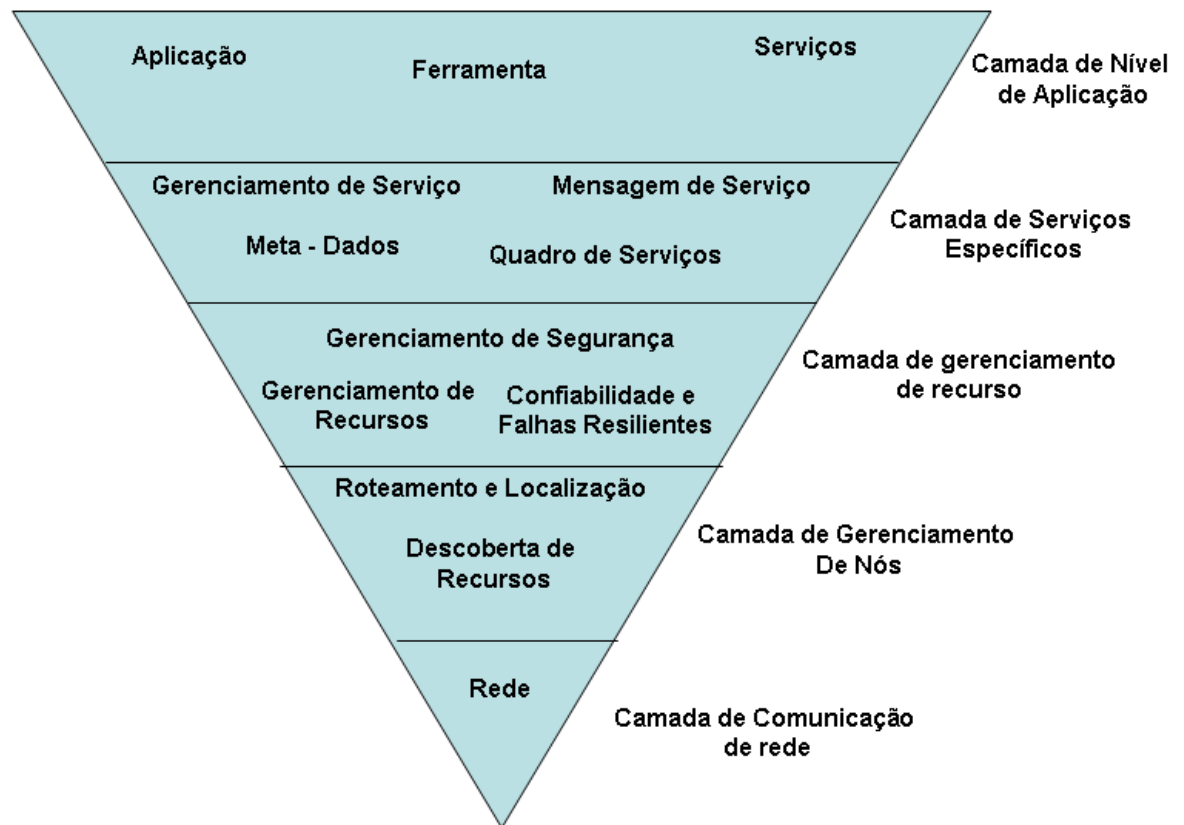


Figura 3-2 - Uma abstração da arquitetura de rede P2P (adaptado de [Luo et al., 2004])

A figura 3-2 mostra uma abstração da arquitetura de uma rede P2P tendo as seguintes camadas de abstração [Lua et al., 2005]:

- **Camada de comunicação de rede**

Descreve as características de rede de computadores: a Internet, pequenas redes sem fio ou nós sensores que estão conectados em um modo *Ad Hoc*.

- **Camada de gerenciamento dos nós**

Abrange a gestão dos seus pares, que inclui a descoberta de pares e algoritmos de otimização de roteamento.

- **Camada de gerenciamento de recursos**

Lida com a segurança, confiança, falha e aspectos da disponibilidade dos recursos agregados, que mantêm a robustez de sistemas P2P.

- **Camada de serviços específicos**

Apóia a infra-estrutura baseada no P2P e os componentes de aplicações específicas, através da sincronização de tarefas paralelas e de uma computação intensiva, permitindo o gerenciamento de arquivo e conteúdo.

O Meta-dado é um mecanismo que descreve o conteúdo armazenado em todos os pares P2P e dados de localização.

- **Camada de nível de aplicação**

Está preocupada com ferramentas, aplicações e serviços que são executados com funcionalidades específicas em cima da infra-estrutura baseada em P2P.

Existem dois modelos de se classificar as redes P2P, Estruturados e não estruturados. As seções seguintes descrevem os modelos existentes de comunicação P2P.

3.1.1 *Redes P-2-P Estruturadas*

Essa classe de rede P2P estruturada atribui chaves para os objetos (dados), e organiza seus *peers* em um grafo que mapeia cada chave. Essa estrutura de grafo permite uma descoberta eficiente de itens de dados usando as chaves dadas [Luo et al., 2004], porém esse modelo estruturado não suporta uma busca complexa [Luo et al., 2005], pois os armazenamentos são feitos através das chaves. Com isso, conteúdos similares podem estar em lugares totalmente diferentes na rede, uma vez que têm chaves diferentes.

Um exemplo desse tipo de modelo de rede p2p estruturado é o CAN.

- **CAN (*Content Addressable Network*)** [Falchi et al., 2007]: Essa rede utiliza um espaço bi-dimensional para mapear nodos e chaves. Quando o primeiro nó entra na rede, ele é responsável por todo o espaço. À medida que mais nos vão entrando, esse espaço vai sendo subdividido entre os nós de forma a fazer balanceamento. Cada parte de responsabilidade de um nó é chamada de zona, e nessas zonas, os objetos são mapeados. O mapeamento prossegue da seguinte forma: Um objeto é mapeado para uma chave K; a chave K é mapeada para um ponto P no espaço 2-dimensional; essa zona em que o ponto P foi mapeado é de responsabilidade de um nó N; Assim, o nó N fica responsável pelo objeto.

Para fazer consultas na rede, tem-se um procedimento com passos semelhantes à inserção. Na figura 3-3 um nó 17 fornece a chave para a rede K, para recuperar o objeto; essa chave é mapeada para um ponto P na rede; esse ponto P fica na zona Z que é de responsabilidade do nó 4; 17 envia uma

mensagem para 4 requisitando o objeto, mas como 17 não sabe o endereço de 4, então 17 envia para uma região vizinha que irá passar essa mensagem de região para região até chegar no nó 4 responsável pela região Z.

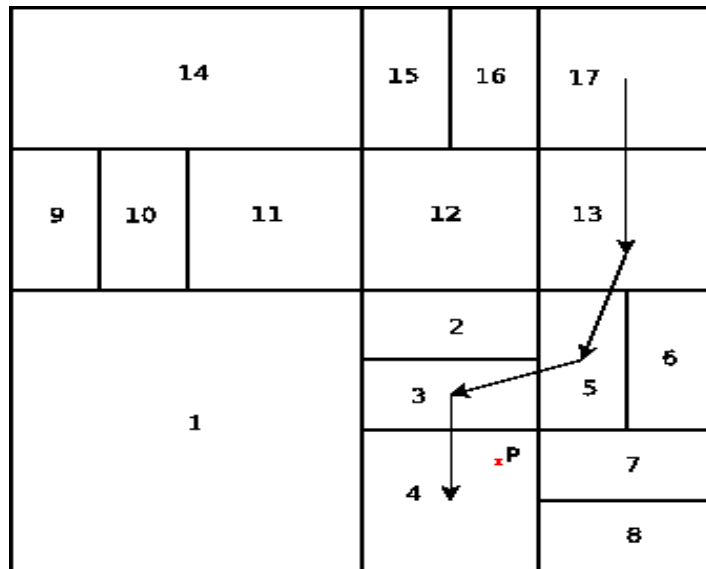


Figura 3-3 - Rota de busca para um modelo CAN

3.1.2 Redes P-2-P Não Estruturadas

O modelo *peer-to-peer* não estruturado organiza seus *peers* em um gráfico aleatório em uma forma plana ou hierárquica (exemplo com o uso dos *super-peer*), ou seja, um nó não tem uma posição pré-definida na rede. Esse modelo permite a realização de consultas mais complexas [Luo et al., 2005] e está subdividido em centralizado, descentralizado e híbrido.

3.1.2.1 Centralizada

No modelo P2P não estruturado centralizado, um servidor central é usado para tarefas como autenticação de usuário, serviço de diretório e mapeamento de recursos

disponíveis na rede. O princípio é que o *peer* contate algum servidor para iniciar a transmissão de algum dado, ou que comece a usufruir de algum serviço, como podemos ver na figura 3-4.

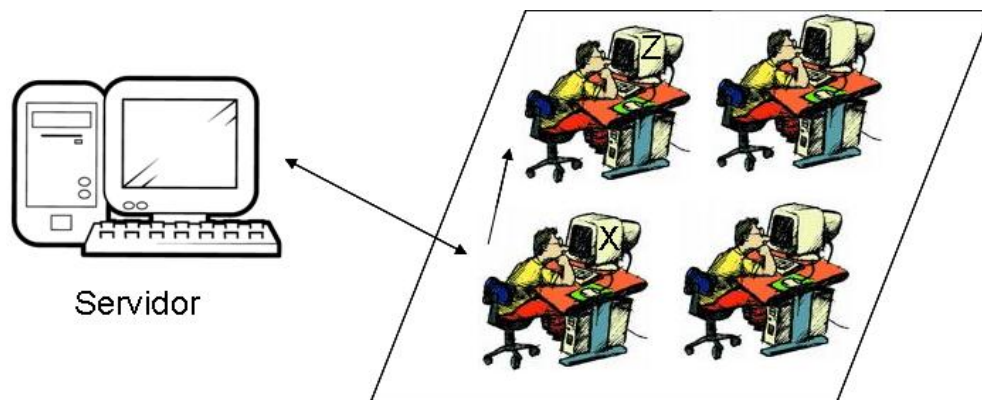


Figura 3-4 - P2P modelo Centralizado

A figura 3.4 mostra que o usuário “X” contata o servidor para saber quem tem um determinado arquivo ou informação; em seguida, o servidor devolve ao par inicial alguma informação pertinente, que permite a ele conectar-se diretamente com outro par “Z” e assim efetuar a transação. O exemplo mais conhecido dessa classe é a napster.

- **Napster**

Foi o primeiro sistema de compartilhamento de arquivos, no qual o nó autentica em um servidor e envia a lista de arquivos que ele pode oferecer.

Para efetuar uma busca, um *peer* envia os dados a um servidor, o servidor, por sua vez, consulta a sua base e aponta aos *peers* que possuem o arquivo desejado. Posteriormente, o *peer* inicia transferência dos dados diretamente dos nós, ou seja, nó a nó [Lua, et all, 2005].

3.1.2.2 Descentralizada

O modelo *Peer-to-peer* descentralizado, também chamado de P2P puro considerado por [Schollmeier, 2001], trabalha sem a ajuda de nenhum tipo de servidor ou repositório central de informações. Todos os *peers* são auto-suficientes e podem assumir simultaneamente as funções de cliente e servidor, chamados de *servents*, como podemos ver na figura 3-5, apresenta um modelo de P2P não estruturado descentralizado.

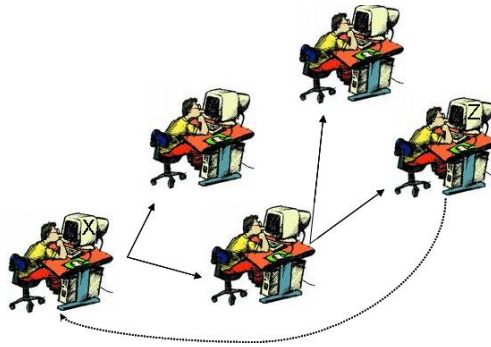


Figura 3-5 - P2P modelo descentralizado

Nesse modelo, os pares efetuam solicitações aos pares vizinhos que estão ao alcance, como demonstra a figura 3.5. O nó “X” procura nas máquinas com as quais tem conexão se esses nós possuem o dado especificado; essas máquinas se conectam com outros pares até que o dado seja localizado; nesse caso, a informação se encontra no nó “Z”. Quando o dado é localizado, é passado um apontador para o par inicial, que estabelece uma conexão direta com o par onde o dado foi encontrado, fazendo assim uma transação sem intermediários. Um exemplo desse modelo é a Gnutella [Gnutella, 2009].

- **Gnutella versão 0.4**

Gnutella versão 0.4 não utiliza um servidor central para rastrear todos os arquivos dos usuários [Gnutella, 2009]. Os usuários precisam de uma aplicação que promova a conexão com outros pares, formando uma rede espontânea sem controle centralizado, e

agindo tanto como cliente quanto servidor de conteúdo. O seu funcionamento será detalhado a seguir e é representado pela figura 3-6.

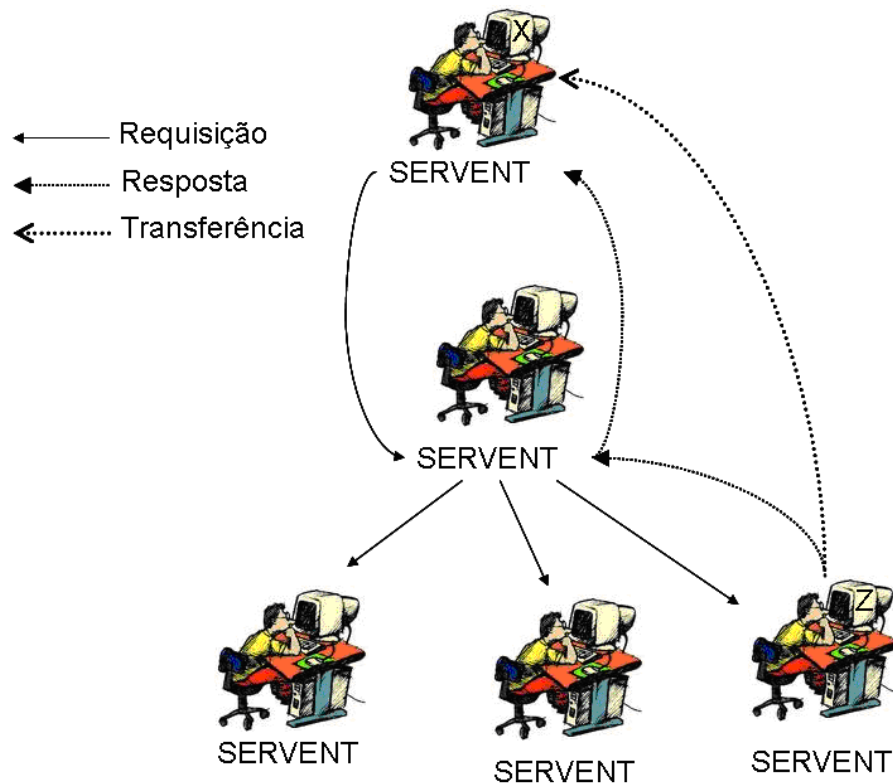


Figura 3-6 - Rede Gnutella

Para o compartilhamento de um determinado arquivo, um nó, aqui chamado de “X”, se conectará a outro computador na rede e avisará que está procurando um determinado arquivo. Se este computador não tiver o arquivo, ele comunicará a todos os computadores com quem está conectado sobre a requisição do nó “X”. Ao encontrar o arquivo, que nesse caso se encontra no nó “Z”, ele responde a requisição até chegar à origem que posteriormente iniciará a transferência.

Para que as mensagens não fiquem sendo retransmitidas eternamente, faz-se o uso de um mecanismo que limita a vida útil da mensagem na rede conhecido como tempo-de-vida (*time-to-live* – TTL).

3.1.2.3 Híbrida

O modelo peer-to-peer não estruturado híbrido possui a interação tanto de um modelo não estruturado centralizado, bem como de modelo descentralizado, onde alguns nós com maior capacidade de hardware irão atuar como servidores (chamados de super-peer ou ultra-peer) para um grupo de nós (chamados de “folhas”)[Stutzbach et al., 2008].

O Super-*peer* manterá a lista de arquivos dos nós conectados a ele. Quando um nó folha dispara uma consulta para um Super-*peer*, este irá verificar localmente, em sua base, se existe o dado requisitado. Caso não tenha, verificará com outros Super-*peers*. Na figura 3-7 podemos ver o modelo P2P não-estruturado híbrido. Os computadores vermelhos são os Super-*peer* e os computadores pretos são os nós folhas.

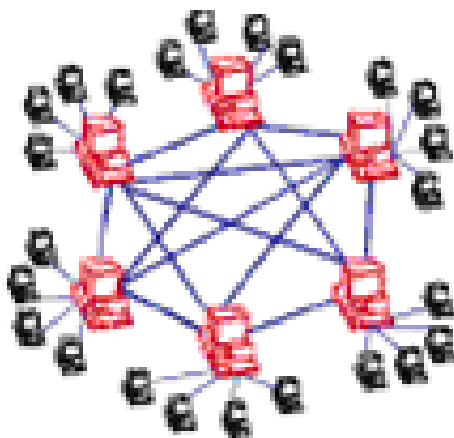


Figura 3-7 - Modelo P2P não estruturado híbrido

Um dos exemplos desse modelo é a versão mais recente da Gnutella versão 0.6 [Gnutella, 2009]

- **Gnutella 0.6**

Gnutella protocolo 0,6 [gnutella, 2009] emprega uma arquitetura híbrida combinando modelo centralizado e modelo descentralizado. *Servents* são categorizados em *leaf* e *ultrapeer*. Uma *leaf* mantém apenas um pequeno número de conexões com os *ultrapeers*. Um *ultrapeer* mantém ligações com outras *ultrapeers* e atua como um *proxy* para a rede Gnutella para o nó ligado a ela. Um *ultrapeer* só transmite uma consulta para uma *leaf* se acreditar que a *leaf* possa responder-lhe, e nunca retransmite consultas entre *ultrapeers* [Xie et al. 2008]. A Figura 3-8 ilustra a topologia da rede Gnutella 0.6. O Protocolo gnutella 0,6 é compatível com o protocolo 0.4.

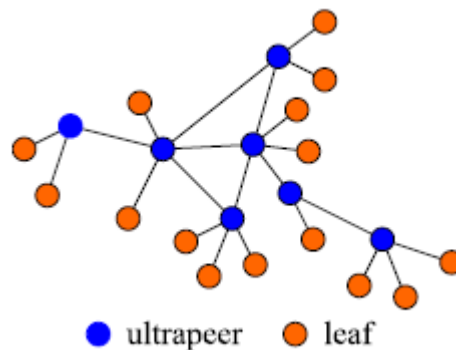


Figura 3-8 -Topologia da rede gnutella versão 0.6 (adotado de [Xie et al. 2008])

3.1.3 Desafios do modelo Peer – to – Peer

O modelo P2P, além de ter como desafio a aplicação em que os nós se comunicam de igual para igual, apresentam outras limitações, tais como a segurança, interoperabilidade, controle de acesso, integridade dos dados, sincronização de uma

aplicação entre várias máquinas (*peers*) além do grande desafio de se ter controle de dados distribuídos [Androutsellis e Spinellis, 2004][Bhatia, 2005].

- **Segurança**

Como todos os tipos de rede, o modelo P2P também está sujeito às ameaças de invasão. As técnicas e modelos para autenticação, autorização, integridade dos dados devem ser integradas ao P2P.

- **Controle de acesso**

Trata de permitir ou não o acesso a uma determinada informação pelos usuários.

- **Interoperabilidade**

Tem como desafio encurtar o tempo de desenvolvimento e habilitar as aplicações a serem implementadas facilmente nos sistemas existentes, pois atualmente as aplicações utilizam protocolos e interfaces específicas.

- **Integridade dos dados**

Se os dados estiverem espalhados pela rede, ou seja, distribuídos por vários pares deverá haver uma maior atenção à integridade desses dados, pois eles podem ou não depender uns dos outros para formar alguma informação.

- **Sincronização**

Os dados podem estar distribuídos em vários pares, logo a sincronização dos dados se torna um desafio nas redes P2P.

- **Distribuição dos dados**

Há diversas formas de se distribuir os dados pela rede, entre elas a replicação (uma alteração de estado é passada para as réplicas) e o particionamento (os dados são divididos em várias partes e cada máquina fica encarregada de uma).

- **Distribuição dos controles**

O controle em rede P2P é necessário e pode ser aplicado em um servidor, como podemos ver no Napster e no ICQ, entre outras aplicações, ou então pode ser aplicado nos próprios clientes, eliminando a presença de um servidor como P2P puro. Um exemplo é o Gnutella que tem o desafio principal de implementar esse controle em rede onde não há um servidor central. Fica, portanto, a cargo dos *peers*, eleger uma máquina para que essa passe a ter o controle, caso a máquina eleita anteriormente se desconecte, ou seja, para qual nó será passado o controle.

3.2 Considerações Finais

Nesse capítulo foram apresentados os principais conceitos de redes *overlay*, que se apresentam como um tipo de rede P2P, suas definições, camada de protocolo, modelos de implementação e os desafios encontrados em seu desenvolvimento.

Do que foi apresentado, é necessário ressaltar que as redes P2P têm características semelhantes às redes VANETs, tais como a descentralização da rede, e a possibilidade que dois ou mais nós colaborem espontaneamente na rede. As redes VANETs, suas características, aplicações e tecnologias envolvidas para o seu desenvolvimento serão introduzidas no próximo capítulo.

4 Redes *Ad Hoc* Veiculares

Redes *Ad Hoc* Veiculares (VANETs) são uma subclasse de redes *ad hoc* e têm como objetivo a comunicação entre veículos e/ou entre veículos e uma infraestrutura de acostamento, pode-se utilizar as torres da telefonia celular ou mesmo um ponto de acesso externo para essa comunicação [Sichitiu e Kihl ,2008], conforme mostrado na figura 4-1.

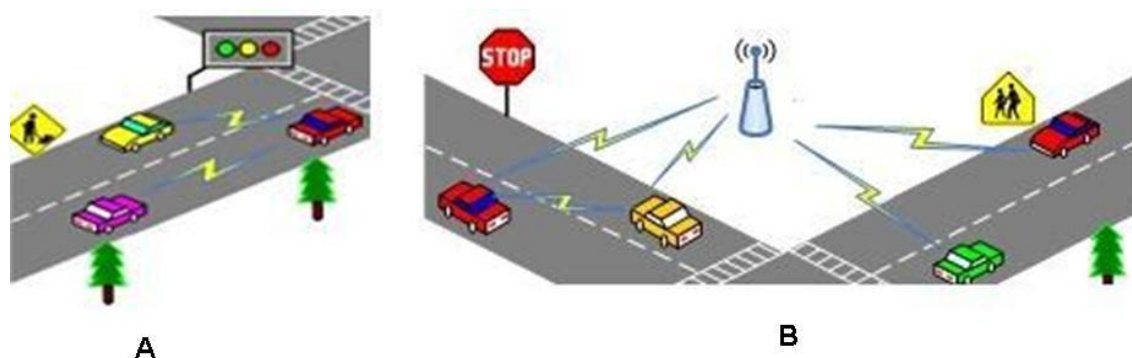


Figura 4-1 - Comunicação de redes ad hoc veiculares (adaptado de [Sterling, 2009])
(A) Troca direta de informações entre veículos (B) Troca de informações entre veículos por meio de infraestrutura.

A figura 4-1 ilustra a comunicação entre veículos em redes VANETs, onde a comunicação entre veículos pode ser direta (Figura 4-1A) ou por meio de uma infraestrutura de acostamento, possibilitando inclusive acesso à internet aos veículos (Figura 4-1B).

A VANET é uma abordagem nova que contribui com novos serviços e aplicações e possui características diferentes de uma rede sem fio tradicional, como podemos observar nas seções a seguir.

4.1 Características de uma rede veicular

O desenvolvimento de aplicações ou serviços para VANETs faz uso de técnicas próprias ou modificadas das redes sem fio, pois essas redes são diferentes das redes sem fio convencionais. As redes VANET não apenas sofrem rápida mudança em suas conexões, mas também têm que lidar com diferentes tipos de densidades da rede [Wisilpongphan, et al., 2007].

A seguir serão apresentadas as características da rede VANET e suas diferenças e semelhanças com uma rede sem fio tradicional.

- **Auto organizáveis**

Essa é uma característica herdada das redes sem fio, sendo as redes veiculares uma subclasse dessas redes, passando assim, à rede, a capacidade de auto-organizar.

- **Mobilidade**

Os nós de uma rede veicular possuem rápida mobilidade, como um automóvel que pode atingir uma alta velocidade, mas possui limitações em sua trajetória: fica restrito a vias públicas, garagens, rodovias, isto é, pode se movimentar somente em locais onde carros possam trafegar.

Diferentemente das redes com fio convencionais as redes sem fio tradicionais não possuem qualquer limitação geográfica, e podem ser levadas para qualquer lugar, ou seja, não há limitações na trajetória de seus nós.

- **Velocidade de Transmissão**

A velocidade de transmissão em redes VANETs é rápida, pois os veículos podem atingir uma velocidade alta e inconstante, podem também estar trafegando em direções opostas e terem somente alguns segundos para transmitir seus dados.

As redes VANET, diferentemente das outras redes sem fio convencionais, não apenas sofrem rápida mudança em sua conexão, mas também têm que lidar com diferentes tipos de densidades da rede [Wisilpongphan et al., 2007].

- **Topologia**

A mudança na topologia é rápida, tornando difícil o seu gerenciamento, devido à alta velocidade relativa entre os veículos na rede.

- **Energia**

Na rede sem fio (MANET), os nós possuem limitação de energia, isto é, os dispositivos de mão como o *palm*, *notebook* ou sensores, ficam presos ao tempo de vida de suas baterias, em lugares onde não há uma fonte de energia elétrica [Mohapatra e Krishnamurthy, 2004], e possuem uma limitação no seu poder computacional, pois a sua arquitetura de *hardware* é restrita. Já na VANET, os veículos podem ser equipados com ampla potência/ energia e podem ser equipados com recursos computacionais significativos [Anda et al., 2005].

- **Largura de banda**

Como todos os dispositivos sem fio que são dependentes da capacidade do hardware para realizar a comunicação.

- **Fragmentação da rede**

A fragmentação da rede ocorre devido ao alcance do raio de comunicação e pelo elevado dinamismo dos veículos.

Embora a rede MANET possa experimentar períodos transitórios de perda de conectividade, na VANET, especialmente sob condições de tráfego ínfimo, os períodos prolongados de desconexão são freqüentes [Abuelela e Olariu, 2007].

A Tabela 4-1 resume uma comparação entre a VANET e a rede sem fio tradicional.

Tabela 4-1 - Comparação VANET vs Redes sem fio tradicional

Características	VANET	Sem Fio Tradicional
Auto configurável	Sim	Sim
Mobilidade	Rápida, com Restrições	Relativamente baixa, Sem restrição
Transmissão	Rápida, Densidades diferentes	Relativamente baixa, Densidades diferentes
Topologia	Dinâmica	Dinâmica
Energia	Constante	Depende de bateria
Banda	Restrita	Restrita
Fragmentação	Freqüente	Aleatória

Podemos observar pela tabela 4-1 que há diferenças significativas entre a rede VANET e a rede sem fio tradicional, sendo as principais, a velocidade que os nós podem ter na rede, o curto tempo de transmissão que nos tem para transmitir e a freqüente fragmentação da rede.

Fazendo uma comparação com as redes de celulares, as redes veiculares possuem duas vantagens principais [Wischhof et al., 2005]:

- **Comunicação Direta**

Os veículos podem se comunicar diretamente com outros veículos sem o uso de uma estação base; com isso, o atraso da comunicação é baixo para comunicação de um salto. Além disso, os carros podem se comunicar em áreas que as redes de celulares não cobrem como em áreas rurais.

- **Ausência de Taxa de Serviço**

A rede VANET não exige infra-estrutura para a comunicação e também nenhum provedor de serviço, diferentemente de um sistema de telefonia celular.

4.2 Aplicações

O potencial impacto social que as redes veiculares apresentam é confirmado pelo aumento de consórcios e de iniciativas que envolvem fabricantes de veículos automotivos, agências governamentais, universidades, dentre outras instituições [Olario, 2007].

Embora o impulso inicial para VANET fosse segurança do trânsito, outras preocupações foram surgindo [Olario, 2007]. Alguns exemplos mais discutidos atualmente são duas importantes classes de aplicações: a segurança e o conforto [Wischhof et al., 2005].

- **Segurança** – é a classe de aplicação que visa o aumento da segurança dos motoristas [Lee et al., 2008] [Transier et al., 2004], através da disseminação das informações de algum acidente que tenha ocorrido na estrada ou na rua. Essas informações poderão servir de informativo para o motorista ou

poderão ativar alguma sinalização na pista. Como exemplo, o sistema de aviso de emergência.

- **Conforto** – é a classe de aplicação que visa o conforto do passageiro, a eficiência do tráfego e a otimização de rota como, por exemplo, Sistemas de informação de Tráfego [Nadeem T. et al., 2004][Wu et al., 2004], interação entre os passageiros dos veículos [Buccioli et al., 2005][Sugiura e Dermawan, 2005] através de *download* de músicas, *chats* e mensagens de voz, entre outros.

Grande parte das aplicações mencionadas acima se beneficia com a utilização da técnica de *Streaming* de vídeo [Guo et al., 2005], que possibilita a visualização e o *download* em tempo real, como por exemplo, em aplicações de segurança onde o motorista e/ou a equipe de resgate receberá vídeos ao vivo da área e poderá antecipar suas ações [Xie et al., 2007].

O desafio de transmitir *Streaming* de Vídeo em redes VANETs envolve o alto dinamismo dos veículos, a alta velocidade dos nós e o alcance de comunicação limitado pelo raio de transmissão do rádio, que pode provocar freqüentes desconexões e até a fragmentação da rede. Isso significa superar os desafios da VANET, que dá suporte a essa técnica, uma vez que uma gama de veículos aderirá a essa rede e estarão aptos a se comunicar. A geração do vídeo é feita de forma não coordenada, sendo que cada veículo é responsável por apenas uma pequena parte dos dados e pode haver redundância e descontinuidade [Guo et al., 2005].

4.3 Tecnologias Envolvidas

Nessa seção veremos as principais tecnologias utilizadas para o desenvolvimento de soluções de comunicação em VANETs, da camada física indo à camada de transporte.

4.3.1 Camada Física

Em 1999, a FCC alocou um espectro de frequência para a comunicação inter veicular e veicular com uma infraestrutura de acostamento, estabelecendo regras de serviços e licenças para DSRC em 2003 [FCC, 2009].

DSRC é um serviço de comunicação que utiliza uma banda de 5.85-5.95GHz (banda de 5.9GHz) de um espectro de 75MHz, com o objetivo de fornecer capacidade de comunicação sem fio para aplicações de transporte a uma distância de um quilômetro [Yousefiet al. 2006]. DSRC Permite dois tipos de modelos de operação [Nandan, et al, 2005]:

- **Modo Ad Hoc**

Caracterizado pela rede de vários saltos distribuídos – tipicamente na comunicação entre veículos.

- **Modo infraestruturado**

Caracterizado por uma rede móvel centralizada de um salto – comunicação entre um veículo e uma estrutura de acostamento (*gateway*).

A banda está dividida em sete canais, sendo seis canais para serviços e um canal para controle como mostra a figura 4-2.

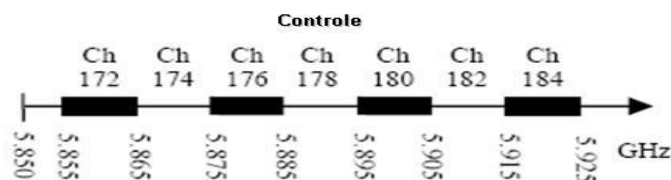


Figura 4-2 - banda DSRC Norte Americana adaptado de [Armstrong, 2009]

O canal de controle é regularmente monitorado por todos os carros [Sengupta, Xu , 2004]. O padrão desenvolvido pelo *American Society for Testing and Material* (ASTM), conhecido como padrão ASTM E2213, destina o DSRC a ser uma extensão do IEEE 802.11, sendo uma tecnologia para o ambiente de veículo de velocidade alta. A camada física (PHY) do DSRC é adaptada da IEEE 802.11a PHY, baseada na tecnologia de *orthogonal frequency division multiplex* (OFDM). A camada de controle de acesso múltiplo (MAC) do DSRC é muito semelhante à do IEEE 802.11 MAC, baseado no *carrier sense multiple access with collision avoidance* (CSMA / CA), sendo um protocolo com algumas modificações [Yin et al., 2004].

O OFDM é um sistema de multiplexação por divisão de frequência (FDM) no qual múltiplos sinais são enviados em diferentes frequências. Embora o OFDM venha ganhando aumento na sua popularidade devido a sua elevada eficiência espectral, inerente à capacidade para combater o multi - caminho enfraquecido, e pela simplicidade de projetar um *transceiver* [Yin et al., 2004].

A figura 4-3 mostra a estrutura de dados da camada física do DSRC [Yin et al., 2004], onde:

A1 – A10 – São 10 símbolos curtos de identificação;

C1 e C2 – Seqüência de formação longa;

CP - Prefixo cíclico que protege contra ISI.

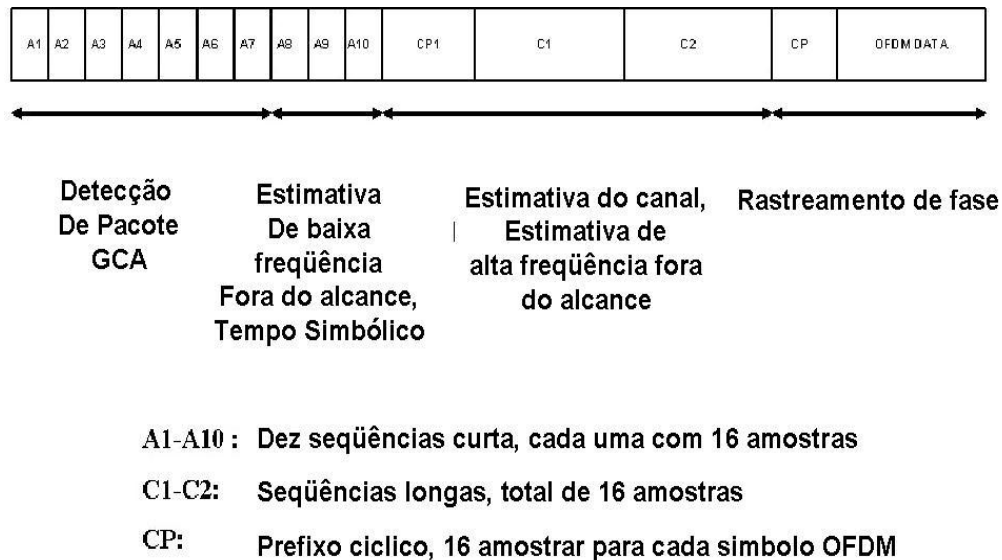


Figura 4-3 - Estrutura de dados da camada física DSRC (adotado de[Yin, at all, 2004])

Para a camada MAC um grupo da IEEE está investigando uma nova alteração da PHY / MAC do padrão 802,11 concebida para VANET: o *Wireless Access in Vehicular Environments* (WAVE), que utiliza o CSMA/CA como base para a camada MAC.

Outra abordagem foi utilizada no projeto europeu CarTALK2000 [Morsink et al., 2003], que é baseado no FleetNET [Festag et al., 2004] com o objetivo de conceber novas soluções para VANET.

Foi desenvolvido também o ADHOC MAC [Borgonovo et al., 2004], um protocolo MAC que implementa um mecanismo dinâmico TDMA, que é capaz de prover um acesso imediato, largura de banda variada e canais confiáveis necessários para QoS de entrega.

O mecanismo dinâmico TDMA é realizado através do protocolo *Reliable R-ALOHA* (RR-ALOHA), um novo protocolo de reserva distribuído, que é capaz de estabelecer dinamicamente um canal de *broadcast* [Borgonovo, et al., 2002].

Como esse protocolo MAC trabalha com uma estrutura de *slotted frame*, seu mecanismo de acesso é facilmente adaptado para o UMTS *Terrestrial Radio Access Time Division Duplex* (UTRA-TDD) [Menouar e Filali, 2006], sendo que a informação do *slotted* pode ser provida pelo *Global Positioning System* (GPS), como foi realizado no projeto CarTALK2000.

No trabalho de Menouar e Filali [Menouar e Filali, 2006] podemos verificar uma comparação do padrão 802.11 que utiliza o protocolo WAVE com o ADHOC MAC, como mostra a tabela 4-2.

Tabela 4-2 - Comparação padrão 802.11 Vs ADHOC MAC

	802.11 (MAC layer)	ADHOC MAC
Baseado em	CSMA/CA	RR-ALOHA
Maturidade da implementação	Madura e evoluindo	Mediano
QoS e capacidade em tempo real	Pequeno	Mediano
Mobilidade	Mediana que evolui para alta	Mediano
Confiabilidade multicast / broadcast	Não	Sim
Tempo de sincronização	Não é necessário	Obrigatório

A tabela 4-2 mostra que há muito a se trabalhar ainda nos protocolos MAC, e que uma combinação entre os dois protocolos poderia resultar em uma solução mais completa para a VANET, pois o padrão 802.11 possui uma mobilidade mais elevada e o ADHOC permitiria maior confiabilidade e garantia de QoS.

4.3.2 *Camada de Rede*

Tradicionalmente, a camada de rede é responsável por endereçar (nomeando os elementos da rede), rotear (encontrar bons caminhos), e transmitir (circulação dos pacotes na rede) dados entre a origem e destino.

Em uma rede sem fio há varias técnicas para se prover o endereçamento dos nós na rede, porém as estratégias mais relevantes para uma rede veicular são [Sichitiu e Kihl ,2008]:

- **Endereçamento fixo** - cada nó tem um endereço fixo atribuído por algum mecanismo no momento em que o nó entra na rede. Grande parte das aplicações de redes *ad hoc* e protocolos existentes assumem essa técnica de endereçamento [Sichitiu e Kihl ,2008].
- **Endereçamento geográfico** - cada nó é caracterizado por sua posição geográfica, ou seja, quando o nó se move, o seu endereço é alterado. Podem-se utilizar atributos adicionais para endereçar o nó, tal como a velocidade, direção, tipo de carro entre outros.

A camada de rede presta um serviço de mapeamento de endereço na rede, evitando um conflito de endereços na rede, ou seja, o endereço tem que ser único para cada nó, não podendo haver dois nos com o mesmo endereço.

Outra função da camada de rede é fornecer a melhor rota entre a origem e o destino para que os dados sejam entregues integralmente e no menor tempo possível. Em uma rede veicular podemos classificar os protocolos de roteamento em diferentes maneiras [Sichitiu e Kihl ,2008][Lie e Wang,2007][Chen et al., 2008]: *Ad Hoc*, baseado no posicionamento geográfico, baseado em *cluster*, *broadcast* e roteamento geográfico. Esses tipos de roteamento são detalhados a seguir:

4.3.2.1 Rotamaneto Ad Hoc

Como a rede veicular herda algumas características de uma rede sem fio tradicional, ela também carrega consigo alguns protocolos já existentes para uma rede *wireless* tradicional, tal como o AODV (*Ad-hoc On-demand Distanc Vector*) [Perkins e Royer, 1999] and DSR (*Dynamic Source Routing*) [Johnson e Maltz, 1996].

Esses protocolos AODV e DSR, são protocolos reativos, ou seja, os nós não possuem uma tabela de rota e a descoberta de rota é feita quando há uma requisição de envio de mensagem. Há diversos estudos que avaliam e comparam o desempenho desses protocolos em diversas condições de tráfego em VANETs [Namboodiri et al., 2004][Santos et al., 2005]. Os testes mostram que a maioria dos protocolos *ad hoc*, como AODV e DSR, sofre com o dinamismo natural da mobilidade dos nós, pois eles tendem a ter um baixo *throughput* de comunicação.

No trabalho de Wang e colaboradores [Wang et al., 2005,11], foi avaliado o protocolo AODV em uma experiência real com seis veículos, em que cada veículo possuía um laptop IBM A-model com uma placa PCMCIA IEEE 802.11b NIC. Foi mostrado que o protocolo AODV é incapaz de rapidamente encontrar, manter e atualizar rotas de longas distâncias em uma rede veicular. Há uma grande perda de pacotes devido ao excesso de erro de rota, ficando difícil estabelecer uma conexão TCP. Por isso são realizadas algumas alterações no protocolo *ad hoc* existente, para que ele possa lidar com alta mobilidade e com topologia altamente dinâmica.

Já Namboodiri e colaboradores [Namboodiri et al., 2004] consideraram o roteamento de um veículo com um *gateway*. O cenário de uma rodovia foi utilizado na simulação, além de informações sobre os nós, como a velocidade e a localização, para predizer o tempo de vida do *link*. A estratégia para reduzir o efeito da interrupção frequente das rotas foi o desenvolvimento de protocolos baseados em predição. Um dos protocolos

foi o PRAODV que constrói uma nova rota alternativa antes do final da vida útil estimado do *link*. Já o PRAODV-M escolhe o caminho que tenha uma vida útil prevista maior entre múltiplas opções de rota, em vez de selecionar a rota pelo caminho mais curto em AODV e PRAODV. Suas simulações revelaram uma pequena melhora com relação à taxa de entrega de pacote, porém esse método depende da precisão do método de previsão.

Outra modificação do protocolo AODV, realizada no trabalho de Ooi e Fisal [Ooi e Fisal, 2004] foi o modo de transmissão: envia mensagens apenas dentro de uma zona relevante (ZOR). A idéia básica é a mesma que a *location-aided routing* (LAR) [Ko e Vaidya, 2000]. ZOR geralmente é especificado como uma série de retângulos ou círculos para a divisão do ambiente físico, sendo determinado pela aplicação [Briesemeister et al., 2000]. Por exemplo, para o modelo de rodovia onde esta é dividida, o ZOR poderá cobrir as regiões ao redor do acidente. Nos casos em que nem todos os veículos na ZOR devem receber a mensagem, a camada de rede decide se o pacote recebido deverá ser enviado para a aplicação ou não. No entanto, é normalmente entendido que todos os veículos dentro de uma determinada zona de roteamento devem participar no processo de transmissão [Sichitiu e Kihl, 2008].

4.3.2.2 Roteamento Baseado no posicionamento geográfico

Roteamento baseado no posicionamento utiliza informações geográficas obtidas por mapas da cidade, modelos de tráfego ou sistemas de navegação de bordo do veículo. Esse modelo de roteamento foi identificado como sendo o paradigma de roteamento mais promissor para as VANETs [Ilie e Wang, 2007].

Nessa fase de roteamento, os nós de uma rede veicular podem fazer uso das informações de posicionamento para tomar a decisão no momento do encaminhamento das informações. Por exemplo, o roteamento *greedy* sempre encaminha o pacote para o nó que está geograficamente mais próximo ao destino. GPSR (*Greedy Perimeter Stateless*

Routing) [Karp e Kung, 2000] é um dos mais conhecidos posição com base em protocolos na literatura.

O protocolo GPSR entende que o nó origem sabe a localização do destino. Além disso, cada nó tem informações atuais sobre a posição de seus vizinhos (obtida através de um procedimento *beaconing*). Todos os pacotes incluem a localização do destino. Assim, quando um nó recebe um pacote, ele pode tomar a melhor decisão ao transmitir o pacote até chegar ao seu destino. Esse protocolo utiliza dois métodos de repassar os dados (ver figura 4.4)

- ***Greedy***

Em que o nó origem busca escolher sempre o nó mais próximo do destino no limite de células de alcance, sem considerar os próximos saltos da comunicação.

- ***Perimeter***

Caso não exista um nó vizinho mais próximo do destinatário, o GPSR ativa o modo Perímetro (*Perimeter Routing*), que realiza um roteamento através de vizinhos mais próximos a ele. Caso não tenha nenhum nó mais próximo, o nó envia a mensagem para o nó que está afastado do nó destino até encontrar outro nó que está próximo ao destino, retornando novamente ao modo *greedy*.

A figura 4-4 demonstra o funcionamento do GPSR. Nesse exemplo, o veículo A deseja enviar dados para o nó G. Para isso, o nó A repassa os pacotes para o nó B que é o vizinho mais próximo de G, o mesmo ocorrendo com o nó B. O veículo C, ao receber o pacote, verifica que nenhum dos seus vizinhos (nós dentro do círculo com o centro em C) está mais próximo de G do que ele mesmo, assim o nó C ativa o modo *perimeter* (seta tracejada) e envia os dados para D, da mesma forma D utiliza-se do modo *perimeter* para enviar dados para E, que por sua vez encontra um nó mais próximo de G que o nó C e assim o pacote volta a ser repassado pela estratégia *greedy* até o nó G.

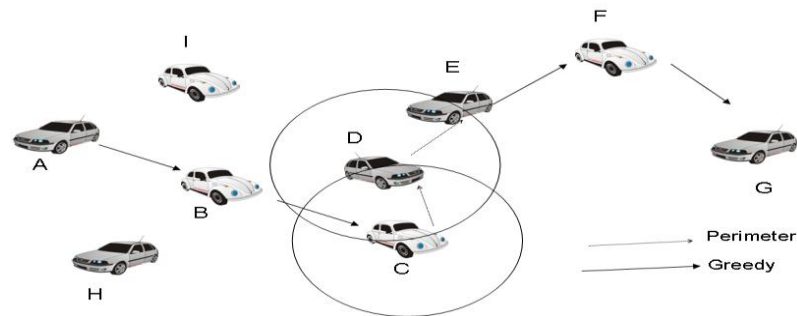


Figura 4-4 - Funcionamento do protocolo GPSR

Devido à natureza dos apátridas GPSR, versões modificadas do protocolo têm sido propostos para a as redes veiculares [Sichitiu e Kihl ,2008]. Alguns trabalhos utilizam o mapa digital no sistema de navegação para calcular uma rota da origem ao destino [Lochert et al., 2003] [LeBrun et al., 2005].

Lochert e colaboradores [Lochert et al., 2003] propuseram um protocolo de roteamento conhecido como *Geographic Source Routing* (GSR), que é auxiliado por um mapa das ruas da cidade. O GSR usa um Serviço de Localização Reativa (SPI) para chegar ao destino. O algoritmo precisa ter o conhecimento da topologia global da cidade, que é fornecido por um mapa estático das ruas. Dada esta informação, o emissor determina os entroncamentos que o pacote tem que percorrer, para qual é utilizando o algoritmo do caminho mais curto Dijkstra. A transmissão entre os entroncamentos é feita através de um modelo baseado no posicionamento, que combina o roteamento geográfico com o conhecimento da topologia das ruas da cidade dadas por um mapa estático. Os resultados da simulação demonstraram que o GSR tem uma taxa média melhor na entrega das mensagens, um menor consumo de banda total e possui uma latência semelhante a os protocolos AOVD e DSR.

Lochert e colaboradores [Lochert et al., 2005] propuseram uma alteração do protocolo GPSR, o GPCR (*Greedy Perimeter Coordinator Routing*), onde não se utiliza

qualquer fonte de encaminhamento ou de disponibilidade de mapas. Ele usa o fato de que os nós em uma junção entre ruas seguem um gráfico planar natural. Essas junções são os únicos lugares onde se decide para quem a mensagem será transmitida, para que esta possa chegar a seu destinatário. As mensagens, ao invés de cruzarem essas junções e irem ao nó mais perto do destinatário, são enviadas para o nó, chamado de coordenador, que está em uma junção. A figura 4-5 (a) ilustra essa estratégia. Caso estivéssemos trabalhando com um algoritmo *greedy*, o nó S teria enviado a mensagem para o nó N1; porém, com o modelo usado neste trabalho, o nó S envia a mensagem para o nó C1.

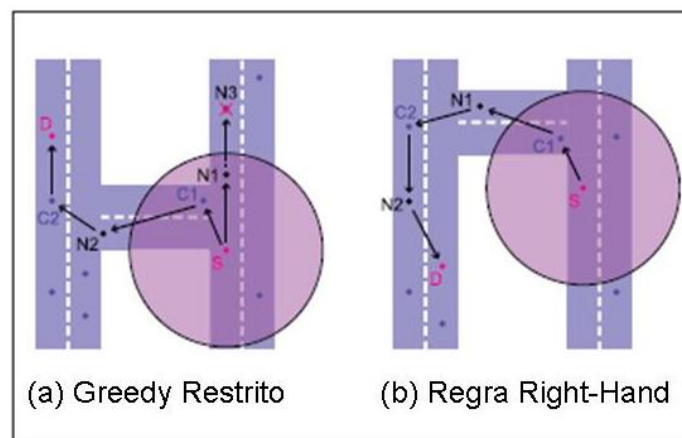


Figura 4-5 Estratégias do protocolo GPCR (adotado de [Lie e Wang,2007])

Alem de usar esta restrição no algoritmo *greedy*, essa solução também usa uma estratégia de reparação para sair do mínimo local, ou seja, em casos em que o nó não tenha nenhum vizinho mais próximo do destino. Essa estratégia de reparação decide, em cada junção, qual seria a próxima rua que o pacote poderia ser transmitido (regra *right-hand*) e aplica o roteamento *greedy*, entre as junções, para encontrar a próxima junção entre as ruas. A figura 4-5 (b) demonstra um exemplo utilizando a regra *right-hand* para decidir qual rua o pacote poderá seguir utilizando a estratégia de reparação. A simulação foi realizada no NS-2 com a topologia da cidade de Berlim, na Alemanha, mostrando que o GPCR tem uma

maior taxa de entrega que GPSR, todavia com uma média maior de saltos e com um aumento no tempo de entrega dos pacotes.

Liu, e colaboradores [Liu et al., 2004] propuseram um novo protocolo baseado no posicionamento geográfico chamado de A-STAR (*Anchor-based Street and Traffic Aware Routing*). Essa solução utiliza mapas das ruas para calcular as seqüências de junções, aqui chamado de âncoras, através da qual o pacote deve passar para se entregue ao seu destinatário. Diferentemente da GSR, essa solução calcula as âncoras conforme as condições do tráfego de carros através de mapas estatísticos (contando o número de rotas de ônibus na cidade para identificar as âncoras com maior conectividade) e com mapas dinâmicos (monitoramento dinâmico das últimas condições de tráfego de veículos para identificar a melhor rota) para identificar o caminho que o pacote irá percorrer até chegar ao destinatário. Outro diferencial é a estratégia de recuperação local. Para evitar que outros pacotes passem por áreas vazias, a via pela qual o mínimo local ocorreu é, temporariamente, marcada como "fora de serviço". Ruas que estiverem "fora de serviço" não são utilizadas como âncora, essas estradas só voltarão a ser computadas após um determinado tempo t quando elas passarem para um estado de "operacional". Essa solução mostrou um melhor desempenho comparado com o GSR e GPSR, pois elas podem escolher o caminho com melhor conectividade para a entrega dos pacotes, entregando 40% a mais de pacotes do que o GSR.

Protocolos baseados no posicionamento geográfico, colocados dentro de uma cidade, enfrentam grandes desafios, uma vez que os veículos têm uma distribuição desigual, devido ao fato de haver maior concentração de veículos em uma estrada do que em outras. Além disso, sua mobilidade é restringida pelos padrões das estradas, juntamente com os obstáculos encontrados tais como edifícios, e podem levar a desconexão da rede.

4.3.2.3 Roteamento Baseado em Cluster

No roteamento baseado em *cluster*, é criada uma infra-estrutura de rede virtual entre os nós através do agrupamento que fornecerá uma maior escalabilidade. A Figura 4-6, ilustra um roteamento baseado em cluster em VANETs. Cada *cluster* pode ter uma cabeça de *cluster*, que é responsável pela coordenação intra e inter-*cluster* na rede, e tem uma função de gestão. Os nós dentro de um *cluster* se comunicam através de ligações diretas. Comunicação inter-*cluster* é realizada através das cabeças de *cluster*.

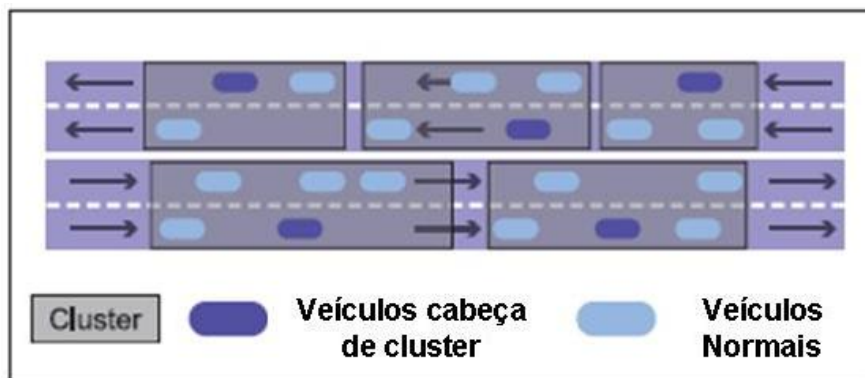


Figura 4-6 – Modelo de roteamento em *cluster* (adotado de [Lie e Wang,2007])

Esses protocolos foram considerados adequados para as redes desde que os veículos estejam em uma auto-estrada, podendo naturalmente formar um *cluster* [Sichitiu e Kihl, 2008].

No trabalho de Blum e colaboradores [Blum et al., 2003] é possível observar um algoritmo chamado de *Clustering for Open IVC Networks* (COIN). A eleição do nó cabeça de cluster baseia-se nas intenções do condutor bem com na dinâmica do veículo. Este algoritmo lida também com as oscilações das distâncias entre os veículos. Os autores demonstraram que o COIN produziu uma estrutura estável, mas introduziu uma sobrecarga na rede.

Santos e colaboradores [Santos et al., 2005] apresentam um algoritmo de roteamento reativo baseado na localização que serve de base para o *cluster* realizar *flooding* na VANETs, chamado de LORA_CBF. Cada nó pode ser a cabeça de *cluster*, *gateway* ou membro de um *cluster*. Cada *cluster* tem exatamente uma cabeça de *cluster*. Se um nó é ligado a mais de um *cluster*, é chamado de *gateway*. A cabeça de *cluster* mantém informações sobre os seus membros e *gateways*. Os pacotes são transmitidos a partir de uma origem para um destinatário por um protocolo de roteamento semelhante ao *greedy*. Se a localização do destino não está disponível, a fonte enviará os pacotes de pedido de localização (LREQ). Esse processo é semelhante à fase de descoberta da rota do AODV, mas apenas a cabeça de *cluster* e os *gateways* são quem divulgarão as mensagens LREQ e LREP (Resposta da Localização). A figura 4-7 ilustra a estrutura desenvolvida por Santos e seus colaboradores. Para a abstração do desempenho foi realizada uma comparação entre os protocolos AODV e DSR e o LORA_CBF em um cenário urbano e também em uma auto-estrada. Os resultados demonstraram que a mobilidade da rede bem como seu tamanho teve um impacto mais significativo nos protocolos AODV e DSR do que no LORA_CBF[Santos et al., 2005].

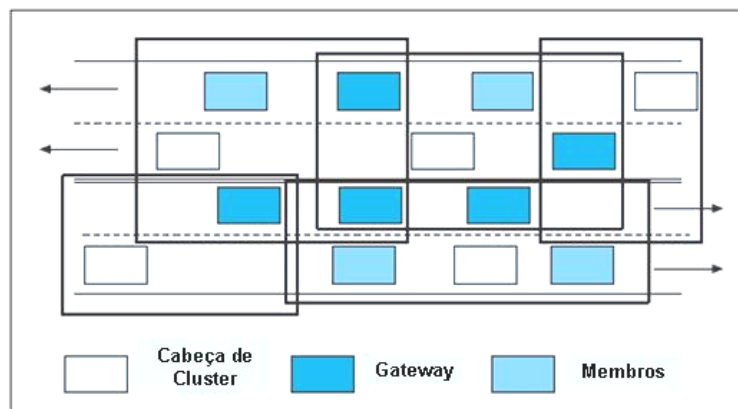


Figura 4-7 – Solução dada por Santos e colaboradores (adotado de [Sichitiu e Kihl, 2008])

Os protocolos baseados em *cluster* podem alcançar uma boa escalabilidade para grandes redes, porém um obstáculo importante em sistemas VANET de rápida modificação, para eles, é a latência e o overhead envolvidos na formação e manutenção destes clusters [Li e Wang, 2007].

4.3.2.4 Rotemaneto Broadcast

Os roteamentos broadcast têm diversas finalidades em uma VANET, tal como a disseminação das condições do trafego de veículos de uma determinada rua, informar sobre as condições do tempo, anúncios, acidentes ao longo da estrada entre outros. Protocolos *unicast* também fazem o uso deste método para encontrar a melhor rota para o destino (fase de descoberta de rota).

O modo mais fácil de implementar um roteamento *broadcast* é usando o *flooding*, fazendo com que cada nó retransmita a mensagem para todos os seus vizinhos com exceção do nó que o enviou a mensagem. O *Flooding* garante que a mensagem seja entregue para todos os nós participantes da rede. Essa estratégia tem um bom desempenho para um número limitado de nós na rede e é facilmente implementada. Mas quando o número de nós aumenta, o seu desempenho cai, pois poderá haver um significativo aumento em sua largura de banda, sobrecarregando a rede.

Durresi e colaboradores [Durresi et al., 2005] apresentaram um protocolo *Broadcast* de emergência, BROADCAST, baseado em uma estrutura hierárquica para uma rede desenvolvida em uma auto-estrada. Em BROADCAST, a rodovia virtual é dividida em células, que se movem como a movimentação dos veículos. Os nós em uma auto-estrada são divididos em dois níveis hierárquicos: o primeiro nível inclui todos os nós em uma célula, o segundo nível é representado pelas células refletoras, que são os nós geralmente próximos ao centro geográfico da célula. As células refletoras se comportam como uma estação base (cabeça de *cluster*) em um determinado intervalo de tempo, que irá lidar com

as mensagens de emergência provenientes dos membros da mesma célula ou de membros próximos de células vizinhas. Além disso, as células refletoras servem como nós intermediários no roteamento de mensagens de emergência transmitidas de células refletoras vizinhas. Este protocolo tem um desempenho semelhante ao protocolo de roteamento baseado em *flooding*. Entretanto, é mais simples e só funciona com redes criadas sobre uma auto-estrada.

O protocolo *Urbano Multi-Hop Broadcast* (UMB) [Korkmaz et al., 2004] foi desenvolvido para superar interferências, colisões de pacotes, e problemas de nós ocultos durante a disseminação das mensagens via *broadcast multihop*. Em UMB, o nó origem tenta selecionar o nó mais distante direcionando o *broadcast* e fazendo com que o pacote seja reconhecido sem qualquer informação a priori da topologia. Nas interseções são instalados repetidores para transmitir os pacotes a todos os segmentos da estrada. O protocolo UMB possui uma porcentagem de sucesso mais alta, com uma sobrecarga de pacote na rede e um denso tráfego de veículos, maior do que o protocolo de distância 802.11 e o protocolo aleatório 802.11 que são adaptados do padrão IEEE 802.11 para evitar colisões durante o reenvio de pacotes.

Vector-based Tracking Detection (V-TRADE) e *History-enhanced V-TRADE* (HV-TRADE) [Sun et al., 2000] são protocolos *broadcast* de mensagens utilizando o GPS. A idéia é semelhante ao protocolo de roteamento unicast *Zone Routing Protocol* (ZRP) [Haas e Pearlman, 1997]. Com base nas informações de posicionamento e movimentação, seus métodos são: classificar os vizinhos dentro de grupos de retransmissão diferentes. Para cada grupo, apenas um pequeno subconjunto de veículos (chamados de veículos de fronteira) é selecionado para retransmitir as mensagens. Essas soluções mostraram ter uma significativa melhora na largura de banda, pois esses protocolos diminuem o número de retransmissão de mensagens. Entretanto, ainda possuem um alto *overhead*, pois os próximos nós a enviar as mensagens são selecionados a cada *hop*.

Protocolos *Broadcast* têm como limitação um grande consumo na largura de banda, pois a disseminação das mensagens cresce exponencialmente [Li e Wang, 2007]. Podem também sofrer de colisões de pacote, já que cada nó precisa enviar e receber as mensagens quase simultaneamente.

4.3.2.5 Roteamento *Geocast*

Roteamento *Geocast* [Maihöfer, 2004] é basicamente um roteamento *multicast*: baseado no posicionamento, que tem como objetivo entregar as mensagens a partir de um nó fonte para todos os outros nós que pertencem a uma região geográfica específica (Zona de relevância, ZOR). Veículos fora dessa região não são alertados, evitando o envio desnecessário de mensagem.

Geocast pode ser implementado com um serviço de *multicast* simplesmente pela definição do grupo *multicast* sendo as regiões geográficas. A maioria dos métodos utilizados para o roteamento *geocast* são baseadas no *flooding* dirigido, que tenta limitar a sobrecarga de mensagens na rede fazendo com que o *flooding* ocorra somente dentro de uma região relevante.

Briesemeister, e colaboradores [Briesemeister et al., 2000] propuseram um esquema *geocast*, a fim de evitar colisões de pacote e reduzir a quantidade de mensagens retransmitidas. Quando um nó recebe uma mensagem, ele não retransmite essa informação imediatamente, este espera até um determinado tempo para decidir para quem ele irá enviar. O tempo de espera depende da distância do nó para o nó que transmitiu a mensagem. Esse tempo de espera é mais curto quando o nó receptor estiver mais distante. Assim, os nós que estão na borda da área de recepção participam da trajetória da mensagem dando maior vazão para a mensagem. Quando esse tempo de espera expira, se o nó não receber a mesma mensagem dos outros nós, então essa mensagem é retransmitida. Com isso, evita-se um *broadcast* em toda rede e a transmissão é otimizada em torno dos veículos

iniciais. O sistema também utiliza um número máximo de *hops* para limitar o escopo do *flooding*. Bachir e Benslimane [Bachir e Benslimane, 2003] desenvolveram um protocolo geocast inter-veicular, chamado IVG, para transmitir uma mensagem de alerta para todos os veículos em uma área de risco. A ideia principal é muito semelhante ao trabalho de Briesemeister e colaboradores [Briesemeister et al., 2000].

Maihöfer e Eberhardt [Maihöfer e Eberhardt, 2004] discutem a estrutura de *cache* e a maneira de selecionar a vizinhança para lidar com a situação de alta velocidade na VANET comparado aos protocolos regulares *geocast*. A ideia principal do seu *cache greed geocast*, dentro do protocolo ZOR, é adicionar uma pequena *cache* para a camada de roteamento que detém aqueles pacotes que um nó não pode transmitir devido à sua falta de vizinhos. Ao chegar um novo nó, a mensagem que está na *cache* poderá ser transmitida para esse nó recém-descoberto. A estratégia do conhecimento da distância do vizinho escolhe o mais próximo do destino em um raio r , em vez do último nó do alcance de transmissão, como no modelo de roteamento *greedy*. Em um modelo *greedy* o nó intermediário sempre seleciona o próximo nó mais próximo dos nós que estão na borda do alcance de transmissão e que irão retransmitir a mensagem, de modo que o próximo nó a ser selecionado tem uma alta possibilidade de deixar o alcance de transmissão devido à alta mobilidade do nó. Os resultados da simulação mostram que a *cache* para mensagem que não tem como transmitir devido à fragmentação da rede ou por falta de vizinhos teve uma melhora significativa na taxa de entrega das mensagens em um *geocast*. A estratégia da seleção dos vizinhos resultou em uma diminuição na carga de mensagens na rede e uma diminuição no atraso na entrega das informações.

Para esclarecer a diferença entre os protocolos, *unicast*, *broadcast* e *geocast* são visto na figura 4-8, que ilustra os três protocolos:

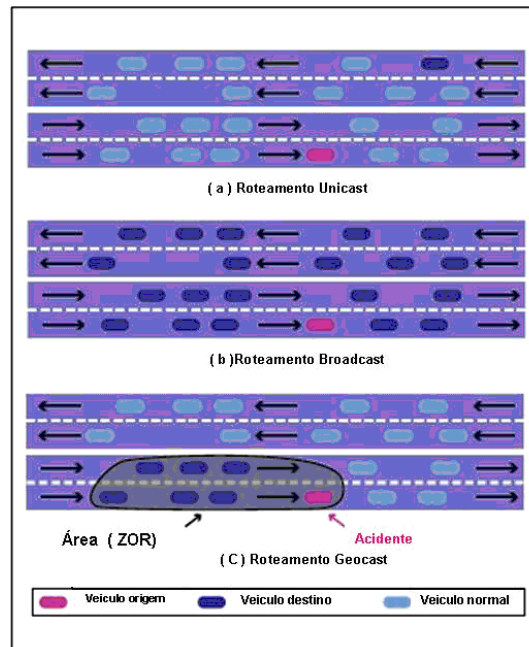


Figura 4-8 - Diferentes protocolos de roteamento (adaptado de [Li e Wang, 2007])

4.3.3 Camada de transporte

Embora o protocolo TCP em rede fixa possa alcançar um bom *throughput*, o seu desempenho no meio de redes sem fio mostrou-se baixo porque o TCP trata a perda de pacote como um sinal de congestionamento da rede. Em redes sem fios, contudo, perdas podem acontecer por razões como congestionamento, além de condições do canal ou colisões. Como resultado de uma perda, o TCP terá reduzido o tamanho de janela de congestionamento e diminuirá o *throughput*. Além disso, pelas características das redes VANET a simetria de caminho não está garantida, conseqüentemente, pode-se diminuir o desempenho do TCP devido à estimativa incorreta de tempos de RTT, resultando em retransmissões desnecessárias [Chen et al., 2007].

Hui, Mohapatra [Hui, Mohapatra, 2005] examinaram o desempenho entre o UDP e o TCP em rede VANET através do uso de dois cenários reais: rodovias interestaduais 80 (I-80), I-5 e zonas suburbanas em Sacramento, Califórnia. Os autores mostraram que embora o *throughput* do TCP diminua em rodovias, é mais eficiente que no UDP porque o nó origem TCP pode detectar um receptor desligado e parar de transmitir dados aos nós intermediários mais rapidamente do que o nó origem UDP. Diferentemente do modo de envio do protocolo TCP onde o remetente envia parte dos dados e aguarda a resposta do receptor antes de enviar dados complementares para os nós intermédios, o UDP remetente continua a enviar dados para os nós intermediários, independentemente se o receptor ainda está a receber ou não. Com isso, os remetentes podem desperdiçar a largura da banda de nós intermediários para um período de tempo.

Para resumir o desempenho do TCP e UDP podemos ver a tabela 4-3, que demonstra que esses protocolos sofrem com a alta mobilidade dos nós e pelas longas distâncias entre eles. Portanto, as redes veiculares podem se beneficiar com o desenvolvimento de novos protocolos de transporte ou alterações nos protocolos já existentes.

Tabela 4-3 - Resultados de que avaliaram os protocolos de transporte [Sichitiu e Kihl, 2008]

Trabalhos	Numero de nós	MAC	Tipo de protocolo	Velocidade	Distancia	Throughput
[Chuang et al., 2005]	4	802.11b	TCP	40 km/h	N/A	~ 800 kb/s
[Watanabe et al., 2005]	3	802.11b	UDP	8-113 km/h	< 145 m	500-2300 kb/s
[Hui e Mohapatra, 2005]	3	802.11g	UDP	<5km/h	N/A	1-5 Mb/s

Schmitz e colaboradores [Schmitz et al., 2006] investigaram os caminhos característicos que são relevantes para um protocolo de transporte para uma rede veicular. Além disso, é apresentado um protocolo de transporte veicular (VTP), que observa a rede e usa dados estatísticos para melhorar o desempenho quando ocorre a desconexão. O *Mobile Control Transport Protocol* (MCTP) [Bechler et al., 2005] é proposto com base em princípios semelhantes ao *Ad Hoc TCP* [Li e Singh, 2001]. O MCTP tem como objetivo prover uma qualidade de serviço fim a fim entre um veículo e um *host* de internet, via um *gateway*, posicionado nos canteiros das estradas.

Todos os protocolos de transporte propostos para as redes veiculares são projetados para aplicações que exijam roteamento *unicast* [Sichitiu e Kihl, 2008]. Uma vez que muitas aplicações requerem um roteamento *geocast* ou *multicast*, há uma clara necessidade de novas abordagens que não são baseadas em protocolos de transporte tradicionais.

4.4 Considerações Finais

Este capítulo apresentou os principais aspectos envolvendo VANETs, incluindo definição, aplicações, principais características, diferenças das redes tradicionais sem fio e as principais tecnologias utilizadas para o desenvolvimento de aplicações.

Do que foi apresentado, é necessário ressaltar as principais características de redes VANETs, as quais refletem também seus principais desafios, a saber: alta velocidade na mobilidade do nó e na transferência dos dados, freqüente fragmentação da rede e rápida mudança da topologia. As características das redes VANETs constituem um grande desafio à construção de aplicações para essas redes, em especial quando as aplicações exigem consistência espaço-temporal e escalabilidade, como é o caso de aplicações de AVCs. No próximo capítulo será apresentada uma solução de redes overlay para o suporte de aplicações de ambientes virtuais colaborativos em redes veiculares.

5 Projeto de um Suporte *Overlay* para Ambientes Virtuais Colaborativos sobre Redes Veiculares

O objetivo deste trabalho é o projeto, a implementação e avaliação de uma rede *overlay* de suporte a ambientes virtuais colaborativos (AVCs) em Redes *Ad Hoc* Veiculares. Os AVCs podem ser desde jogos multiusuário de entretenimento sendo acessados por usuários participantes em dispositivos integrados a diferentes veículos que trafegam em diferentes velocidades, até simulações de AVCs, por exemplo, de treinamento de equipes de preparação e resposta a emergências por meio de dispositivos em veículos enquanto estes estão sendo dirigidos a um local de acidente. Um AVC acessado nos veículos das equipes de gerenciamento da emergência, enquanto estas se dirigem para o foco da emergência, pode representar geometricamente, de forma sintética (mundo virtual), o ambiente físico alvo da emergência. Assim, o treinamento no ambiente virtual que reflete o ambiente real, enquanto em trânsito, pode ajudar na tomada de decisão das equipes de comando quando estas chegam ao local, reduzindo perdas de vida e de patrimônio.

Os veículos poderão estar tanto fixos, ao redor do acidente, ou em movimento, indo para uma mesma direção (local do acidente) ou em direção oposta (veículos que fazem a ronda do local). Essa movimentação dos veículos faz com que a conexão entre eles tenha que ser estabelecida rapidamente, e que as transferências de dados para as atualizações do ambiente virtual sejam feitas no menor espaço de tempo possível para evitar inconsistências na visualização do ambiente pelos usuários participantes. O objetivo da rede *overlay* é auxiliar na comunicação entre os usuários participantes, diminuindo o atraso na entrega das

mensagens para a aplicação, e abstraindo a complexidade da rede veicular para a aplicação. Diferentemente de outras redes overlay existentes, que visam troca de arquivos ou *streaming* de vídeo, a rede proposta visa suportar ambientes virtuais em redes veiculares que não possuem infra-estrutura externa.

Neste capítulo serão descritos como os AVCs são organizados bem como a dinâmica de entrada, saída de avatares e manutenção de consistência dos AVCs. Será descrito também o simulador de redes VANETs utilizado nos experimentos. Finalmente, serão apresentados a rede Gnutella modificada que atende as necessidades de AVCs, bem como o ambiente integrado de avaliação e resultados obtidos.

5.1 Estrutura dos Ambientes Virtuais Colaborativos

Assumimos que os ambientes virtuais colaborativos considerados são divididos em hexágonos iguais, como visto na Figura 5-1, gerando um conjunto de células, onde usuários participantes de uma mesma célula formam um grupo de participantes com o mesmo interesse. Cada **célula** é chamada de **zona** [Boukerche et al., 2008]. Essa divisão visa diminuir a quantidade de mensagens que irá trafegar na rede, pois o usuário participante não terá que enviar mensagens para todos os participantes do ambiente virtual, mas somente para aqueles que pertencem à sua célula.

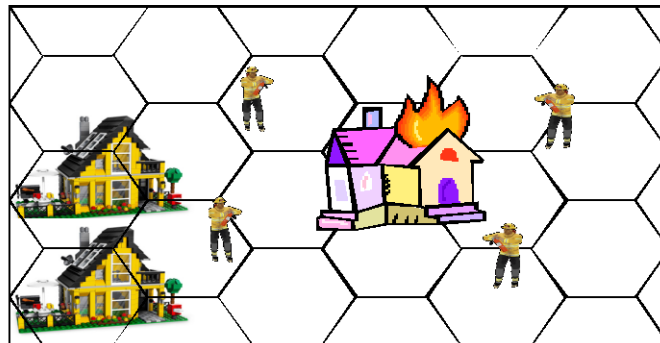


Figura 5-1 - Divisão do ambiente virtual em hexágonos iguais

Assumimos também que o usuário já possui um identificador único podendo ser, por exemplo, as coordenadas dadas pelo dispositivo de GPS (*Global Positioning System*). Essa identificação é utilizada para identificar a representação geométrica do usuário participante (avatar) tanto em relação ao ambiente virtual quanto à rede veicular. Na solução sendo proposta, todos os participantes são considerados confiáveis, isto é, não têm a intenção de prejudicar a simulação, e todo o ambiente virtual e seus componentes já serão de posse do usuário, isto é, o veículo já possui tanto o hardware como o software para executar o ambiente virtual.

O usuário, ao entrar em um AVC, engatilha o envio de uma requisição de entrada no ambiente para a rede *overlay*, que interpreta essa requisição. Posteriormente, essa requisição é mapeada em uma mensagem que é transmitida via rede *overlay* e rede veicular, para todos os outros participantes. Ao chegar aos outros participantes, essa mensagem é desempacotada pelas camadas de redes veicular e *overlay*. Se a mensagem for relacionada à simulação alvo, são passadas para a aplicação quais são as zonas ativas, ou seja, regiões onde há participantes. Com as informações das regiões o usuário seleciona qual é a região em que ele deseja entrar. Ao selecionar a zona, a aplicação emite uma solicitação à rede *overlay*, que interpreta essa requisição e a envia. O usuário, ao mover-se no ambiente, gera uma mensagem de atualização contendo o seu estado atual, sua posição, sua direção, e o identificador que é enviado a todos os usuários participantes da simulação. A aplicação não suporta, ainda, persistência, isto é, caso o usuário tenha saído e queira voltar, o estado do sistema não será mantido. A Figura 5-2 ilustra a interação entre as camadas de rede veicular, *overlay* e aplicação.

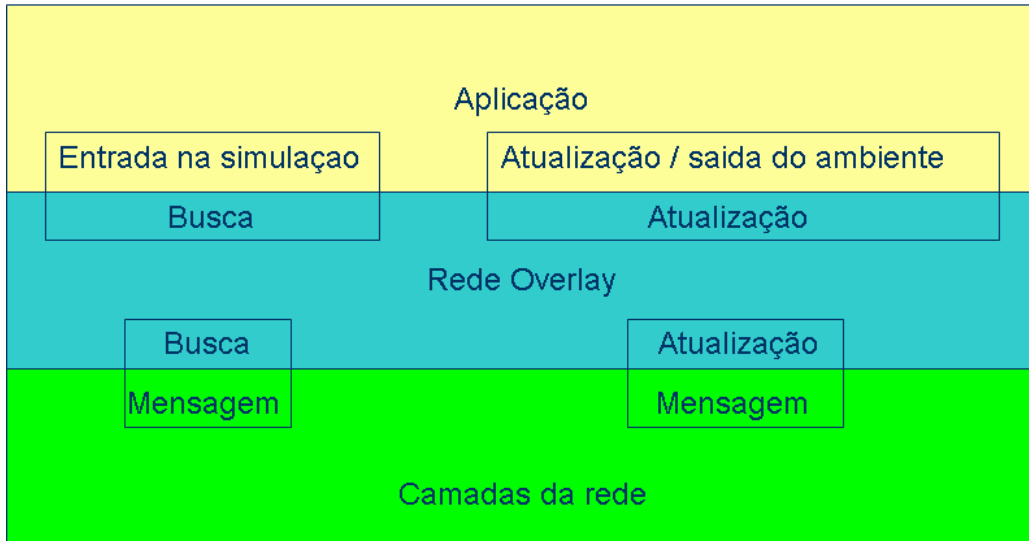


Figura 5-2 - Interação entre a aplicação (simulação) a rede overlay e as camadas de rede

5.1.1 Utilização do Jogo DukeBeanEm como AVC integrado à rede overlay

Neste trabalho foi utilizado o jogo DukeBeanEm [DukeBeanEm, 2009] como exemplo de AVC para a avaliação de nossa rede *overlay*, pois este é simples e mais fácil de adaptação para integração com a rede *overlay*. Esse jogo foi baseado no Dukenuke, onde o avatar tem como objetivo matar o seu oponente. A estrutura do jogo DukeBeanEm foi modificada para se integrar à rede *overlay* projetada (descrita na próxima seção). Esse ambiente virtual, desenvolvido pelos laboratórios da Sun, foi escrito em java utilizando o blender [blender, 2009], que é um programa de código aberto para modelagem, animação, texturização, composição, renderização, edição de vídeo e criação de aplicações interativas em 3D (utilizado para a criação dos avatares e dos objetos 3D), além do editor de imagens

gimp [gimp, 2009] para a geração das texturas de fundo, do chão e dos pilares do mundo virtual.

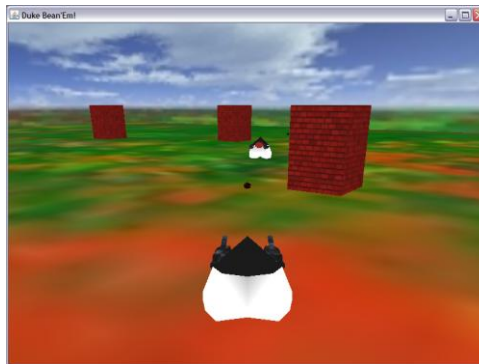


Figura 5-3 -Cena de Jogo DukeBeanEm

O DukeBeanEm é um jogo com uma representação geométrica simples e o *avatar* tem a mesma representação para todos os usuários, como pode ser visto na figura 5-3. Para a avaliação de nossa rede *overlay*, não foi implementado controle de zona no DukeBeanEm. Ou seja, todo o mundo virtual está somente em uma única zona, sendo utilizado como teste para a rede *overlay*.

Para dar suporte a esses ambientes virtuais na próxima seção veremos um suporte *overlay* para que esses sistemas possam ser executados em redes veiculares.

5.2 Descrição do funcionamento da Rede Overlay Proposta (Gnutella modificada)

Para se iniciar a rede é necessária a entrada de um usuário participante em uma determinada simulação ou jogo que possui um identificador único, como, por exemplo, o identificador DUKEEN, utilizado em nosso trabalho para identificar o jogo DukeBeanEm. O usuário entra no processo de busca de usuários para verificar se tem algum participante

ativo. Para isso, é enviada uma mensagem `ping_avc`, através de um broadcast, responsável também por realizar a primeira conexão da rede - essa primeira conexão irá qualificar o protocolo `gnutella`. Caso o usuário tenha uma zona favorita, poderá buscar essa zona, habilitando a identificação da zona.

Cada participante ativo, ao receber o `ping_avc`, verifica a distância entre ele e a origem para analisar se irá aceitar essa primeira conexão ou não, e envia um `pong_avc` como resposta. Ao aceitar a conexão, o participante passa a ser o coordenador para aquele usuário e adiciona esse novo participante à sua tabela de vizinhos com o identificador do usuário, tempo de envio de mensagem e à área de interesse. A qualidade da conexão é dada pelo tempo de resposta das informações. Caso esse tempo comece a extrapolar um intervalo de aceitação t o nó do usuário começa a buscar novas conexões.

Após esse processo de seleção do cenário da simulação e busca de usuários, é mostrado para o participante quais são as zonas ativas e o número de participantes que estão atuando naquelas zonas. Ao selecionar uma zona, uma solicitação de entrada é enviada na simulação. Essa solicitação é mapeada em uma `query_avc` que é enviada para o coordenador correspondente, ou seja, para o nó a que o usuário solicitante está conectado. Ao receber uma `query_avc`, esse nó verifica se está participando daquela zona ou se tem conhecimento de algum nó que está participando e responde com uma `queryhit_avc` (com informações sobre a zona), preenche a sua tabela de zona, que armazena as informações sobre os nós que estão participando na zona como pode ser visto na figura 5-10. O nó origem, ao receber a `queryhit_avc`, envia uma mensagem HTTP para os outros participantes da mesma zona que iniciarão a transferência das informações das atualizações do ambiente (coordenadas de localização, direção, velocidade e uma flag que informará o estado do usuário). Caso o usuário queira deixar a simulação ele modifica essa flag e deixa o ambiente virtual.

Se um nó desejar mudar de uma zona para outra ele verifica novamente em sua tabela de vizinhos se algum nó já está participando e envia uma query_avc direcionada para ele. Caso não exista nenhum participante, ele envia essa mensagem para o seu líder e, ao receber uma queryhit_avc nova ele apenas fecha a conexão anterior.

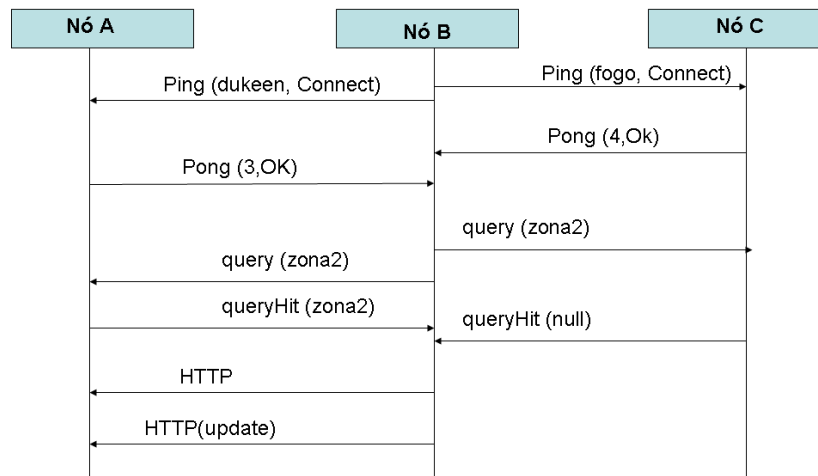


Figura 5-4 - Diagrama de seqüência do funcionamento da rede overlay

A verificação dos nós ativos e inativos da rede é feita através do envio das mensagens ping_avc e pong_avc, da mesma forma que é feita no protocolo gnutella padrão. Porém, o ping_avc só é enviado para os nós pertencentes à mesma área de interesse (zona) como podemos ver na figura 5-11. Se o tempo do pong_avc estiver com um atraso muito grande e não receber nenhuma confirmação de nova conexão, uma informação de alerta é mostrada para o usuário informando que ele está saindo da cobertura da rede. Caso esse tempo ultrapasse um intervalo t, esse usuário passa a não ter mais nenhuma colaboração na rede até se reconectar à rede.

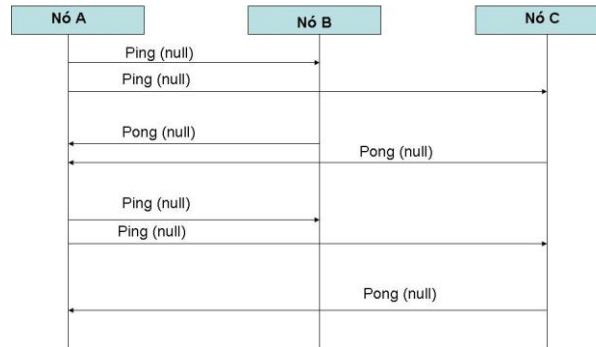


Figura 5-5 - Diagrama de Seqüência para a verificação de nós ativos e inativos na rede.

Se o usuário deseja sair da aplicação, ele modifica a flag de estado e envia essa informação para somente os participantes que pertence a sua zona, avisando sua saída. Posteriormente fecha a conexão e realiza a saída do usuário. A desconexão abrupta, isto é, quando há uma falha de hardware ou de conexão, não é tratada neste trabalho – nesse caso seria necessária a existência de apoio de infra-estrutura externa para que se possa armazenar o estado do ambiente. Para esse caso poderá ser usada uma estrutura externa como um ponto de acesso ou com a utilização da tecnologia de celular como o 3G. A superação desse desafio será tratada em trabalho futuro.

5.3 Projeto de Rede *Overlay* de suporte a AVCs em redes Veiculares

A rede *overlay* proposta baseia-se na rede Gnutella [Gnutella, 2009] pelas seguintes razões: Gnutella é software de fonte livre [Michael et al., 1994] e é amplamente utilizada por pesquisadores em todo mundo [Boukerche et al., 2008]. Neste projeto, a rede Gnutella foi modificada para melhor atender os ambientes virtuais. Essas modificações foram necessárias, pois como o ambiente virtual pode requerer freqüentes trocas de informações para manutenção de consistência, o tempo de entrega de informação pode impactar no

desempenho do sistema e, conseqüentemente, na qualidade da experiência do usuário no AVC. Em uma rede gnutella padrão, sob condições de alta mobilidade dos nós, bem como alta velocidade de transmissão, alta latência pode ser gerada. Para diminuir essa latência foi necessária a modificação de suas mensagens, bem como de seu processo de roteamento, para diminuir a quantidade de mensagens na rede e aumentar o dinamismo da comunicação entre o ambiente virtual (aplicação) e a rede *overlay*.

As seguintes alterações foram realizadas: novos campos foram adicionados ao cabeçalho de algumas mensagens; campos existentes foram alterados para melhor atender as necessidades de AVCS, isto é, diminuir o tempo de entrega das mensagens, bem como reduzir o tempo de estabelecimento de conexão na rede.

Na Figura 5-4 (a) são representadas as mensagens originais da rede Gnutella (ping, pong, query e queryhit). Na Figura 5-4 (b) são mostradas as mensagens modificadas (ping_ave, pong_ave, query_ave e queryhit_ave).

Ping		Ping	
Campo	opcional Ping Data	Campo	<Id_simulação, Id_zona, string>
Byte	0..L-1	Byte	0..L-1

Pong						Pong					
Campo	Porta	id	N. arquivos compartilhado	N Kilobytes compartilhado	opcional Pong Data	Campo	Porta	id	N. arquivos compartilhado	Área de interesse	string
Byte	0..1	2..5	6..9	10..13	14..L-1	Byte	0..1	2..5	6..9	10..13	14..L-1

Query					Query				
Campo	Velocidade	critério de busca	NUL (0x00) Terminador	(Opcional) Query Data	Campo	Velocidade	critério de busca	NUL (0x00) Terminador	visitas
Byte	0..1	2..N	N+1	N+2..L-1	Byte	0..1	2..N	N+1	N+2..L-1

QueryHit							QueryHit						
Campo	Arq. Index	Arq. Tam.	Nome do arquivo	NUL (0x00) Terminador	Opcional Result Data	NUL (0x00) Terminador	Campo	arq Index	arq tam.	Nome do arquivo	NUL (0x00) Terminador	part	NUL Terminador
Byte	0..3	4..7	8..7+K	8+K	9+K..R-2	R-1	Byte	0..3	4..7	8..7+K	8+K	9+K..R-2	R-1

Figura 5-6 - (a) mensagens originais da rede Gnutella; (b) mensagens modificadas baseadas na rede Gnutella

As mensagens alteradas são descritas a seguir:

Mensagem Ping_avc

Foram adicionados no campo opcional da mensagem ***ping*** a identificação da simulação, identificação da zona e a string de conexão. A mensagem modificada fará também a primeira conexão na rede, assim não necessita do envio de outra mensagem, ou seja, o próprio ping_avc qualificará o protocolo, em outras palavras, verificará qual é o protocolo de comunicação utilizado.

Campo	Id_simulação	Id_zona	String
Byte			0...L-1

Figura 5-7 -Mensagem ping_avc

Onde:

Id_Simulação: é a identificação da simulação

Id_Zona: é a identificação da zona

String: é a string de conexão(qualificação do protocolo)

Mensagem Pong_avc

Foi adicionada no campo opcional da mensagem ***pong*** a string de conexão que trará a confirmação da conexão, e modificamos o campo ***número de Kbyte*** compartilhado para área de interesse, pois identificará a zona atual do usuário como também quais serão as prováveis zonas de interesse desse nó.

Campo	Porta	id	N. arquivos compartilhado	Área de interesse	string
Byte	0...1	2...5	6...9	10...13	14...L-1

Figura 5-8 - Mensagem pong_avc

Onde:

Porta: porta de comunicação.

Id: é a identificação do participante

N. de arquivos compartilhados: número de zonas que está compartilhando

Área de Interesse: regiões em que está participando

String: é a string de resposta da conexão (qualificação do protocolo)

Mensagem Query_ave

Foi adicionado no campo opcional da mensagem **query** o campo de nós vistoriados, que diminuirá a quantidade de **query** enviadas na rede, conseqüentemente não sobrecarregando a rede com mensagens.

Campo	Velocidade	critério de busca	NUL (0x00) Termina	vistoriadas
Byte	0..1	2..N	N+1	N+2..L-1

Figura 5-9 - Mensagem query_ave

Onde:

Velocidade: velocidade de transmissão.

Critério de busca: identificação da zona que será buscada

Visitadas: identificação dos nós que já foram consultados

Mensagem QueryHit_ave

Foram adicionados no campo opcional da mensagem **queryhit** o campo de participante, que indica quais serão os outros participantes da zona desejada.

Campo	arq Index	arq tam.	Nome do arquivo	NUL (0x0) Terminator	part	NUL Termin o
Byte	0...3	4..7	8..7+K	8+K	9+K...R-2	R-1

Figura 5-10 - Mensagem queryhit_avc

Onde:

arq index: indica o início do arquivo.

arq. tam: tamanho da mensagem que irá trafegar

Nome do arquivo: nome da simulação

Part: identificação dos participantes daquela zona

O roteamento da rede gnutella padrão também foi modificado. As mensagens modificadas de resposta, como *queryhit_avc* e *pong_avc*, podem não realizar o mesmo caminho de volta realizado pelas mensagens correspondentes da Gnutella padrão conforme pode ser visto na figura 5-9.

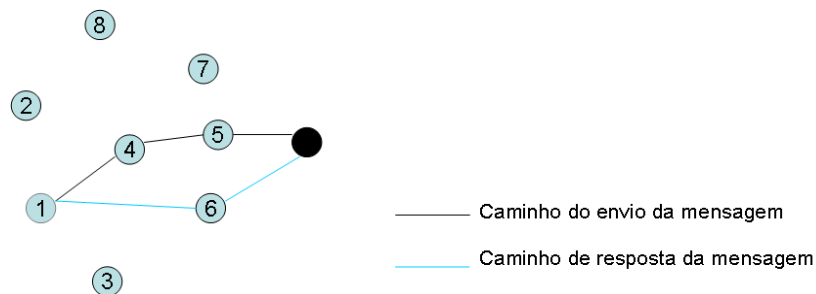


Figura 5-11 - roteamento gnutella modificada

5.4 Simulador da Rede

Para o desenvolvimento deste suporte *overlay* juntamente com a adaptação do ambiente virtual colaborativo utilizamos o simulador jist/SWANS, pois além de este ser

escrito em java, possibilitando um aprendizado rápido, o simulador mostrou ter um melhor desempenho que outros simuladores[JiST / SWANS, 2009].

Barr e colaboradores [Barr et al, 2005] utilizando um processador de 2GHz com 2 Gb de memória RAM executando um *benchmark* para medir a quantidade de memória utilizada bem como o *throughput* onde foram variado o número de nós na rede, sendo 500, 5.000, 50.000, 1.000.000, demonstraram que o simulador jist/swans tem um melhor desempenho que o ns2 [ns2, 2009] e o Parsec/ GloMoSim[GloMoSim, 2009]. O Jist/SWANS mostrou que ocupou menos memória que os demais consequentemente possibilitando uma quantidade maior de nós a serem simulados.

A nova versão do Jist/Swans ++ já contém integrado o modelo de mobilidade STRAW (STreetRAndom Waypoint) que simula a mobilidade dos veículos, onde o nó utiliza um modelo de tráfego baseado em mapas de dados reais fornecido por órgãos que analisam o comportamento de tráfego das cidades dos Estados Unidos, ou seja, através de senso de tráfego de veículo, com isso dando mais realismo à simulação, sendo uns dos principais critérios para a seleção do modelo de mobilidade.

A escolha deste modelo é pela praticidade, por já estar acoplado com o simulador, com isso, gerando ganho de tempo na criação da implementação, bem como na facilidade de utilizar este modelo, pois é escrito em java. Diferentemente de outros modelos, como SUMO (Simulation of Urban MObility)[SUMO, 2009] e o VanetMobiSim[VanetMobiSim, 2009] , onde é necessário realizar uma adequação para que o simulador possa utilizar os modelos, demandando tempo para essa tarefa.

Na próxima seção veremos as métricas e o cenário de simulação usados para a extração dos resultados.

5.5 Experimentos de Simulação

A solução proposta neste trabalho, chamada de **gnutella_avc** é comparada a uma rede gnutella padrão. O objetivo principal dessa comparação é avaliar o desempenho da gnutella_avc e a gnutella padrão segundo métricas descritas nas seções abaixo, utilizando o simulador jist/swans++[SWANS++, 2009].

5.5.1 Métricas Utilizadas

Pelo menos duas métricas importantes foram consideradas para avaliação da rede Overlay proposta e implementada:

- (i) latência – é o tempo que a mensagem leva da saída de um nó origem até chegar ao um nó destino – é possível com essa métrica calcular o tempo médio para o envio das mensagens.
- (ii) inconsistência espaço-temporal relativa – é a inconsistência da visualização dos estados dos objetos no ambiente virtual.

Para avaliarmos o impacto sofrido na visualização do ambiente virtual, utilizamos a relação da inconsistência espaço-temporal de um ambiente virtual, métrica concebida por pesquisadores em [Zhou et al., 2004], que é dada pela equação:

$$\omega = \frac{\max\{\Omega\}}{\varepsilon * \sigma} \quad (1)$$

Onde:

- ω - Inconsistência espaço-temporal relativa

- σ - Tempo da percepção visual humana para informações espaciais. Os autores [36,37] acreditam que σ esteja entre 0.1 a 0.2 segundos. Nesse trabalho adotamos que σ vale 0.15 segundos.
- Ω - Inconsistência espaço-temporal
- ε - Espaço mínimo entre os participantes

Essa métrica evidencia o impacto na consistência do mundo virtual, sendo diretamente influenciada pelo atraso na entrega das informações de atualização do ambiente virtual, bem como pelo tempo de envio das atualizações geradas pelas entidades participantes do ambiente.

5.5.2 *Cenário de Simulação*

O texto abaixo descreve os ambientes de aplicação e de rede que foram utilizados para a simulação.

Ambiente da aplicação

Primeiramente utilizamos somente o modelo matemático para a obtenção dos resultados, onde assumimos que os *avatares* possuem uma velocidade máxima 2m/s e ao esbarrar em um 0,3m/s. Com isso, obteremos a inconsistência espaço-temporal.

Para a criação do modelo matemático como aplicação foi necessário a utilização das equações [Zhou et al., 2004]:

$$\Omega_1 = \bar{v} \cdot \gamma \cdot T_d + \delta \cdot T_d + |a|_{max} \cdot T_{DR} \cdot \frac{1}{2} T_d^2 \quad (2)$$

E

$$\Omega_2 = (\bar{v} \cdot \gamma + \frac{\delta}{2}) \cdot (T_{DR} - T_d) \quad (3)$$

Que calculam a inconsistência no momento de envio (Ω_1) e recebimento (Ω_2), respectivamente, das mensagens de atualização do ambiente, onde:

\bar{v} - Velocidade média dos participantes

γ - Diferença entre o *clock* do ambiente virtual de cada participante

δ - Diferença do posicionamento do avatar

a - Velocidade máxima dos participantes

T_{DR} - média de intervalo de atualização do DR, dado pela equação $T_{DR} = \sqrt{\frac{2\delta}{|a|}}$

T_d - Latência.

Após a soma das duas inconsistência, dado pela equação (4), podemos facilmente obter a inconsistência espaço-temporal relativa equação 1.

$$\Omega = \Omega_1 + \Omega_2 \quad (4)$$

Para essa primeira fase utilizamos os parâmetros listados na Tabela 5-1, juntamente com o tempo do envio das mensagens de atualização pela rede onde obtemos o valor de T_d .

Tabela 5-1 - Parâmetros de Simulação da aplicação

<i>Aplicação</i>	
Espaço mínimo entre os participantes	0,1m
Velocidade media dos participantes	0,3 m/s
Diferença entre o clock do ambiente virtual de cada participante	0,01 s
Diferença do posicionamento do avatar (δ)	0,3 m
Velocidade máxima do avatar	2 m/s
Participantes	15 entidades

Ao mesmo tempo em que eram enviadas as mensagens de atualização era realizada a verificação de tempo em tempo dos nós ativos da rede overlay, ou seja, eram enviadas as mensagens ping_avc e pong_avc para verificar os nós ativos e inativos e para fornecer uma melhor forma de se transferir a mensagem de atualização, acarretando com isso um aumento no número de mensagens da rede.

Com o jogo DukeBeanEm foi possível constatar o impacto visual sofrido pela nossa solução. Este jogo foi adaptado para que pudesse ser integrado à nossa rede overlay dentro do simulador jist.

Ambiente da Rede Ad Hoc Veicular

Para a realização da avaliação, os parâmetros de rede listados na Tabela 5-2 foram assumidos antes do início da simulação.

Tabela 5-2 - Parâmetros da Simulação de rede Vanet

Rede	
Modelo de Mobilidade	random waypoint
Número de nós na rede veicular	150 veículos
Número de nós na rede overlay	5-10-15 participantes
Velocidade máxima	20 Km/h
Velocidade mínima	1 Km/h
Duração da simulação	5 minutos

Camada Física

Adotamos para cada nó na camada física o uso do protocolo DSRC que é uma adaptação da IEEE 802.11a PHY baseada na tecnologia de *orthogonal frequency division multiplex* (OFDM). Além disso, a camada de controle de acesso múltiplo (MAC) do DSRC é muito semelhante à do IEEE 802,11 MAC, baseado no *carrier sense multiple access* com *collision avoidance* (CSMA / CA) [Yin et al., 2004].

Camada de Rede

Na camada de rede utilizamos as coordenadas do GPS como um identificador do nó e o protocolo de roteamento GPSR, pois é um protocolo baseado em posição e trata-se de um tipo de protocolo identificado com um promissor paradigma para as redes Vanet's [Li e Wang, 2007].

Camada de Transporte

Na camada de transporte utilizamos o UDP, pois o tipo da aplicação não requer uma mensagem de confirmação, conseqüentemente o uso do UDP se torna viável. Pelo fato de não precisar de um ACK, há uma diminuição na quantidade de mensagem na rede.

Na próxima seção serão discutidos os resultados obtidos pela simulação, mostrando a comparação entre a gnutella normal e a nossa solução de gnutella modificada.

5.5.3 Resultados Obtidos

As métricas avaliadas foram a latência e a inconsistência espaço-temporal do ambiente virtual. O protocolo gnutella padrão foi comparado à versão gnutella modificada.

Foram realizados vários testes, sendo executadas 10 vezes cada teste, onde cada nó da rede overlay suportava no máximo 4 conexões ao mesmo tempo.

Para os primeiros testes variamos o número de entidades em 5, 10, 15 e tendo em cada simulação uma duração de 300 segundos (5 minutos), objetivou-se avaliar a média da latência gerada por essas mensagens de atualização (figura 5-12), bem como a inconsistência espaço-temporal da visualização do ambiente virtual visto na figura 5-13.

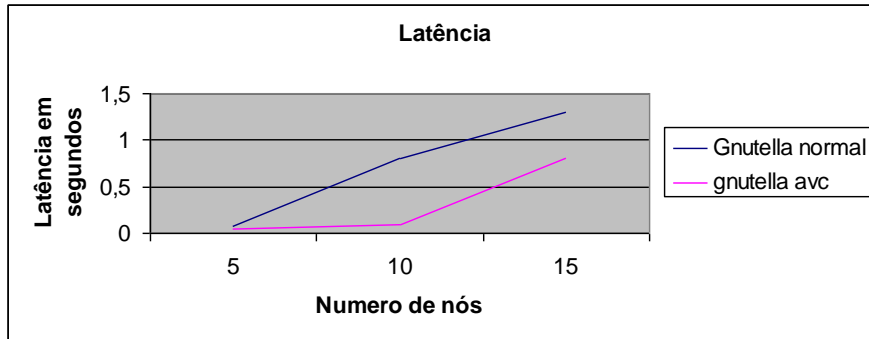


Figura 5-12 - Gráfico da latência gerada pela simulação

Na figura 5-12 podemos observar que a gnutella AVC gerou uma latência menor que a gnutella normal devido a liberdade de roteamento e as mudanças realizadas nas mensagens para a realização da troca de zona no ambiente virtual. Embora a latência da gnutella AVC tenha aumentado com o acréscimo de novos usuários (15 entidades participantes), a gnutella AVC se manteve abaixo da gnutella padrão. Esse aumento na latência foi decorrente do crescimento exponencial de mensagens enviadas para rede. Esses atrasos na entrega das informações, bem como a frequência de envio de informações de atualização têm um impacto diretamente proporcional na inconsistência espaço-temporal como podemos ver na figura 5-13.

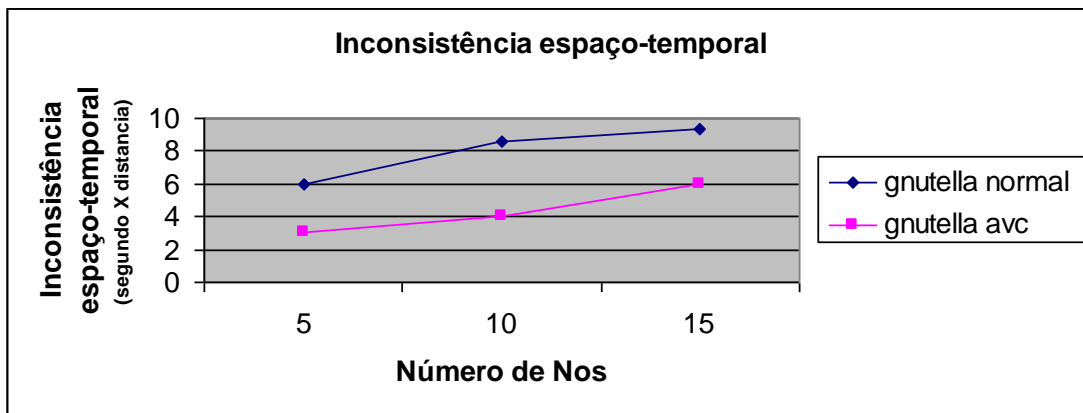


Figura 5-13 - Gráfico da inconsistência espaço-temporal da visualização do ambiente virtual.

A figura 5-13 mostrou que a inconsistência gerada pela gnutella normal foi maior que a gnutella AVC, pois a inconsistência está diretamente ligada à latência gerada pela rede. Podemos observar essa influência analisando a simulação com 10 entidades, na qual foi gerado um pico de latência, conseqüentemente afetando a sua consistência. Embora essa influência seja inerente a todos os ambientes virtuais, a gnutella AVC teve um desempenho melhor que a gnutella normal.

Outro critério avaliado foi a frequência de envio de pacotes na rede para se obter a latência, como podemos ver pela figura 5-14, e a inconsistência vista na figura 5-15, sendo utilizadas 15 entidades do ambiente virtual. Houve uma variação no valor de δ que influenciou a frequência de envio de informação de atualização do ambiente.

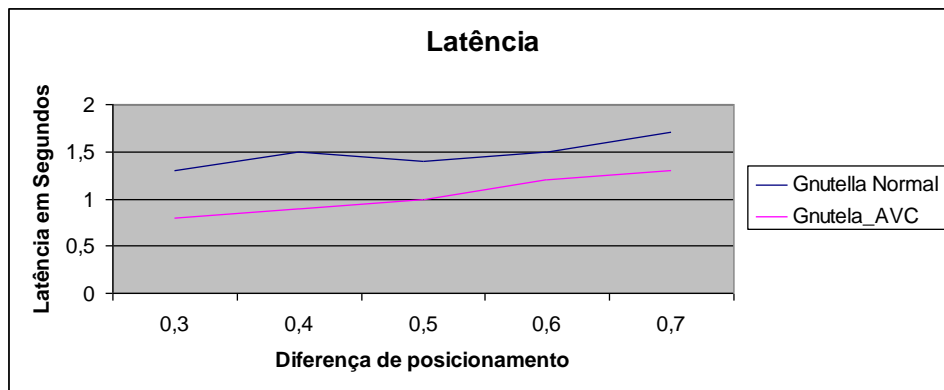


Figura 5-14 Gráfico da latência gerada pela simulação em função da diferença do posicionamento real e simulado.

No gráfico da figura 5-14 podemos observar que quanto maior o valor de δ , ou seja, quanto maior o espaçamento do envio da mensagem, maior a latência gerada devido ao posicionamento físico do veículo ao enviar e ao receber mensagens, sendo que no momento de envio da mensagem o nó origem poderá estar distante do nó destino nó. Essa distância entre os veículos ocasionou picos na gnutella padrão, levando a um pior resultado na consistência do ambiente, como podemos ver na figura 5-15.

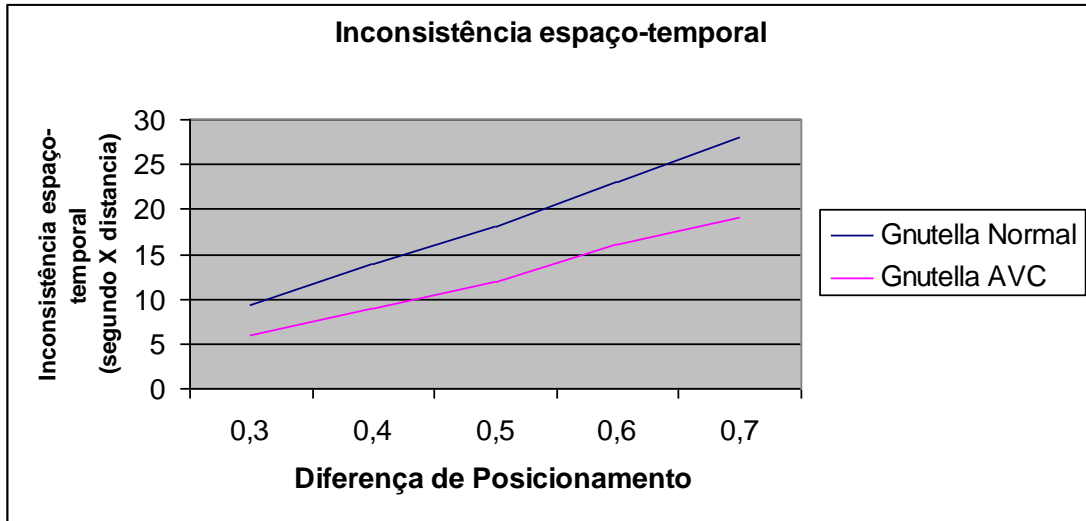


Figura 5-15 - Gráfico da inconsistência espaço-temporal pela simulação em função da diferença do posicionamento real e simulado.

Observamos pela figura 5-15 que a inconsistência gerada pelos diferentes intervalos de envio de mensagem da gnutella AVC foi menor e sua reta teve um menor crescimento que a gnutella normal devido ao seu gerenciamento de zona. Esses valores podem ser melhorados conforme a estratégia utilizada no desenvolvimento do ambiente virtual, ou seja, com a adição de filtros para diminuir o fluxo de mensagem de atualização.

5.6 Considerações Finais

Nessa seção foi apresentada uma rede overlay (Gnutella AVC) de suporte a ambientes virtuais colaborativos em redes veiculares. Para isso, modificações foram realizadas nos cabeçalhos das principais mensagens da rede Gnutella padrão, bem como em sua forma de roteamento, para que pudesse ser obtido melhor desempenho para suporte a AVCS em redes VANETS.

Foram utilizadas métricas importantes para avaliar o impacto de nossa solução sobre os ambientes virtuais. Para avaliação da nossa Gnutella AVC, esta foi comparada à gnutella padrão. A nossa solução mostrou um melhor desempenho tanto em termos de latência como em termos de inconsistência espaço-temporal. O próximo capítulo tece as considerações finais.

6 Conclusões

Este capítulo descreve as contribuições geradas com a realização deste trabalho, bem como conclusões finais e trabalhos futuros.

6.1 Contribuições Geradas

Esse trabalho vem trazer suporte que irá auxiliar a superar os desafios de executar um ambiente virtual colaborativo sobre uma rede veicular, com isso, diminuindo a quantidade de mensagens na rede, bem como fornecendo uma abstração da complexidade da rede para os usuários. Para isso foi realizada a modificação das redes gnutella para a criação da rede *overlay*, para melhor atender a aplicação. Esse trabalho vem trazer mais uma aplicação voltada ao entretenimento para os passageiros dos veículos, sendo que ainda poucos trabalhos sobre essa classe de aplicação.

6.2 Conclusões Finais

Os ambientes virtuais colaborativos vêm se tornando cada vez mais acessíveis em dispositivos móveis, como dispositivos de mão, celular, entre outros, através de jogos e simulações, que visam principalmente o entretenimento e conforto do participante. O oferecimento de AVCs para usuários em veículos em movimento abriria novas oportunidades para diferentes classes de aplicações, dentre elas, simulações de treinamento de equipes de preparação e resposta a emergência. Os AVCs requerem baixa latência e escalabilidade, para que um grande número de participantes possa interagir e colaborar em um ambiente compartilhado e consistente sem perder qualidade. Por outro lado, as características singulares de redes veiculares, tais como alta mobilidade dos nós, curto tempo de transmissão e topologia altamente dinâmica provocam desafios que precisam ser superados se AVCs forem oferecidos em rede VANETs.

Neste trabalho foi apresentada uma rede overlay de suporte a AVCs em redes VANETs, baseada no protocolo padrão gnutella, que suporta AVCs em redes veiculares. Várias alterações foram feitas à rede Gnutella padrão, dentre elas modificações no cabeçalho de suas principais mensagens, bem como em seu roteamento. Esta rede modificada foi denominada Gnutella AVC. Para a avaliação desta rede foram utilizadas métricas importantes, tais como latência e inconsistência espaço-temporal. Foi utilizado o simulador de redes VANETs Jist/SWANS. A aplicação utilizada foi o jogo DukeBeans com diferentes cenários. Os resultados obtidos mostraram que a gnutella AVC teve um melhor desempenho que a gnutella padrão, com menor latência e menor inconsistência espaço-temporal, requisitos importantes de AVCs.

6.3 Trabalhos Futuros

As seguintes questões serão tratadas como trabalhos futuros:

- Avaliação de nossa solução com um número maior de usuários participantes;
- Avaliação de nossa solução em redes VANETs com diferentes modelos de mobilidade;
- Integração (à rede Gnutella AVC) e avaliação de AVCs para treinamento de equipes de emergência sendo construídos pelo laboratório LRVNet (Laboratorio de Realidade Virtual em Rede - local em que este trabalho de mestrado foi realizado);
- Implementação de persistência nos AVCs.

7 Referências Bibliográficas

Abuelela Mahmoud; Olariu Stephan "POSTER: Traffic-Adaptive Packet Relaying in VANET", 2007

Ahmed S. and Kanhere S. S.: VANETCODE: a network coding to enhance cooperative downloading in vehicular ad-hoc networks, Proc. ACM International Conference on Communications and Mobile Computing, Vancouver, British Columbia, (2006).

ANARG - Advanced Network Architecture Research Group
<http://www.anarg.jp/research-e.html> acessa em 2009

Andersen D. et al., "Resilient Overlay Networks," ACM Symp. Operating Syst. Principles, Chateau Lake Louise, Banff, Canada, pp. 131–45, Oct. 2001.

Androutsellis-Theotokis S.; Spinellis, D. A survey of Peer-to-peer Content Distribution Technologies. ACM Computing Survey (CSUR) v. 36 Issue 4, p. 335-371, December 2004.

Armstrong L. Dedicated short range Communications (DSRC) at 5.9 GHZ, Armstrong Consulting, disponível em: www.learnstrong.com/DSRCHome/DSRCHomeset.htm
Consultado em 03/03/2009

Artimy, M. M.;Robertson, W.;Phillips " Connectivity with Static Transmission Range in Vehicular Ad Hoc Networks", 3rd Annual Communication Networks and Services Research Conference (CNSR),2005.

Artimy, M. M.;Robertson, W.;Phillips, W. J. "Connectivity in Inter-Vehicle Ad Hoc Networks", CCGEI 2004, Niagara Falls, Maylmai, 2004.

Bachir A. e Benslimane A., "A multicast protocol in ad hoc networks intervehicle geocast," in Proceedings of the 57th IEEE Semiannual Vehicular Technology Conference, vol. 4, pp. 2456–2460, 2003.

Barr R., Haas Z. J. e Renesse R. V. Scalable Wireless Ad hoc Network Simulation. Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless, and

Peer-to-Peer Networks. Ch. 19, pp. 297-311, CRC Press, 2005.

Bechler M., Jaap S., e Wolf L., “An Optimized TCP for Internet Access of Vehicular Ad Hoc Networks,” Proc. Networking 2005, May 2–6, 2005.

Bender - <http://www.blender.org/>

Bhatia, K. Peer-to-peer Requirements On The Open Grid Services Architecture Framework. Technical report, Global Grid Forum (GGF) GFD.49, p.1-38, July 2005.

Blum J., Eskandarian A., e Hoffman L., “Mobility management in IVC networks,” in IEEE Intelligent Vehicles Symposium, 2003.

Borgonovo, F.; Capone, A.; Cesana, M.; Fratta, L. ”RR-ALOHA, a Reliable R-ALOHA broadcast channel for ad-hoc inter-vehicle communication networks”, Med-Hoc-Net 2002, Baia Chia, Italy, 2002.

Boukerche, A., Araújo, R.B., Laffranchi, M.M., A Hybrid Solution to Support Multiuser 3D Virtual Simulation Environments in Peer-to-Peer Networks, Proceedings of the Eighth IEEE International Symposium on Distributed Simulation and Real-Time Application (DS-RT 2004), p. 61-68. Budapest, Hungria, 21-23, Outubro, 2004.

Boukerche, A., McGraw, N. J., and Araujo, R. B.: A Grid-Filtered Region-Based Approach to Support Synchronization in Large-Scale Distributed Interactive Virtual Environments The 2005 International Convergence on Parallel Processing Workshops, 1530-2016, 2005

Boukerche, A.; Zarrad, A.; Araujo, R.B.; A Novel Optimized Caching Technique for Mobile Gnutella Based Network to Support Large-Scale Collaborative Virtual Environment Simulation Symposium, 2008. ANSS 2008. 41st Annual; 13-16; P:289 -

297, April 2008

Briesemeister, Schäfers L., e Hommel G., “Disseminating messages among highly mobile hosts based on inter-vehicle communication,” in Proceedings of the IEEE Intelligent Vehicles Symposium, pp. 522–527, 2000.

Bucciol P., Masala E. e Martin J. C. D., “Dynamic Packet Size Selection for 802.11 Inter-Vehicular Video Communications,” Proc.1st Int’l. Wksp. Vehicle-to-Vehicle Commun., 2005.

Carlsson C., Hagsand O.: DIVE- A Platform for Multi-User Virtual Environments. Computers and Graphics, USA, pp: 663-669 (1993)

Chen W., Cunha R. K., Known T. J., Lee J. e Hsu I. Y A Survey and Challenges in Routing and Data Dissemination in Vehicular Ad-hoc Networks Proceedings of the 2008 IEEE International Conference on Vehicular Electronics and Safety Columbus, OH, USA. September 22-24, 2008

Chen, A.; Khorashadi, B.; Ghosal, D.; Ghosal, D.; Chuah, C.; Impact of Transmission Power on TCP Performance in Vehicular Ad Hoc Networks, p. 65- 71, 2007

Chuang B.-W. et al., “System Development and Performance Investigation of Mobile Ad-Hoc Networks in Vehicular Environments,” Proc. IEEE Intelligent Vehicles Symp., pp. 302–07, 2005

Das T. K., Singh G., MitchellA., Kumar P. S. , & McGee K.: “NetEffect: A Network Architecture for Large- scale Multi-user Virtual Worlds”. Proceedings ACM Virtual Reality software and technology, USA, pp. 157–163, 1997

Doval D.; O’Mahony D.; Overlay Networks a scalable alternative for P2P IEEE

Computer Society, p. 79 – 82, August 2003

DukeBeanEm: <http://jsolutions.se/DukeBeanEm/>

Durresi M., Durresi A., e Barolli L., “Emergency broadcast protocol for intervehicle communications,” in ICPADS '05: Proceedings of the 11th International Conference on Parallel and Distributed Systems—Workshops (ICPADS'05), 2005

EverQuest Homepage: <http://www.everquest2.station.sony.com/>

Faisstnauer C.: Priority Scheduling for Networked Virtual Environments and Online Games, PhdThesis, Technische Universitat Wien, 2001

Falchi F., Gennaro C., e Zezula P. A Content-Addressable Network for Similarity Search in Metric Spaces, Springer-Verlag Berlin Heidelberg LNCS 4125, pp98-110, 2007

Festag A , Fußler H , Hartenstein H , Sarma A , Schmitz R , FLEETNET: BRINGING CAR-TO-CAR COMMUNICATION INTO THE REAL WORLD, 11th World Congress on ITS, Nagoya, Japan, October 2004

Freenet. <http://www.freenet.com>. acessado em 2009

Gang P. CDN: Content distribution network. Technical Report TR-125, Experimental Computer Systems Lab, 2003.

Ghandeharizadeh S. K. S. e Krishnamachari B., PAVAN: A policy framework for content availability in vehicular ad-hoc networks, Proc. First ACM Workshop on Vehicular Ad Hoc Networks, (VANET'2004), Philadelphia, PA, 2004

Gimp - <http://www.gimp.org/>

GloMoSim - global mobile information system simulation library

<http://pcl.cs.ucla.edu/projects/glomosim/>

Gnutella. <http://www.gnutella.com> acessado em 2009

Greenhalgh C. e Benford. S. Boundaries, awareness, and interaction in collaborative virtual environments. In Proceedings of the Sixth IEEE Workshop on Enabling Technologies (WETICE), pages 193–198, 1997

Guo, M.; Ammar, M. H.; Zegura, E. W. V3 A vehicle-to-vehicle live video streaming architecture. Elsevier. P. 404-424, September 2005.

Haas Z.J. e Pearlman M.R., “The zone routing protocol (ZRP) for ad hoc networks,” in Internet draft—Mobile Ad hoc NETWORKing (MANET), Working Group of the Internet Engineering Task Force (IETF), Nov. 1997.

Haerri J. and Cristian Bonet “a lower bound for vehicles trajectory direction” in Proc of the IEEE 62 Semiannual Vehicular Technology conference (VTCF all’05), Dallas, USA, September 2005

Hui, F.; Mohapatra, P.; Experimental Characterization of Multi-hop Communications in Vehicular Ad Hoc Network; VANET’05, p. 85- 86 Cologne, Germany; 2 September 2005

Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator – Jist/SWANS
<http://jist.ece.cornell.edu/>

Johnson D.B. and Maltz D.A., “Dynamic source routing in ad hoc wireless networks,”

in *Mobile Computing*, vol. 353, 1996

Joslin, C.; Pandzic, I. S.; Magnenat-Thalmann, N. Trends in Networked Collaborative Virtual Environments, *Computer Communications*, v. 26, I.5, p. 430 -437, March 2003

Karp B. e Kung H.T., “GPSR: Greedy perimeter stateless routing for wireless networks,” in *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, 2000

Khorashadi, B.; Chen, A.; Ghosal, D.; Chuah, C.; Zhang M. Impact of Transmission Power on the Performance of UDP in Vehicular Ad Hoc Networks; p. 3698- 3703; IEEE Communications Society, ICC 2007

Ko Y.B. e Vaidya N.H., “Location-aided routing (LAR) in mobile ad hoc networks,” *Wireless Networks*, vol. 6, no. 4, pp. 307–321, 2000.

Korkmaz G., Ekici E., Özgüner F., and Özgüner Ü., “Urban multi-hop broadcast protocol for inter-vehicle communication systems,” in *ACM International Workshop on Vehicular Ad Hoc Networks*, pp. 76–85, 2004.

LeBrun J., Chuah C.N., Ghosal D., e Zhang M., “Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks,” in *Proceedings of the IEEE Vehicular Technology Conference*, vol. 4, pp. 2289–2293, 2005.

Lee J., Chen W., Onishi R., e Vuyyuru R., Vehicle local peer group based multicasting protocol for vehicle-to-vehicle communications,. *The Fourth International Workshop on Vehicle-to-Vehicle Communications*, 2008.

Lee K. E., Lee S.-H., Cheung R., Lee U. and Gerla M.;First experience with CarTorrent in a real vehicular ad hoc network testbed, *Proc. IEEE Mobile Networking*

for Mobile Environments, (INFOCOM/MOVE), 2007.

Lee U., Park J.-S. and Gerla M.: CodeTorrent: a content distribution using network coding in VANET, Proc. First ACM Workshop on Decentralized Resource Sharing in Mobile Computing and Networking, 2006.

Lety E., Turetli T., Baccelli F, "SCORE: a scalable communication protocol for large-scale virtual environments". IEEE/ACM Transactions on Networking Journal, USA, pp 247-260, 2004

Li J. e Singh S., "ATCP: TCP for Mobile Ad Hoc Networks," IEEE JSAC, 2001.

Lin F., Wang Y. Routing in Vehicular Ad Hoc Networks: A Survey, IEEE VEHICULAR TECHNOLOGY MAGAZINE, JUNE 2007

Liu G., Lee B.S., Seet B. C., Foh C.H., Wong K.J., e Lee K. K., "A routing strategy for metropolis vehicular communications," in International Conference on Information Networking (ICOIN), pp. 134–143, 2004.

Living Worlds Proposal Draft 2. On-line: http://medialab.it.fht-esslingen.de/da/goetz/da_goetz/anhang/living-worlds_draft/draft_2/index.html, Abril, 1997

Lochert C., Hartenstein H., Tian J., Herrmann D., Füßler H., e Mauve M., "A routing strategy for vehicular ad hoc networks in city environments," in Proceedings of IEEE Intelligent Vehicles Symposium (IV2003), pp. 156–161, June 2003.

Lochert C., Mauve M., Füßler H., e Hartenstein H., "Geographic routing in city scenarios," ACM SIGMOBILE Mobile Computing and Communications Review

(MC2R), vol. 9, no. 1, pp. 69–72, Jan. 2005.

Lua, E. K.; Crowcroft, J.; Pias M.; Sharma R.; Lim S. A Survey and Comparison of Peer-to-peer Overlay Network Schemes, IEEE Communications Survey and Tutorial, Second Quarter 2005

Luo, J.; Hubaux, J-P.; " A Survey of Inter-Vehicle Communication", Technical Report, School of Computer and Communication Science, EPFL, Switzerland, 2004.

Macedonia, M. R. Zyda, M. J. Pratt, D. R. Brutzman, D. P. and Barham. P. T. Exploiting Reality with Multicast Groups. IEEE Computer Graphics and Applications, Vol. 15(5), pages 38–45, 1995.

Maihöfer C. e Eberhardt R., “Geocast in vehicular environments: Caching and transmission range control for improved efficiency,” in Proceedings of IEEE Intelligent Vehicles Symposium (IV), pp. 951–956, 2004.

Maihöfer C., “A survey of geocast routing protocols,” IEEE Communications Surveys & Tutorials, vol. 6, no. 2, pp. 32–42, 2004.

Menouar, H.; Filali, F. A Survey and Qualitative analysis of MAC protocols for Vehicular Ad Hoc Network, IEEE Wireless Communications; October 2006

Michael R. Macedonia, R, M. Zyda, J. R. David, R. Pratt “NPSNET: A NETWORK SOFTWARE ARCHITECTURE FOR LARGE SCALE VIRTUAL ENVIRONMENTS”, USA, pp: 256-287, Apl 1994.

Michael R. R, Macedonia M., Zyda J., David R., Pratt R.: NPSNET: A NETWORK SOFTWARE ARCHITECTURE FOR LARGE SCALE VIRTUAL

ENVIRONMENTS, USA, pp: 256-287 (1994)

Milojicic, D. S.; Kalogeraki, V.; Lukose, R.; Nagaraja, K; Pruyne, J.; Richard, B; Rollins, S.; Xu, Z. Peer to peer Computing. HP Laboratories Palo Alto, HPL-2002-57, 8 March, 2002

Mohapatra Prasant and Krishnamurthy S.. “Ad Hoc Networks: Technologies and protocols”. Springer Science and Business Media, Inc, Chapter 1, 2004

Morsink P., Hallouzi R., Dagli I., Cseh C., Schäfers L., Nelisse M., Bruin D., CARTALK 2000: DEVELOPMENT OF A COOPERATIVE ADAS BASED ON VEHICLE-TOVEHICLE COMMUNICATION, 10th World Congress and Exhibition on Intelligent Transport Systems and Services 16-20 November 2003, Madrid, Spain, 2003

Nadeem T. et al., “Trafficview: Traffic Data Dissemination Using Car-to-Car Communication,” ACM Mobile Comp. and Commun. Rev., vol. 8, no. 3, pp. 6–9, 2004.

Namboodiri V., Agarwal M., e Gao L., “A study on the feasibility of mobile gateways for vehicular ad-hoc networks,” in Proceedings of the First International Workshop on Vehicular Ad Hoc Networks, pp. 66–75, 2004

Nanda A.; Das, S.; Pau G.; Gerla M.; Sanadidi M. Y. Co-operative Downloading in Vehicular Ad Hoc Wireless Networks Wireless On-demand Network Systems and Services, 2005

ns2 - The Network Simulator <http://www.isi.edu/nsnam/ns/>

Olariu, S. Peer-to-peer Multimedia Content revisioning for Vehicular Ad Hoc

Networks Proceedings of the 3rd ACM workshop on Wireless multimedia, 2007

Ooi C.C. e Faisal N., "Implementation of geocast-enhanced AODV-bis routing protocol in MANET," in Proceedings of the IEEE Region 10 Conference, vol. 2, pp. 660–663, 2004.

Pandžic S., Tolga K., Elwin, L., Thalmann, N. e Thalmann. D. "A flexible architecture for Virtual Humans in Networked Collaborative Virtual Environments". Proceedings Eurographics, Hungary, pp: 143-152, 1997

Perkins C.E. and Royer E.M., "Ad-hoc on demand distance vector routing," in Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pp. 90–100, Feb. 1999.

Robert D. e Wolff R Controlling Consistency within Collaborative Virtual Environments, Eighth IEEE International Symposium on Distributed Simulation and Real-Time Applications (DS-RT'04). 2004

Rueda S., Morillo P., Ordu J.M.: On the Characterization of Peer-To-Peer Distributed Virtual Environments, IEEE Virtual Reality Conference 2007, Charlotte, North Carolina, USA (2007)

Santos R.A., Edwards A., Edwards R., e Seed L., "Performance evaluation of routing protocols in vehicular adhoc networks," The International Journal of Ad Hoc and Ubiquitous Computing, vol. 1, no. 1/2, pp. 80–91, 2005.

Schmitz R. et al., "Analysis of Path Characteristics and Transport Protocol Design in Vehicular Ad Hoc Networks," Proc. IEEE 63rd VTC, pp. 528–32, 2006.

Schollmeier, R. A. Definition of Peer-to-peer Networking for the classification of Peer-

to-peer Architecture and Applications. First International Conference on Peer-to-peer Computing Suécia, 27-29 August, 2001

Sengupta, R.; Xu, Q. DSRC for Safety Systems; An Automated Snowblower for Highway Winter Operations; 10v; 4n 2004

Shi S. & Turner J.. Routing in overlay multicast networks. In IEEE Infocom 2002, New York, USA, June 2002. IEEE.

SICHITIU M. L., Kihl M., INTER-VEHICLE COMMUNICATION SYSTEMS: A SURVEY, 2ND QUARTER 2008, VOLUME 10, NO. 2, 2008

Singh, G., Serra L., Png, W. and Ng. H. "BrickNet: A Software Toolkit for Network-Based Virtual Worlds". Presence: Teleoperators and Virtual Environments, 3(1), USA, pp19-34, 1994.

Singhal, S. e Zyda, M. Networked Virtual Environments: Design and Implementation, (Siggraph Series), ACM Press, New York, 1999

Smallworld Homepage: <http://www.smallworlds.com/>

Stemm M., Katz R. H. Vertical handoffs in wireless overlay networks. Mobile Networks and Applications 3, p 335-350, (1998)

Sterling B. Turning cars into wireless networks nodes Disponível em URL: http://blog.wired.com/sterling/2007/06/turning_cars_in.html

Consultado: 20/02/2009

Stott, J. A., "The Effects of Frequency Errors in OFDM," BBC Research Department

Report No RD 1995/15 1995.

STRAW - STreet RANdom Waypoint - vehicular mobility model for network simulations

Stutzbach D., Rejaie R., Sen S., Characterizing Unstructured Overlay Topologies in Modern P2P File-Sharing Systems, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 16, NO. 2, APRIL 2008

Sugiura A. e Dermawan C. , “In traffic jam IVC-RVC system for ITS using Bluetooth,” IEEE Trans. Intelligent Transportation Sys., vol. 6, no. 3, pp. 302–13, 2005.

SUMO - Simulation of Urban MObility <http://sumo.sourceforge.net/>

Sun M., Feng W., Lai T.H., Yamada K., Okada H., e Fujimura K., “GPS-based message broadcasting for inter-vehicle communication,” in ICPP’00: Proceedings of the 2000 International Conference on Parallel Processing, 2000

SWANS++ - Extensions to the Scalable Wireless Ad-hoc Network Simulator

<http://www.aqualab.cs.northwestern.edu/projects/swans++/>

The FCC DSRC, disponível em <http://wireless.fcc.gov/services/its/dsrc/>. Consultado em 20/03/2009

Toor Y., Mühlethaler P., Laouiti A., Fortelle A. L, Vehicle Ad Hoc Networks: Applications and Related Technical Issues, IEEE Communications Surveys & Tutorials 3rd v. 10, n. 3, pp 74-88 (2008)

Transier M., Fubler H., Widmer J., Mauve M., e Effelsberg W., Scalable position-based multicast for mobile adhoc networks,. Proc. of the First International Workshop on Broadband Wireless Multimedia: Algorithms, Architectures and Applications, 2004.

VanetMobiSim <http://vanet.eurecom.fr/>

Watanabe M. et al., “The Video Streaming at Multihop Communication on WACNET (Wireless Ad Hoc Community Network) Test System for Its Inter Vehicle Communications,” Proc. Euro. Conf. Wireless Tech., pp. 179–82, 2005.

Wischhof, L.; Ebner, A.; Rohling, H. Informations disseminations in self-organizing intervehicle networks. IEEE transactions on intelligent transportation systems., v. 6, n. 1, p. 90-101. march 2005.

Wischhof, L.; Ebner, A.; Rohling, H. Informations disseminations in self-organizing intervehicle networks. IEEE transactions on intelligent transportation systems., v. 6, n. 1, p. 90-101. march 2005

Wisilpongphan, N.; Bai, F.; Mudalige, P.; Sadekar, V.; Tonguz, O. R. Routing in sparse vehicular ad hoc wireless networks. IEEE journal on selected areas in communications. V. 25, n. 8, p. 1538-1556, oct. 2007.

Wu H. et al., “MDDV: A Mobility-Centric Data Dissemination Algorithm for Vehicular Networks,” Proc. 1st ACM Wksp.Vehic. Ad Hoc Networks, pp. 47–56, 2004.

Xie, F.; Hua, K. A.; Wang, W.; Ho, Y. H. performance study of live video streaming over highway vehicular ad hoc networks. 2007.

Xie, X., Chen G., Vandenberg A. e Pan Y. Analysis of hybrid P2P overlay network topology, *Computer Communications* 31, p. 190–200, 2008

Yin, J.; Elbatt, T.; Yeung, G.; Ryu, B.; Habermas. S.; Krishnan, H.; Talty, T. Performance Evaluation of Safety Applications Over DSRC Vehicular Ad Hoc Networks, VANET'04 Philadelphia, Pennsylvania, USA 1 October 2004

Yousefi. Saleh, Mousavi. Mahmoud Siadat, Fathy. Mahmood Vehicular Ad Hoc Networks (VANETs): Challenges and Perspectives 6th International Conference on ITS Telecommunications Proceedings, 2006

Zhou S., Cai W., Lee B.-S., and Turner S.: Time-Space Consistency in Large-Scale Distributed Virtual Environments, *ACM Transactions on Modeling and Computer Simulation*, Vol. 14, No. 1, pp. 31-47,2004.