



Universidade Federal da Bahia
Escola Politécnica / Instituto de Matemática
Programa de Pós-Graduação em Mecatrônica

MARGARETE OLIVEIRA DOS SANTOS DE SÁ

**I-CAR: SERVIÇOS DE COMUNICAÇÃO
INTELIGENTES PARA APLICAÇÕES
MULTIAGENTES SOBRE REDES
VEICULARES**

DISSERTAÇÃO DE MESTRADO

Salvador
2012

MARGARETE OLIVEIRA DOS SANTOS DE SÁ

**I-CAR: SERVIÇOS DE COMUNICAÇÃO INTELIGENTES PARA
APLICAÇÕES MULTIAGENTES SOBRE REDES VEICULARES**

Dissertação apresentada ao Programa de Pós-Graduação em Mecatrônica da Escola Politécnica e do Instituto de Matemática, Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Mestre.

Orientador: *Prof. Dr. Sérgio Gorender*

Salvador
2012

Sistemas de Bibliotecas - UFBA

Sá, Margarete Oliveira dos Santos.

I-CAR: serviços de comunicação inteligentes para aplicações multiagentes sobre redes veiculares / Margarete Oliveira dos Santos de Sá. - 2012.

101 f. : il.

Orientador: Prof. Dr. Sergio Gorender.

Dissertação (mestrado) - Universidade Federal da Bahia, Instituto de Matemática e Escola Politécnica, Salvador, 2012.

1. Mecatrônica. 2. Sistemas de comunicação móvel. 3. Sistemas inteligentes de veículos rodoviários. 4. Sistemas operacionais distribuídos (Computadores). I. Gorender, Margarete. II. Universidade Federal da Bahia. Instituto de Matemática. III. Universidade Federal da Bahia. Escola Politécnica. IV. Título.

CDD - 621.38
CDU - 621.391

TERMO DE APROVAÇÃO

MARGARETE OLIVEIRA DOS SANTOS DE SÁ

I-CAR: SERVIÇOS DE COMUNICAÇÃO INTELIGENTES PARA APLICAÇÕES MULTIAGENTES SOBRE REDES VEICULARES

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Mecatrônica, Universidade Federal da Bahia - UFBA, pela seguinte banca examinadora:



Prof. Dr. Sérgio Gorender

Doutor em Ciência da Computação, Universidade Federal de Pernambuco, Brasil
Professor da Universidade Federal da Bahia



Prof. Dr. Raimundo José de Araújo Macêdo

Ph.D. em Ciência da Computação, University of Newcastle upon Tyne, Inglaterra
Professor da Universidade Federal da Bahia



Prof. Débora Abdalla Santos

Doutora em Ciência da Computação, Universidade Federal de Pernambuco, Brasil
Professor da Universidade Federal da Bahia

Salvador, 20 de junho de 2012.

TERMO DE APROVAÇÃO

MARGARETE OLIVEIRA DOS SANTOS DE SÁ

I-CAR: SERVIÇOS DE COMUNICAÇÃO INTELIGENTES PARA APLICAÇÕES MULTIAGENTES SOBRE REDES VEICULARES

Dissertação aprovada como requisito parcial para obtenção do grau de Mestre em Mecatrônica, Universidade Federal da Bahia - UFBA, pela seguinte banca examinadora:



Prof. Dr. Sérgio Gorender

Doutor em Ciência da Computação, Universidade Federal de Pernambuco, Brasil
Professor da Universidade Federal da Bahia



Prof. Dr. Raimundo José de Araújo Macêdo

Ph.D. em Ciência da Computação, University of Newcastle upon Tyne, Inglaterra
Professor da Universidade Federal da Bahia



Prof. Débora Abdalla Santos

Doutora em Ciência da Computação, Universidade Federal de Pernambuco, Brasil
Professor da Universidade Federal da Bahia

Salvador, 20 de junho de 2012.

RESUMO

Formação veicular é uma aplicação típica de sistemas multiagentes, na qual veículos automatizados (e.g. veículos autônomos, robôs móveis) devem entre si assumir e manter uma formação. Para a correta execução desta aplicação é necessário ter disponíveis mecanismos de comunicação confiáveis, garantindo a troca de mensagens entre os veículos. Estas mensagens são trocadas através de uma rede de comunicação móvel, como as redes *MANETs* e *VANETs*. Entretanto, tais redes, chamadas de redes *Ad Hoc*, não fornecem garantias à comunicação, podendo existir perda de mensagens ou da conectividade entre os nós. Isto ocorre devido à movimentação dos veículos, efeitos climáticos, entre outros que degradam a capacidade de comunicação entre eles. Assim, propomos uma abordagem com base em regras para o desenvolvimento de serviços de comunicação para as aplicações veiculares e criamos o *I-CAR* que é um conjunto de serviços inteligentes de comunicação para redes de comunicação veiculares, focado na aplicação da formação veicular. Estes serviços executam de forma distribuída, como módulos associados a cada agente veicular do sistema. Estes serviços são construídos como regras, possibilitando ao veículo autônomo tomar decisões inteligentes a cerca das melhores formas de transmitir mensagens através da rede, assim como, de detectar a falha de outros veículos a partir da comunicação. Construímos uma implementação destes serviços utilizando o simulador de redes *OMNet++*, em conjunto com o simulador de tráfego veicular *SUMO*, sendo realizadas diversas simulações para verificar o funcionamento do sistema. Os resultados obtidos demonstram que a sobrecarga e a perda de mensagens geradas pelo sistema não comprometem o funcionamento do mecanismo proposto.

Palavras-chave: Mecatrônica, Sistemas Multiagente, Comunicação Veicular, *VANET*, Sistemas Distribuídos

ABSTRACT

Vehicular Formation is a typical application of multiagent systems which automatized vehicles (e.g. autonomous vehicle, mobile robot etc) must compose and keep a formation. For a correct execution of this kind of application, reliable communication mechanisms are required, guaranteeing the message exchange between vehicles. These messages are exchanged using a mobile communication networks, such as *MANETs* and *VANETs*. However, such networks, named Ad Hoc Networks, do not provide communication guarantees – this is, messages exchanged or even the connectivity between nodes may be lost. It happens due to vehicle movement, climatic effects etc which degradate the communication capacity between nodes. In this work, we present a rule-based approach for the development of communication services for vehicular applications and created the *I-CAR*, a set of intelligent communication services for vehicle communication networks, focused on the implemenation of the vehicular formation. These services run in a distributed manner, as related modules in each vehicular agent of the system. Such services of the I-CAR are built using rules which allow the autonomous vehicle to make intelligent decisions about the best way to transmit messages through the network as well as to detect faults of other vehicles, using information of the communication. We built an implementation of these services using a network simulator (named *OMNet++*) combined with a vehicular traffic simulator (named *SUMO*). In this simulation environment, we performed a set of experiments and the obtained results demonstrate that the generated overload and lost messages do not compromise the functioning of the system.

Keywords: Mechatronics, Multiagent Systems, Vehicular Communication, VANET, Distributed Systems

CONTEÚDO

Capítulo 1—Introdução	1
Capítulo 2—Redes Veiculares	6
2.1 Comunicação Veicular	6
2.2 Formas de interação em redes veiculares	8
2.3 Redes VANET	9
2.4 Padrões e Protocolos em VANETs	10
2.4.1 Camadas Física e de Enlace	11
2.4.2 Camada de Rede	13
2.4.2.1 Endereçamento	14
2.4.2.2 Roteamento	14
2.4.3 Camada Transporte	17
2.5 Serviços de Comunicação Veicular	17
2.6 Considerações Finais	19
Capítulo 3—Aplicações sobre Redes Veiculares	21
3.1 Aplicações Veiculares	21
3.2 Requisitos de comunicação das Aplicações Veiculares	22
3.2.1 Requisitos para Aplicações de Controle do Movimento Individual	22
3.2.2 Requisitos para Aplicações de Controle Do Movimento Coletivo de Veículos	23
3.2.3 Requisitos para Aplicações de Serviços de Informações no Trânsito	24
3.3 Aplicações Baseadas em Sistemas Multiagentes	24
3.3.1 Formação Veicular	27
3.3.1.1 Estratégia de Controle de Formação	27
3.3.1.2 Estratégia de Controle de Formação Veicular Distribuída	28
3.3.1.3 Requisitos para a Comunicação na Formação Veicular Distribuída	29
3.4 Considerações Finais	30
Capítulo 4—A Proposta de Serviços Inteligentes de Comunicação para Aplicações Veiculares	31
4.1 Uma Abordagem para a Criação de Serviços Inteligentes de Comunicação	31
4.1.1 Percepção do Ambiente de Comunicação	34

4.1.2	Visão do Estado do Ambiente de Comunicação	34
4.1.3	Ações	39
4.1.4	Regras	40
4.1.5	Serviços de Comunicação	40
4.2	O I-CAR	41
4.2.1	Serviço para a Comunicação Fim-a-Fim - Unicast	43
4.2.2	Serviço de Monitoramento do Estado da Comunicação	49
4.2.3	Serviço Geral de Recebimento de Mensagens	60
4.2.4	Serviço para a Comunicação um Agente para Todos - Broadcast	64
4.2.5	Serviço de Detecção de Defeitos	64
4.3	Considerações Finas	66
Capítulo 5—Implementação, Simulação e Resultados		67
5.1	Implementação do I-CAR	67
5.1.1	Implementação dos Serviços do I-CAR	67
5.1.2	Implementação da Visão do Estado da Comunicação dos Veículos	68
5.1.3	Implementação dos Tipos de Mensagens	70
5.2	Simulação	71
5.2.1	Configuração da Mobilidade no SUMO	72
5.2.2	Configuração da Comunicação no OMNET	73
5.2.3	Parâmetros para Simulação do I-CAR	74
5.3	Análise do Funcionamento do I-CAR	76
5.3.1	Transição entre os Estados	76
5.3.2	Atraso das Mensagens Fim-a-Fim	77
5.3.3	Overhead de Mensagens de Monitoramento	78
5.3.4	Mensagens Perdidas	79
5.3.5	Mensagens de Monitoração Perdidas	80
5.3.6	Detecção de Defeitos	80
5.4	Considerações Finais	81
Capítulo 6—Conclusão		83
6.1	Trabalhos Futuros	85
6.2	Considerações Finais	86

LISTA DE FIGURAS

2.1	Esquemático de um veículo com equipamentos internos para o suporte à comunicação veicular [19]	7
2.2	Taxonomia dos Sistemas de Comunicação Veicular [33]	8
2.3	Canal DSRC [18]	12
4.1	Mecanismo de Comunicação e o Agente Veicular	32
4.2	Estrutura do Mecanismo de Comunicação	33
4.3	Exemplo de uso de atributos do ambiente para definir a situação da comunicação entre dois agentes	36
4.4	Exemplo de uma ação para o envio de mensagem entre dois agentes	39
4.5	Exemplo de uma regra	40
4.6	Exemplo do encadeamento e combinações de regras para definir um serviço	41
4.7	Interação entre Veículos em uma Formação Veicular - modificada de [42]	42
4.8	Transmissão de mensagem do agente α para o ω	45
4.9	Regras para disparar ações para envio de mensagens	47
4.10	Dados Analisados	49
4.11	Regras para enviar solicitação de monitoramento	54
4.12	Regras para enviar resposta de solicitação de monitoramento	56
4.13	Regras para encaminhar uma solicitação de monitoramento	57
4.14	Regras para encaminhar resposta de solicitação de monitoramento	59
4.15	Regras de atribuição de estado dos agentes na recepção de mensagens	62
4.16	Regras de atribuição de estado do agente local	63
4.17	Regras para enviar mensagens de aplicações do agente para o grupo	64
4.18	Regra para deteção de Falhas do agente	66
5.1	Diagrama de classes dos Serviços do I-CAR	68
5.2	Diagrama que representa a estrutura de dados do estado da comunicação	68
5.3	Trajetória Curvas Acentuadas	72
5.4	Amostra de Parte da Trajetória Implementada no <i>SUMO</i>	72
5.5	Módulo <i>VehicularFormationGComm</i> embarcado no módulo Car do <i>VEINS</i>	73
5.6	Transição entre os estados de um agente remoto	76
5.7	<i>Overhead</i> de mensagens de monitoramento	78
5.8	Mensagens Perdidas	79
5.9	Mensagens de Monitoração Perdidas	80
5.10	Percentual de falsas suspeitas de falhas	81

LISTA DE TABELAS

2.1	Comparativo entre VANETs vs MANETs comuns [33]	10
4.1	Tabela de Transição dos Estados da Comunicação Entre Dois Agentes . .	37
5.1	Atraso Médio Fim-a-Fim	78

INTRODUÇÃO

A formação veicular distribuída é uma aplicação de coordenação e assistência de tráfego que permite que veículos autônomos inteligentes (e.g. robôs móveis) interajam entre si e se movimentem para atender às missões individuais e/ou coletivas (e.g. um grupo de veículos mantendo uma trajetória pré-definida) [7, 10]. Este tipo de aplicação é considerado um problema de consenso que pode ser resolvido por sistemas multiagentes [46] – ver por exemplo [12, 29].

Em uma formação veicular distribuída, o grupo de agentes veiculares autônomos realiza o controle individual e coletivo por meio de sistemas de controle do movimento. Estes agentes precisam não só garantir a formação, mas também fazê-la de forma eficaz, desviando de obstáculos e evitando colisões entre veículos sem realizar movimentos abruptos e/ou equivocados durante a trajetória [7, 10].

Para tanto, os agentes veiculares autônomos interagem entre si trocando informações de controle do movimento (e.g. velocidade, posição, aceleração, entre outras) e de emergência para evitar colisões e/ou falhas na formação (e.g. mensagens alertando sobre obstáculos nas vias).

De modo geral, as informações de controle do movimento são transmitidas, periodicamente, para cada veículo e/ou ao grupo de veículos envolvidos na formação. Estas transmissões possuem restrições temporais quanto ao atraso e perda de mensagens [42, 7, 10].

As informações de emergência para evitar colisões e/ou falhas na formação, também possuem restrições temporais e são transmitidas quando necessário para um grupo de veículos alvo ou para todos os veículos envolvidos na formação [42, 7, 10].

Entretanto, a formação veicular distribuída, assim como a maioria das aplicações vei-

culares, é executada sobre um ambiente de comunicação dinâmico onde as conexões entre os veículos não são garantidas e podem mudar ao longo da trajetória. Estas mudanças podem fazer com que as mensagens transmitidas pela formação veicular sejam perdidas ou atrasadas fazendo com que os veículos não consigam manter a formação de maneira eficaz.

Esse ambiente de comunicação caracteriza as *VANETs* (*Vehicular Ad Hoc Network*) que são redes veiculares *ad hoc*, sem fio, vulneráveis aos efeitos da mobilidade, do clima e de outros tipos de interferências nos sinais de comunicação. Além disso, os nós podem ser heterogêneos e várias aplicações podem estar sendo executadas concomitantemente aumentando a complexidade para manter a eficácia na comunicação [33, 42].

Para que as mensagens consigam chegar ao seu destino de forma correta e em tempo hábil, são necessários serviços de comunicação distribuídos como *unicast*, *broadcast*, *multicast*, detectores de defeitos, entre outros que se adaptem à dinâmica do ambiente, de modo a, reduzir o impacto das mudanças topológicas, provendo assim, melhores garantias na entrega destas mensagens [7, 42, 21].

Prover serviços de comunicação distribuído que se adaptem ao ambiente de comunicação das redes veiculares é um grande desafio que motiva os pesquisadores a investigar formas de adaptação para tais serviços [7, 42, 33]. Uma abordagem muito utilizada é a criação de heurísticas baseadas nas informações do ambiente (e.g. distância, velocidade, condição de vizinhança, capacidade do sinal de comunicação, entre outros) para definir a forma de comunicação de um veículo. Entre os vários trabalhos que utilizam esta abordagem citamos: os sistemas de encaminhamento de mensagens desenvolvido em [1] e [2] que utilizaram informações sobre a capacidade de comunicação para encaminhar mensagens; em [32] os autores usaram informações de mobilidade para adaptar a taxa de transmissão em relação à velocidade do veículo; o *Middleware RT-STEAM* apresentado em [28] transmite informações a uma área geográfica relevante para a aplicação.

Além disso, vários trabalhos estão sendo realizados em relação aos protocolos de rede já existentes, buscando aproveitar as tecnologias, adaptando estes protocolos à dinâmica

do ambiente de comunicação veicular, como por exemplo [33]: o uso dos protocolos *IEEE 802.11* e de tecnologia celular *GSM/3G*; o uso de técnicas para os protocolos de roteamento como as baseadas em topologia, em posição, região geográfica, *clusters*, *broadcast*, etc; a busca pela adaptação dos protocolos de comunicação fim-a-fim como o *UDP* e o *TCP*.

Desse modo, neste trabalho, propomos o uso de regras para o desenvolvimento de serviços de comunicação para aplicações veiculares. Esta abordagem propõe a construção de serviços de comunicação que se adaptam à dinâmica e a heterogeneidade do ambiente das redes veiculares. Utilizando esta abordagem, criamos o *I-CAR (Intelligent Communication Mechanism for Multi-Agent Vehicular Applications)* que é um conjunto de serviços inteligentes de comunicação multiagente baseado em regras sobre redes veiculares. Tal conjunto contém o serviço de monitoramento do estado do ambiente de comunicação, o de detecção de defeitos e os serviços de comunicação *unicast* e *broadcast*.

O uso desta abordagem possibilita a construção de serviços de comunicação que se adaptem ao cenário o qual a aplicação está sendo executada. Permite que as regras e ações possam ser diferenciadas entre os agentes mesmo provendo o mesmo serviço de comunicação. Além disso, é possível combinar as regras dos variados serviços de modo a aproveitar as mensagens que trafegam na rede (oriundas das aplicações e dos próprios serviços em execução) buscando melhorar o desempenho do sistema.

Para isso, desenvolvemos uma estrutura que define os elementos necessários para a construção de um serviço que se adapte ao ambiente de comunicação. Com esta estrutura, o projetista define o conjunto de informações do ambiente que são relevantes para o serviço a ser provido, considerando aspectos como: 1) os requisitos das aplicações a serem atendidas; 2) a dinâmica da topologia da rede; 3) as características de cada veículo (*e.g.* tipo e quantidade de dispositivos de comunicação e protocolos de rede). Os elementos que compõem essa estrutura são: a visão do estado do ambiente de comunicação, as ações, as regras e os serviços.

A visão do estado do ambiente de comunicação consiste em um conjunto de propri-

idades, tais como: estado entre o agente local (a) e um agente remoto (b); o estado da vizinhança entre dois agentes; o estado da comunicação de um agente local em relação ao grupo. Estas propriedades são mantidas por informações do ambiente que devem ser coletadas durante a execução do sistema, como por exemplo, dados de recebimento, transmissão e atraso de mensagens na rede.

As ações devem estar relacionadas às características do agente juntamente ao tipo de serviço a ser provido. As regras devem estar associadas à visão do estado do ambiente de comunicação e às ações a serem executadas. Já o serviço é um conjunto de regras encadeadas e combinadas entre si para definir o comportamento do serviço proposto.

Com base na abordagem proposta, foi criado o *I-CAR* que é encapsulado em cada agente veicular autônomo de um grupo de veículos em formação. Deste modo, o *I-CAR* provê serviços de comunicação que permitem que cada veículo decida sobre a sua forma de se comunicar com um ou mais agentes, tomando como referência a sua própria visão do estado do ambiente de comunicação.

O serviço de monitoramento verifica a situação de cada propriedade da visão do estado do ambiente de comunicação, caso necessário, envia uma mensagem solicitando informações sobre a situação de um agente remoto e atualiza a visão do estado do ambiente de comunicação.

O *unicast* e o *broadcast* enviam, respectivamente, uma mensagem para um e para todos os veículos envolvidos na comunicação. Para isso, cada agente assume estratégias de transmissão de mensagens de acordo com a sua própria visão do estado do ambiente de comunicação.

O detector de defeitos do *I-CAR* foi desenvolvido para identificar em um grupo de veículos em formação se existe algum deles com falha por *crash*, este utiliza as mensagens trocadas pela aplicação e da monitoração para verificar se existe falha no ambiente.

O *I-CAR* foi desenvolvido utilizando a *UML* (*Unified Modeling Language*) sobre o paradigma de programação orientada a objetos e implementado na linguagem de programação *C++*. Após a implementação, o código foi encapsulado no *Framework Veins*,

o qual, utiliza o simulador de redes *OMNet++*, em conjunto com o simulador de tráfego veicular *SUMO*. A simulação foi realizada de forma experimental, os resultados encontrados ainda são iniciais, mas podemos observar, entre outros, que as perdas de mensagem diminuíram em relação ao um ambiente sem o mecanismo de comunicação.

Estes serviços de apoio a comunicação são empacotados como um mecanismo de comunicação (o *I-CAR*). Assim, ao longo do texto os termos “conjunto de serviços de comunicação” e “mecanismo de comunicação” serão usados como sinônimos, salvo os casos em que são explicitamente declarados no texto. Os próximos capítulos deste trabalho estão organizados da forma a seguir. O Capítulo 2 apresenta aspectos básicos da comunicação veicular, as formas de interação entre os elementos envolvidos em uma comunicação, a definição das *VANETs*, os padrões e protocolos para redes veiculares com seus desafios para o desenvolvimento e as abordagens utilizadas nos serviços de comunicação existentes. O Capítulo 3 discute sobre os tipos de aplicações veiculares e os respectivos requisitos de comunicação, dentre os quais detalhamos os a aplicação de formação veicular suas estratégias e, também, seus respectivos requisitos. O Capítulo 4 apresenta a abordagem baseada em regras para a construção de serviços de comunicação e o desenvolvimento do mecanismo de comunicação *I-CAR*. No Capítulo 5 apresentamos a implementação, a simulação e os resultados encontrados e no Capítulo 6 estão as conclusões, trabalhos futuros e considerações finais sobre o trabalho.

CAPÍTULO 2

REDES VEICULARES

Este Capítulo aborda conceitos básicos associados à comunicação em redes veiculares, para tanto: discute-se questões relacionadas aos dispositivos embarcados que suportam a comunicação intra-veicular e extra-veicular – Seção 2.1; comenta as formas de interação entre veículos ou entre estes e as vias – Seção 2.2; enumera os principais requisitos característicos de Redes Veiculares *Ad Hoc* (*VANET*, *Vehicular Ad Hoc NETWORKS*) – Seção 2.3; aborda protocolos e padrões de comunicação utilizados pelas aplicações de redes veiculares – Seção 2.4; e traz algumas considerações finais que assinalam os principais pontos discutidos em todo o capítulo – Seção 2.6.

2.1 COMUNICAÇÃO VEICULAR

A comunicação veicular (*VC - Vehicular Communications*) busca prover facilidades da comunicação em redes veiculares. Para isto, são pesquisadas e desenvolvidas arquiteturas de comunicação embarcadas nos veículos que são responsáveis tanto pela interação intra-veicular (entre os dispositivos internos aos veículos) quanto pela interação extra-veicular (entre os dispositivos de redes nos veículos e nas vias), ver [16].

As arquiteturas de comunicação intra-veicular não apenas coordenam a interação entre os dispositivos embarcados como sensores, atuadores, controladores lógicos¹ ou microcontroladores situados em cada veículo, mas também coletam e mantêm informações locais em cada veículo. De um modo geral, estas arquiteturas intra-veiculares fazem uso de redes de comunicação de tempo real que permitem algum determinismo com relação às prioridades ou atrasos nas entregas das mensagens – a exemplo da rede *CAN*², a qual é

¹e.g. *PLC, Programmable Logic Controller* – ver [30].

²*Controller Area Network* – ver [38].

usada em soluções como a proposta por [15].

As arquiteturas de comunicação extra-veicular permitem a cooperação entre os veículos e a troca de informações entre estes com outros elementos das vias (e.g. equipamentos de controle de tráfego, mecanismos de sinalização de trânsito, mecanismos de monitoramento etc.[16, 15].

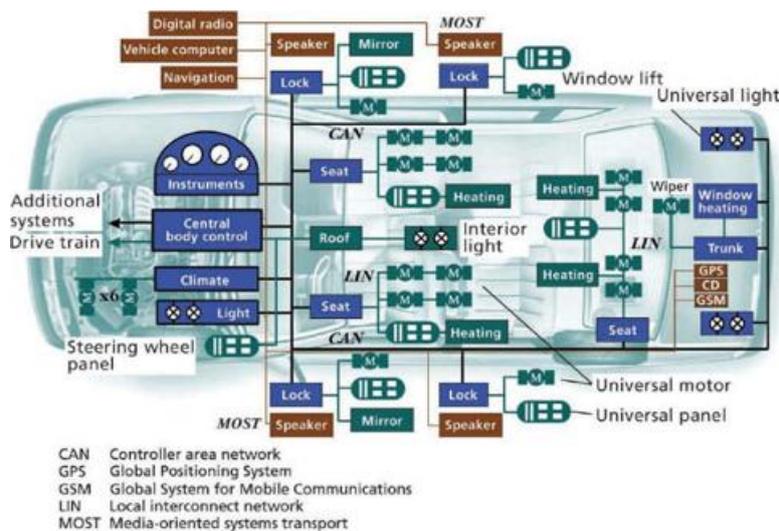


Figura 2.1. Esquemático de um veículo com equipamentos internos para o suporte à comunicação veicular [19]

A Figura 2.1 apresenta um veículo equipado com dispositivos (*OBU*, *On-Board Unit*) que permitem as comunicações intra-veicular e extra-veicular [33]. Nesta figura, são ressaltados os equipamentos em um veículo típico de sistemas veiculares: (i) CPU (unidade central de processamento), responsável por executar os softwares da aplicação e os protocolos de comunicação; (ii) transceptor de rede sem fio, que transmite e recebe dados de dispositivos nas estradas e dos veículos do sistema; (iii) GPS (*Global Positioning System*), responsável por receber informações de posicionamento e sincronização do relógio; (iv) sensores usados na coleta e distribuição interna de vários parâmetros que podem ser usados durante a cooperação entre os veículos; interface de E/S, responsáveis por viabilizar a interação humana com o sistema; entre outros [15, 19].

A interação extra-veicular é um aspecto relevante para as discussões realizadas nos próximos capítulos. Por conta disto, a seção a seguir discute mais detalhadamente diver-

os aspectos associados a esta modalidade de interação.

2.2 FORMAS DE INTERAÇÃO EM REDES VEICULARES

A comunicação extra-veicular pode ocorrer a partir das interações veículo-para-veículo, veículo-para-via ou via-para-via. Neste sentido, [33] classificou os sistemas de comunicação veiculares em três grupos básicos: I. Sistemas de comunicação entre veículos (*IVC*, *Inter-Vehicle Communications*); II. Sistemas de comunicação entre veículos e vias (*RVC*, *Roadside-to-Vehicle Communication Systems*); e III. Sistemas de comunicação híbridos (*HVC*, *Hybrid Vehicular Communication*). A Figura 2.2 apresenta um diagrama da taxonomia proposta por [33] – a qual é discutida com maiores detalhes a seguir.

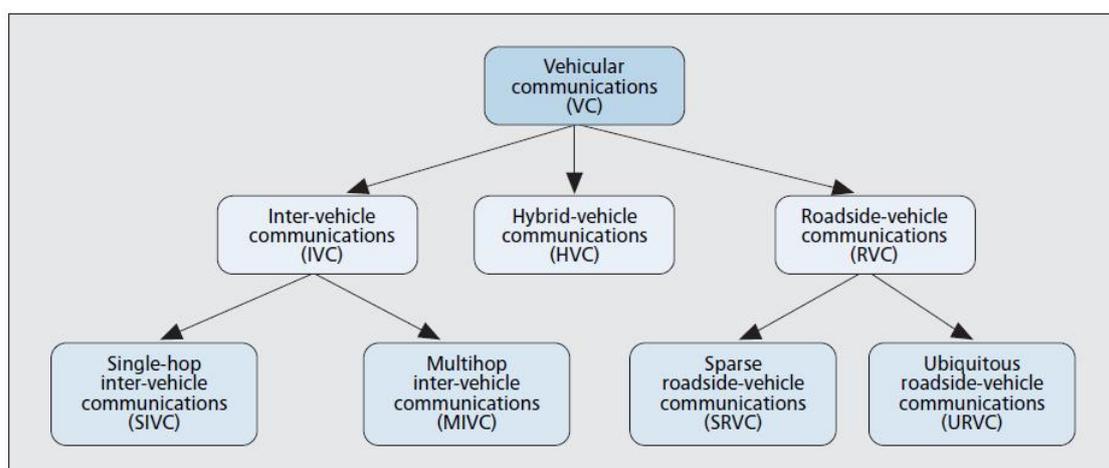


Figura 2.2. Taxonomia dos Sistemas de Comunicação Veicular [33]

Os sistemas de comunicação veicular do tipo *IVC* descrevem os mecanismos de comunicação de veículo-para-veículo. Neste tipo de sistema, a comunicação pode ser: de curto alcance, i.e. as mensagens são transmitidas apenas para os veículos mais próximos em um único salto – sendo por conta disto denominado de sistema de comunicação *SIVC* (*Single-hop IVC*); ou de longo alcance, i.e. as mensagens são transmitidas do veículo de origem até o veículo de destino, podendo ser encaminhadas por veículos intermediários em múltiplos saltos – sendo assim denominado por *MIVC* (*Multihop IVC*). Os sistemas de comunicação *MIVC*, por conta de suas características, demandam por algoritmos e

protocolos adequados para roteamento e transporte das mensagens.

Os sistemas do tipo *RVC* descrevem os mecanismos de comunicação de rodovia-para-veículo. Nestes sistemas, as rodovias possuem toda a infra-estrutura de comunicação necessária para a aplicação e podem ser do tipo esparsos (*SRVC - Sparse roadside-vehicle communications*) e ubíquos (*URVC - Ubiquitous Roadside-Vehicle Communications*). Tais sistemas são capazes de fornecer serviços de comunicação em pontos importantes das vias. A programação de semáforos em um cruzamento movimentado e a disponibilidade de estacionamento em um aeroporto, são exemplos de aplicações que requerem um sistema de comunicação do tipo *RVC*.

Já os sistemas *HVC* são propostos como uma extensão de sistemas de *RVC*, contudo a infra-estrutura de comunicação necessária para a aplicação é provida, também, pelos veículos disponíveis nas vias, como por exemplo, quando os dispositivos das vias não estão no alcance direto de um veículo, são usados outros veículos como roteadores móveis.

Todas as aplicações do tipo *RVC* podem ser implementadas como sistemas *HVC*, uma vantagem para isso é que a quantidade de dispositivos nas vias diminui. Entretanto, podem ter um intervalo de tempo maior de transmissão devido aos encaminhamentos das mensagens na rede. Além disso, a conectividade da rede não pode ser garantida em cenários com baixa densidade de veículos, pois, podem não existir veículos para encaminhar as mensagens [33].

2.3 REDES VANET

Dadas as características dos sistemas de comunicação *IVC*, os mesmos podem ser suportadas por *Redes Móveis Ad hoc (MANET, Mobile Ad hoc NETWORKS)*, as quais não possuem uma infra-estrutura fixa e podem acomodar uma constante evolução da topologia da rede. Contudo, algumas das características dos ambientes associados aos sistemas de comunicação *IVC* diferem das redes *MANETs* comuns – criando assim uma subclasse de redes *MANETs* para a comunicação veicular, denominada de *Redes Veiculares Ad hoc*

(*VANET*, *Vehicular Ad hoc NETWORKS*). A Tabela 2.1 apresenta um quadro comparativo que aponta as principais diferenças entre as redes *VANETs* e as redes *MANETs* comuns.

Tabela 2.1. Comparativo entre *VANETs* vs *MANETs* comuns [33]

Requisitos	<i>VANETs</i>	<i>MANETs</i> comuns
Aplicações	Aplicações diferentes (requisitos de rede específicos de cada aplicação)	Aplicações similares (requisitos de rede das aplicações são similares)
Endereçamento	Freqüentemente, os endereços são baseados em requisitos geográficos, demandando modelos de roteamento diferenciados	As aplicações exigem endereçamento fixo
Taxa de variação de link	Devido a alta mobilidade, caminhos com múltiplos saltos para a comunicação fim-a-fim precisam ser reparadas com alta freqüência	Mobilidade moderada
Modelo de mobilidade	Os modelos gerais usados em <i>MANETs</i> não atendem às características de mobilidade	<i>RWP</i> (<i>Random Waypoint</i>)
Modelo de Energia	Considera que os nós possuem fonte de energia ilimitada	Precisa considerar restrições no consumo de energia em cada nó

De um modo geral, as redes *VANETs* possuem os seguintes requisitos de comunicação:

- **nós com alta mobilidade** – como, por exemplo, aviões, trens, automóveis, robôs, entre outros;
- **tipos variados de aplicações** – os quais interferem nas arquiteturas e infraestruturas do sistema de comunicação;
- **modelo de roteamento diferenciado** – em muitos casos, requer um novo paradigma de roteamento (baseado em requisitos geográficos), os quais são significativamente diferente dos disponíveis em redes *MANETs*;
- **mudanças frequentes na topologia da rede** – deve manter a conectividade em uma comunicação fim-a-fim, mesmo com o alto grau de mobilidade dos nós;
- **ausência de restrições energéticas** – a eficiência da energia dos nós não é um fator relevante, pois o sistema considera que a carga da energia é renovada.

2.4 PADRÕES E PROTOCOLOS EM VANETS

As aplicações em *VANETs* demandam por diferentes requisitos de comunicação veicular. Estes requisitos motivam o desenvolvimento de diferentes padrões e protocolos

de comunicação. Esta seção discute alguns destes padrões e protocolos. Para facilitar o entendimento dos mesmos e de seus desafios associados, os padrões e protocolos são apresentados seguindo o princípio das camadas de acordo com o modelo OSI/ISO³. Contudo, não são abordadas todas as camadas do modelo OSI, sendo dada maior atenção aos aspectos relacionados às camadas física e de enlace de dados; à camada de redes; e à camada de transporte – as quais são mais comumente utilizadas na maioria das abordagens propostas na literatura.

2.4.1 Camadas Física e de Enlace

Nesta camada são verificados os meios físicos e os protocolos de acesso ao meio. Existem vários padrões de comunicação que estão sendo utilizados em sistemas de comunicação veicular, tais como: óptico, laser, infra-vermelho; radar; banda ultralarga (UWB *Ultra-Wide Band*); DSRC (*Dedicated short-range communication*); IEEE 802.11; bluetooth; e redes móveis celular (GSM /GPRS - *Global System for Mobile Communications* e 3G). Cada uma destas tecnologias possui suas vantagens e desvantagens associadas às camadas física e de enlace [33], [45].

Os padrões baseados em meios óptico, laser e infra-vermelho realizam comunicação em linha reta (do transmissor ao receptor), possuem alcance muito curto, são apropriados para sistemas de comunicação veicular de um único salto (*SIVC*). A comunicação baseada em radares possui uma frequência de micro-ondas de 60GHz, trabalha melhor em linha reta, possui curto alcance e, atualmente, é muito usado nas estradas para a detecção de excesso de velocidade. Já a banda ultralarga possui uma alta taxa de transmissão de dados em curtas distâncias e bom desempenho em alcançar as longas distâncias. Devido as limitações imposta pela *FCC* (*Federal Communications Commission*) nos EUA (Estados Unidos da América), o alcance da transmissão direta para comunicação veicular só é permitida para poucos metros.

O *DSRC* é um padrão de transmissão também utilizado para a comunicação veicular

³*Open System Interconnect/International Organization for Standardization.*

(veículo-para-veículo e infra-estruturas-para-veículos), possui um espectro de 5.9 GHz e foi reconhecida pela FCC. Este possui uma faixa de espectro de 75MHz dividido em 7 partes, como mostra a Figura 2.3. Nesta divisão contempla uma faixa para a transmissão e recepção de dados para aplicações críticas [18].

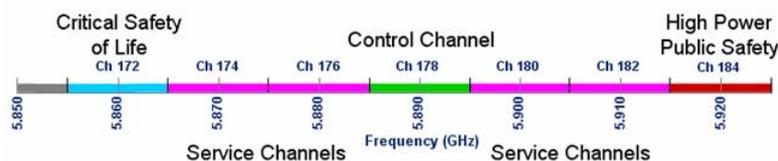


Figura 2.3. Canal DSRC [18]

Atualmente, o IEEE está desenvolvendo o protocolo 802.11p, cujo padrão é denominado IEEE 1609 Padrões *WAVE* (*Wireless Access in Vehicular Environments*) e está baseado no DSRC. Este protocolo está sendo preparado para atender a ambientes com alta velocidade, à taxa de transmissão acima de 27 Mb/s e a um curto alcance mínimo de 1000m. Muitas pesquisas estão buscando, também, baixar a latência para comunicações sem fio e desenvolver e adequar os mecanismos para a comunicação em redes *ad hoc* [18].

Também estão sendo testados o uso do protocolo padrão IEEE 802.15.1, conhecido como bluetooth, principalmente, para a comunicação em curto alcance e aplicações não críticas que transmitem dados em formato *MP3*⁴ de um veículo para outro, por exemplo. A área de atuação deste protocolo é *WPAN* (*Wireless Personal Area Network*) e em sua rede conhecida como piconet⁵ o número de nós é limitado, além disso, o tempo de estabelecimento de conexão entre um nó mestre e um escravo é, significativamente, longo para as características dos sistemas de comunicação veicular. Assim, manter uma estrutura de piconet em sistemas dinâmicos como são os sistemas de comunicação inter-veiculares é um grande desafio [33].

As pesquisas que envolvem tecnologias de redes móveis celular estão sendo desenvolvidas em projetos na Europa, principalmente, na Itália e Alemanha, os principais protocolos

⁴MPEG Layer 3 é um padrão de compactação de áudio que permite que as músicas fiquem com 1/10 do tamanho original sem uma degradação muito grande da qualidade [37].

⁵Piconet consiste em uma rede com um grupo de nós sendo um mestre e os outros escravos em uma rede sem fio [37].

estudados são: *GSM / GPRS - Global System for Mobile Communications / Universal Mobile Telecommunications System e Tecnologia* e *Tecnologia 3G (Third Generation)* [33].

Os pesquisadores Europeus se motivaram a pesquisar a comunicação celular 3G para a comunicação veicular devido as características atuais desta tecnologia, tais como: as várias áreas geográficas contempladas pelo serviço de comunicação; uma boa solução para fora das grandes cidades e auto-estradas; a infra-estrutura já existente permitindo adaptar e reutilizar a tecnologia diminuindo tempo e custo de desenvolvimento de novas tecnologias; o número de usuários cada vez mais crescente. Além disso, a tecnologia 3G provê limites de longo alcance, oferece garantias de qualidade de serviço que são essenciais às aplicações críticas e estão projetadas para a mobilidade de alta velocidade.

Contudo, as principais desvantagens da telefonia celular para comunicação veicular estão em: não estarem preparados para um grande número de usuários simultâneos por longos períodos de tempo e grande volume de tráfego; a infra-estrutura centralizada para coordenar a transmissão de nós móveis é insuficiente e pode causar congestionamentos, principalmente, em aplicações críticas; a liberdade de atribuir diferentes espectros para a mobilidade de alta velocidade pode causar colisões e perdas de dados.

2.4.2 Camada de Rede

Na camada de rede, os principais desafios de implementação dizem respeito à manutenção de endereços e rotas em ambiente com alta mobilidade e mudanças topológicas da rede. Assim, nesta seção são discutidas as formas de endereçamento adotadas e as principais técnicas e desafios dos mecanismos de roteamento de mensagens em redes veiculares.

2.4.2.1 Endereçamento

Em alguns sistemas de comunicação veicular, como *MIVC* e *HVC* (ver Seção 2.2), por exemplo, as mensagens saem de um nó transmissor e precisam chegar aos nó(s) destinatário(s). O problema do endereçamento surge, pois, o nó transmissor e/ou intermediário precisa conhecer quem é o nó destino no momento em que uma aplicação requer a transmissão e/ou encaminhamento de mensagens.

No endereçamento fixo, os endereços são estáticos e cada nó da rede possui um endereço único e este não muda com sua localização. No endereçamento geográfico, os endereços são dinâmicos e cada nó possui um endereço associado a sua posição geográfica correspondente – latitude e longitude, por exemplo, [13].

Os endereços fixos são utilizados por aplicações em que existe um grupo de nós alvo (*ZOR*, *Zone of Relevant*). Nestas aplicações, estes grupos de nós são conhecidos e não mudam durante toda a execução – a formação veicular é um exemplo deste tipo de aplicação [7].

Os endereços geográficos são utilizados por aplicações em que os nós destinatários não são conhecidos a priori. Estas aplicações, definem a região geográfica para a qual a mensagem deve ser encaminhada, mas não se sabe ao certo quais nós estão presentes em tal localidade – alertas de congestionamento e notificações de acidentes são exemplos de aplicações que comumente usam endereçamento geográfico [42].

Muitos trabalhos tentam lidar com a mobilidade dos nós das redes veiculares, a partir do suporte a informações de localização geográficas em protocolos de prateleira como o *IPv6* (*Internet Protocol Version 6*), por exemplo – ver [13], [14] e [33].

2.4.2.2 Roteamento

Vários estudos têm buscado novos paradigmas para os protocolos de roteamento para a comunicação veicular. Além disto, os pesquisadores têm procurado aplicar e/ou adaptar

as técnicas de já existentes para a construção dos protocolos para as *VANETs*, as quais apresentaremos a seguir [33, 5, 17, 3].

Os protocolos baseados em topologia

Os protocolos baseados em topologia (*Topology based protocols*) são aqueles que precisam descobrir e manter as rotas em uma tabela antes de enviar os dados. Estes podem ser reativos, proativos e híbridos [17] e são muito aplicados em soluções de roteamento para *MANETs*. Manter atualizadas as rotas da rede nestes protocolos é um grande desafio, pois, as características do ambiente, principalmente, da mobilidade podem provocar mudanças abruptas na topologia da rede [33].

Os protocolos de roteamento *unicast* com endereço fixo permitem utilizar as técnicas baseadas na topologia da rede, como por exemplo, o protocolo reativo *AODV* (*Ad hoc On-Demand Distance Vector*) que é muito aplicado em redes móveis *ad hoc* e o seu uso em *VANETs* tem sido analisado [33, 5]. Além deste, existem outros protocolos reativos que estão sendo estudados para o uso em *VANETs* como: *DSR*, *Temporally-Ordered Routing Algorithm (TORA)*, etc [33, 17].

Protocolos baseados em Grupo de Nós

Protocolos baseados em grupo de nós (*Cluster based protocols*) são outras soluções para o roteamento inter-veicular que estão sendo pesquisadas. O encaminhamento dos pacotes é feito pelos nós líderes de cada grupo de veículos. Tais grupos, são formados e mantidos dinamicamente conforme a proximidade entre os veículos. São vários os protocolos de roteamento baseados em *clusters*, como exemplo podemos citar o *HCB novel based Hierarchical Cluster* [17].

Utilizar um único nó líder para encaminhar mensagens para um grupo é um desafio, pois, devido a alta mobilidade, atrasos e sobrecarga dos nós líderes podem ocorrer durante

a formação e manutenção dos grupos, como citado em [33].

Protocolos baseados em posição dos nós

Os protocolos baseados em posição (*position based protocols*) realizam o encaminhamento de mensagens usando informações sobre a posição geográfica para selecionar o próximo nó. Não é necessário manter rotas entre a origem e o destino.

O *GPSR (Greedy Perimeter Stateless Routing)* é um protocolo de roteamento de rede sem fio que usa a posição dos nós roteadores para tomar decisões sobre o encaminhamento [33]. Além deste, existem outros protocolos como: *Vertex-Based predictive Greedy Routing (VGPR)*; *Greedy Traffic Aware Routing Protocol (GYTAR)* para ambientes urbanos, entre outros.

Protocolos baseados em região geográfica alvo

Já os protocolos de roteamento baseados em região geográfica alvo (*Geocast based protocols*) possuem uma região geográfica alvo identificada pela camada de rede, a qual define se entrega o pacote à aplicação ou não, como por exemplo [13]. Assim, mesmo entre nós vizinhos, só os nós aptos podem receber os pacotes.

Protocolos baseados em difusão

Diversos protocolos utilizam a técnica de inundação por difusão (*broadcast flooding*), para encaminhar uma mensagem para todos os veículos. Nesta técnica cada veículo retransmite uma cópia de cada mensagem recebida. Existem algumas variações desta técnica, procurando restringir a retransmissão de mensagens, uma vez que o *flooding* gera um alto uso de largura de banda [25].

2.4.3 Camada Transporte

A camada de transporte provê mecanismos de comunicação fim-a-fim, tais como, confiabilidade, controle de fluxo e de congestionamento, entre outros. Contudo, em uma rede veicular prover tais mecanismo é um grande desafio para as aplicações veiculares, devido, principalmente, aos efeitos da mobilidade dos veículos em relação a topologia da rede.

Os pesquisadores tem tentado aproveitar os protocolos atuais da camada de transporte como *TCP* (*Transmission Control Protocol*) e o *UDP* (*User Datagram Protocol*), adaptando-os para atender aos requisitos das aplicações veiculares. Contudo, os trabalhos ainda são iniciais apenas buscando identificar qual o protocolo de melhor desempenho quando utilizado em um tipo de aplicação veicular, como o trabalho realizado em [34].

2.5 SERVIÇOS DE COMUNICAÇÃO VEICULAR

As aplicações veiculares, como a formação veicular, demandam por serviços de comunicação (e.g. unicast, multicast, detector de defeitos) que consigam, sobre um ambiente dinâmico de comunicação, atender aos seus requisitos temporais de comunicação e de entrega de mensagens. Contudo prover serviços de comunicação em um ambiente dinâmico como as redes *VANETs* é um grande desafio, já que a topologia da rede pode mudar ao longo da trajetória (como discutido nas Seções 2.3 e 2.4). Vários trabalhos encontrados na literatura buscam prover serviços de comunicação sobre as redes veiculares [21, 42, 20]. Muitos destes serviços propõem adaptar o seu comportamento à dinâmica do ambiente, utilizando heurísticas baseadas nas informações do próprio ambiente. Esta adaptação muitas vezes é feita a partir de um conjunto de informações (atributos) do próprio ambiente. Com esta abordagem o comportamento dos serviços varia de acordo com as mudanças dos dados coletados sobre os atributos.

Nesta seção apresentamos, dentre vários, alguns trabalhos encontrados na literatura sobre os serviços de comunicação para aplicações veiculares que propõem adaptar o serviço

de comunicação proposto à dinâmica do ambiente, utilizando heurísticas baseadas nas informações do próprio ambiente.

O serviço de encaminhamento de mensagens desenvolvido por [1], provê uma técnica para organizar os envios de mensagens *Hello* para a atualização da topologia da rede. Tal técnica consiste em enviar uma mensagem para um grupo de veículos quando a direção e a velocidade de um veículo muda, caso contrário, ele irá parar de transmitir a mensagem *Hello* e esperar por outra mudança. Neste caso, as atualizações da topologia de rede realizadas ao enviar uma mensagem *hello* ficam muito associadas ao movimento do carro, contudo, como discutimos anteriormente, não só a mobilidade interfere na mudança da topologia da rede.

Já em [32], autores usaram informações da mobilidade para adaptar a taxa de transmissão em relação à velocidade do veículo para diminuir as perdas de conexão ocasionadas pela mudança de velocidade.

O trabalho apresentado em [2], propõe um protocolo de *broadcast* denominado *FROV* (*Father Relay and Oracle for VANET*), que procura realizar escolhas sobre o modo da transmissão de mensagens a partir de informações fornecidas por um oráculo. Este oráculo mantém informações tais como: uma lista de veículos vizinhos; uma lista de veículos conhecidos de transmissores de mensagens do oráculo; etc.

Em [27] propõe a implementação de um protocolo de disseminação de mensagem na vizinhança (*Reliable Neighborcast Protocol - RNP*). Tal protocolo foi implementado para aplicações que precisam de comunicação inter-veicular e que trocam informações como velocidade, posição e estado dos veículos próximos. Este identifica a presença de veículos vizinhos e propaga mensagens entre eles.

O *Middleware RT-STEAM* apresentado em [28] é um tipo de aplicação que permite vincular os eventos relevantes da aplicação a uma área geográfica, chamada de proximidade, em que informações para os tipos de eventos devem ser difundidas. Este utiliza um filtro de evento para verificar se a localização do nó está dentro do escopo do evento definido.

Uma outra abordagem é o trabalho desenvolvido em [24], o qual, utiliza o estado dos atributos do ambiente para mudar o serviço prestado por ele. Este trata dos aspectos de tempo real dos dispositivos locais do veículo e não considera as questões de comunicação entre os veículos, apenas se preocupa em sincronizar entre os nós, os serviços que estão sendo fornecidos entre os veículos.

O detector de defeitos de [6] possui um detector de conectividade entre processos vizinhos, envia mensagens periódicas para manter uma lista de vizinhança em cada veículo envolvido e considera um processo como vizinho o veículo que está dentro de um raio específico. Este detector dispõe de um algoritmo de detecção de defeitos que mantém uma lista de processos vizinhos suspeitos, a partir cálculo de tempo máximo de espera de cada vizinho da lista.

Os trabalhos apresentados possuem seus comportamentos definidos por suas heurísticas pré-definidas. Qualquer mudança no fluxo do serviço proposto deve ser alterado em tempo de projeto.

2.6 CONSIDERAÇÕES FINAIS

Apresentamos neste capítulo a comunicação veicular com suas principais classificações e tipos, também descrevemos a VANET e sua correlação com as redes *MANETs*. Além disso, discutimos sobre as tecnologias de comunicação para as redes veiculares com base no modelo *OSI* da *ISO*, descrevendo as soluções e principais desafios para as principais camadas do modelo.

Neste contexto, observamos que há uma tendência dos pesquisadores em adaptar as tecnologias de comunicação existentes atualmente para atender aos requisitos das aplicações veiculares. Percebemos também que para a comunicação veicular, aspectos importantes para as aplicações veiculares críticas como transmissão de mensagens fim-a-fim e de confiabilidade ainda são desafiadores e com muitas pesquisas e em andamento.

Além disso, apresentamos alguns serviços de comunicação para aplicações veiculares

que possuem heurísticas que utilizam atributos do ambiente para definir o comportamento do serviço proposto e prover algum tipo de adaptação do serviço em relação ao ambiente.

No próximo capítulo apresentaremos e discutiremos as principais características, requisitos e desafios da comunicação das aplicações veiculares.

CAPÍTULO 3

APLICAÇÕES SOBRE REDES VEICULARES

Neste capítulo apresentaremos os tipos de aplicações veiculares e como elas são classificadas em relação aos respectivos requisitos de comunicação. Dentre as aplicações discutidas destacamos a formação veicular, suas estratégias de controle do movimento e seus desafios em relação aos requisitos de comunicação.

3.1 APLICAÇÕES VEICULARES

Propagar mensagens de emergência em uma via, melhorar o tráfego de veículos diminuindo congestionamentos e poluições ao meio ambiente, permitir a realização de escalonamento de rotas de transporte, monitoramento de tráfego, entre outros são requisitos importantes para aplicações como sistemas automatizados de auto-estrada (*automated highway systems - AHS*) que podem ser classificadas como [33]:

- **aplicação de coordenação e assistência de tráfego** como a formação veicular, Pelotão de veículos e a ultrapassagem e mudança de faixa seguras.
- **aplicações veiculares seguras** que são aplicações de alta criticidade, a exemplo de sistemas para evitar colisões entre veículos.
- **as aplicações para o gerenciamento de tráfego** que podemos citar o monitoramento de tráfego, o escalonamento de sinais de trânsito para controlar o fluxo de veículos e avisos sobre veículos de emergência nas vias.
- **suporte para informações de viagem** como a atualização de informações locais e notificações sobre as vias como pista molhada, neblina, incêndio, etc.

- **aplicações para o bem-estar no trânsito** como a troca de mensagens instantâneas, imagem entre veículos, aplicação para realizar o pagamento de estacionamento, registros de manutenção e reparo, entre outros.

Essas aplicações se tornam um desafio para a área de *ITS (Intelligent Transport Systems)*, a qual pesquisa os aspectos científicos e tecnológicos da comunicação, informação e da eletrônica para atender aos requisitos das aplicações veiculares atuais [11].

Nas próximas seções serão apresentados os requisitos de comunicação das aplicações veiculares e o uso da inteligência artificial distribuída para atender às demandas de tais aplicações.

3.2 REQUISITOS DE COMUNICAÇÃO DAS APLICAÇÕES VEICULARES

As aplicações que necessitam de mecanismos para a comunicação veicular possuem requisitos de comunicação específicos em relação ao objetivo a ser atendido. De acordo com tais requisitos, as aplicações podem ser classificadas em: I. aplicações que provêm o controle do movimento individual do veículo; II. aplicações que provêm o controle do movimento coletivo de um grupo de veículos; III. aplicações que provêm serviços de informações no trânsito [42, 33].

3.2.1 Requisitos para Aplicações de Controle do Movimento Individual

As aplicações de serviço de controle do movimento individual emitem alerta de operador ou regulam atuadores do veículo para garantir a segurança e/ou a operação, como por exemplo, as aplicações veiculares seguras (ver Seção 3.1). Estas propagam os estados do movimento e dos atuadores, coletando dados para melhorar o desempenho do controle adaptativo do movimento [29, 12].

A latência das mensagens nestes tipos de aplicação é de tempo real firme e a confiabilidade na entrega das mensagens é considerada de probabilidade alta, pois, atrasos

e perdas de mensagens podem provocar acidentes ou movimentos abruptos durante a trajetória [42, 12].

A quantidade de veículos possível varia em pequena (abaixo de dez veículos), média (de onze até mil veículos) e alta (acima de 1000 veículos) de acordo com o objetivo da aplicação. Nessas aplicações os participantes são todos os veículos envolvidos, inclusive aqueles que encaminham ou consomem as mensagens e estão próximos uns dos outros [42, 33].

3.2.2 Requisitos para Aplicações de Controle Do Movimento Coletivo de Veículos

O serviço de controle de movimento de um grupo de veículos consiste em aplicações que planejam o movimento do veículo envolvendo otimizações globais ou negociações que podem ou não acionar a regulação do movimento individual de cada veículo, como por exemplo, as aplicações de coordenação e assistência de tráfego. Estas aplicações propagam, assim como, controlam as mensagens relacionadas ao movimento para aplicações centralizadas ou distribuídas [42, 33].

Neste tipo de aplicação, a depender do objetivo (e.g. formação veicular, *platoon*, planejar rotas para robôs, entre outros) a quantidade de veículos e a extensão da área alvo variam e é possível ou não tolerar algum atraso na entrega das mensagens. Contudo, deve-se garantir de forma determinística a entrega das mensagens e os veículos participantes devem ser conhecidos por todos os membros do grupo [42, 33].

Além disso, muitas vezes, as aplicações que controlam o movimento de um grupo de veículos necessitam conhecer e/ou definir (as vezes em tempo de execução) quais os veículos que devem participar do grupo. Entretanto, muitos dos protocolos para comunicação inter-veicular não apresentam um serviço de *membership*, fundamental para prover comunicação em grupo [8]. O serviço de *membership* gerencia a formação de um grupo de agentes e gerencia este grupo, provendo a cada agente do grupo uma visão de sua composição. O conhecimento da composição dos grupos é fundamental para permitir

o envio de uma mensagem para agentes do grupo (*multicast*) e não para todos os veículos existentes (*Broadcast*).

3.2.3 Requisitos para Aplicações de Serviços de Informações no Trânsito

Para manter a segurança dos veículos em vias públicas, as aplicações de serviços de informação provêm dados que alertam sobre potenciais perigos, tais como, anormalidades no comportamento do veículo ou nas condições do ambiente. A latência de entrega de mensagens deve ser de tempo real firme, pois, os atrasos podem comprometer o funcionamento e a usabilidade da aplicação. São sistemas críticos e necessitam de garantias, por isso, a confiabilidade é considerada de probabilidade alta. As mensagens podem alcançar uma grande quantidade (mais de 1000) de veículos ao longo de toda a rede ou trajetória e, para os veículos participarem, basta serem capazes de encaminhar e/ou consumir tais mensagens. Sistemas para evitar colisões ou para cruzamento inteligente de vias são exemplos de aplicações com tais requisitos [42, 33].

Existem aplicações de serviços de informação que estão direcionadas, principalmente, ao conforto, entretenimento e no provimento de dados não críticos. Nestes casos, a latência de entrega é a típica da rede utilizada e a entrega das mensagens é a de melhor esforço [42, 33].

3.3 APLICAÇÕES BASEADAS EM SISTEMAS MULTIAGENTES

A inteligência artificial esforça-se para solucionar problemas cada vez mais complexos, realistas e de grande escala. Tais problemas estão, muitas vezes, além das capacidades de um componente computacional individual. Pois, a capacidade desse componente é limitada pelo seu conhecimento, seus recursos de computação e suas perspectivas [36].

Analogamente, surge a necessidade de agrupar componentes com objetivos coletivos e associar aspectos sociais da inteligência humana, baseando-se no fato de que a maioria das

atividades inteligentes dos seres humanos é realizada por mais de uma pessoa [35, 4, 43]. Assim surgiu, no início dos anos 80, a inteligência artificial distribuída (*IAD*) e é baseada nos conceitos de agentes inteligentes da IA [35, 36].

Entre as principais técnicas da IAD destaca-se os Sistemas Multiagentes (SMA - Sistema Multiagente – *MAS Multiagent Systems*) que tratam da gestão do comportamento em coleções de diversas entidades (agentes) independentes [9, 35].

Os sistemas multiagentes são utilizados para a construção de aplicações complexas baseadas em múltiplos agentes que interagem entre si, possuem objetivos, ações e conhecimento coletivo [35]. As aplicações baseadas em *SMA* podem ser divididas em dois grupos principais: 1) o assistente de software pessoal; 2) os sistemas distribuídos [43].

O *Assistente de Software Pessoal* é um grupo de sistemas multiagentes, no qual, os agentes realizam suas funções de assistente proativo para os usuários trabalharem em alguma aplicação, por exemplo, agentes para o gerenciamento de fluxo de trabalho e de processo de negócios [43].

No grupo de *Sistemas Distribuídos*, os agentes se tornam nós de processo em um sistema distribuído, como por exemplo, as aplicações de sensoriamento distribuído. Tais aplicações usam os sistemas multiagentes para gerenciar as redes de sensores distribuídos espacialmente, e tem como objetivo monitorar e localizar todos os veículos envolvidos no sistema. Um exemplo, é a troca de informações de quando um veículo irá passar em uma região sensoriada [43]. Neste grupo estão as aplicações veiculares (e.g. robôs, aviões, carros, submarinos, etc.) que usam abordagens de sistemas multiagentes para realizar a coordenação, controle e planejamento dos veículos nas vias, como pode ser visto em [41].

Agente Inteligente

Um agente inteligente pode ser definido como um sistema de computador que está situado em algum ambiente, e que é capaz de realizar ações autônomas, a fim de cumprir os objetivos para os quais foi projetado [41].

Desse modo, as ações realizadas estão associadas diretamente ao comportamento do ambiente para o qual o agente foi implementado. Um agente possui uma ou mais das seguintes características [44]:

Autonomia – operação sem a intervenção humana ou de outros dispositivos e tem algum tipo de controle sobre suas ações e estados internos;

Reativa – capacidade de perceber seu ambiente e responder às alterações que nele ocorrem, a fim de satisfazer seus objetivos de projeto;

Veracidade – sua hipótese, resultado ou valor são sempre verdadeiros;

Sociável (com habilidade social)– capacidade de interagir com outros agentes (e possivelmente humanos) a fim de satisfazer seus objetivos de projeto;

Deliberativa – possui a capacidade de decidir e interferir no comportamento do ambiente;

Móvel– a habilidade de se mover entre os ambientes e/ou plataformas. São capazes de transmitir seus programas e estados por meio de uma rede eletrônica e recomeçar a sua execução em outro local;

Benevolente– agentes cujos os objetivos não são conflitantes;

Situados ou estacionários – o comportamento é contrário aos móveis. Isto é, são fixos em um mesmo ambiente e ou plataforma. Não se movimentam em uma rede;

Competitivas – são agentes que competem entre si para a realização de seus objetivos ou tarefas. Ou seja, não há colaboração entre os agentes;

Tais características ajudam a determinar a arquitetura de um agente inteligente. A arquitetura, por sua vez, define a estrutura necessária para atender às características do agente, como por exemplo: arquitetura baseada no reflexo do agente; arquitetura baseada no objetivo do agente; arquitetura baseada na necessidade de aprendizagem do agente, entre outros.

3.3.1 Formação Veicular

A formação veicular permite que um grupo de veículos interaja e se movimente entre si para atender às missões individuais e/ou coletivas, como por exemplo, um grupo de veículos mantendo uma trajetória pré-definida [7, 10], mesmo desviando de obstáculos e evitando colisões.

Tal aplicação é o tipo de aplicação de coordenação e assistência de tráfego (ver Seção 3.1). Muitas dessas aplicações fazem uso de técnicas de sistemas multiagentes para realizar a coordenação do movimento de um grupo de veículos autônomos para manter uma formação desejada, como por exemplo em [12, 29], entre outros. Nestas consideram um veículo como um agente autônomo inteligente o qual toma suas próprias decisões sobre as ações que deve executar em relação ao ambiente.

Existem dois problemas fundamentais na formação veicular. Um diz respeito a coordenação do movimento individual de cada veículo, para o qual leis de controle são especificadas a fim de controlar os mecanismo de direção, sentido, aceleração, entre outros. O outro consiste na manutenção da posição e movimento em relação aos outros veículos de um grupo [42],[23] e [12]. Para solucionar tais problemas, pesquisadores têm desenvolvido estratégias de controle de formação veicular [23] [12].

3.3.1.1 Estratégia de Controle de Formação

Há muitas questões que precisam ser consideradas em um projeto de controlador para a formação de veículos móveis, tais como a estabilidade da formação, a controlabilidade de diferentes padrões da formação, a confiabilidade (*safety*) e as incertezas das formações. Ao longo do tempo, pesquisadores [7] vêm desenvolvendo estratégias de controle que atendam a estes questionamentos. Por exemplo, a abordagem baseada em comportamento é combinada com a aplicação do controle de formação criando esquemas básicos de motores em cada veículo. Cada esquema gera um vetor que representa o comportamento desejado de resposta aos estímulos sensoriais. Nestes esquemas possíveis são incluídos o

anti-colisão, o desvio de obstáculos, a busca das metas, sempre mantendo uma formação. A ação de controle de cada veículo é uma média ponderada dos vetores de controle para cada esquema do comportamento do motor. São várias as desvantagens, uma delas é a não interação com os outros veículos, permitindo não identificar o comportamento real do grupo.

Uma outra estratégia é controlar a formação de um grupo de veículos baseado em líder-seguidor. Nesta, o grupo possui um líder o qual é responsável por controlar a movimentação dos seus seguidores, por meio de envio de mensagens (dos seguidores para o líder). Uma das vantagens é a facilidade da implementação. Contudo, um problema típico é a falta de retorno de informações do estado dos seguidores que podem ser deixados para trás, separando-os da formação, se o movimento do líder não for controlado precisamente. Além disso, o líder é um ponto de falha crítico e o desempenho total do sistema irá degradar catastróficamente. A tendência de muitas pesquisas têm sido concentrada em estratégias de controle cooperativo descentralizadas ou distribuídas para superar este tipo de problema [7].

Existem várias outras estratégias de controle para a formação veicular e a maioria tem como pilares as estratégias centralizadas ou descentralizadas, a exemplo em [7] das que são baseadas em mais de um grupo de líderes, as que utilizam coordenadas dos seus veículos vizinhos (e.g. localização, orientação e formato), as que permitem criar estruturas virtuais dinamicamente, entre outros.

3.3.1.2 Estratégia de Controle de Formação Veicular Distribuída

As estratégias distribuídas ou descentralizadas de controle de formação cooperativo possuem leis de controle acopladas a cada veículo que faz as suas próprias decisões de acordo com os estados dos seus vizinhos. Os veículos interagem entre si trocando mensagens sobre os seus estados. A vantagem é que isso permite que o grupo continue a alcançar um objetivo ainda na presença de falhas de qualquer membro [7].

Como resultado, a execução descentralizada oferece mais flexibilidade, confiabilidade e robustez do que a alternativa correspondente a estratégia centralizada. Uma desvantagem é que cada veículo precisa manter a sincronização dos dados entre os veículos envolvidos na formação, o que representa um aumento da complexidade e dos requisitos de comunicação, especialmente quando uma grande quantidade de veículos está envolvida no sistema [7].

3.3.1.3 Requisitos para a Comunicação na Formação Veicular Distribuída

Para manter uma formação entre veículos autônomos distribuídos é necessário a troca de informações sobre os estados de cada veículo, como por exemplo, velocidade, posição, sentido, entre outros. Além disso, é necessário que cada veículo conheça seus vizinhos para garantir, não só uma formação entre eles, mas também para garantir a segurança e um melhor desempenho no cumprimento da missão. Esta troca de informações é periódica, a partir da estratégia de controle de formação (ver seção 3.3.1.1) adotada pelo sistema.

Um ambiente de formação veicular é formado, basicamente, pelos sistemas de comunicação intra-veicular (interligam equipamentos do próprio veículo) e inter-veicular (interligam veículos e dispositivos nas vias) que foram discutidos na Seção 2.1. Para que haja a comunicação entre os veículos em movimento é necessário fazer uso de tecnologias de comunicação como as *VANETs* (ver Seção 2.3) permitindo a transmissão e a recepção de informações.

Contudo, características intrínsecas dessas redes como perda de pacotes e atrasos podem causar instabilidade nos sistemas de controle do movimento e da formação do grupo de veículos e podem levantar questões de segurança no pelotão de formação em sistemas aéreos e de auto-estrada, por exemplo [42].

Neste tipo de aplicação, o número máximo de veículos em uma formação veicular é pequeno e a extensão da área alvo das mensagens está entre os veículos em formação, o atraso na entrega das mensagens deve ser considerado de tempo real firme e a entrega das mensagens deve ser determinística e os veículos participantes devem ser conhecidos

por todos os membros do grupo [42, 33].

A partir de tais requisitos observamos que a formação veicular distribuída necessita de mecanismos de comunicação que possam coordenar a comunicação entre os veículos para diminuir os efeitos das características da rede sobre a aplicação.

Para isso, é necessário criar serviços de comunicação em grupo como: *membership* que identifica a existência de vizinhos (veículos e dispositivos fixos nas vias) e os veículos que participam da formação; o *multicast* confiável que provê a transmissão de uma mensagem para o grupo de veículos garantindo que todos os membros irão receber a mensagem.

Além desses, outros serviços são importantes para a comunicação distribuída, como por exemplo: os serviços de detecção de defeitos que busca identificar se existe na formação algum veículo com falha na comunicação; os serviços *unicast* e *multicast* confiáveis que proveem a transmissão de mensagens fim-a-fim garantindo a entrega da mensagem de acordo com o tipo de estratégia de formação veicular adotada; o serviço de **Entrega Ordenada de Mensagens** que busca controlar a sequência de entrega das mensagens transmitidas pelos veículos quanto a sequência de envio das mensagens dos veículos que contribui, principalmente, para a estabilização do movimento do veículo, entre outros; o serviço de **Tolerância a Falhas** que busca manter o serviço de comunicação entre os veículos mesmo na presença de falhas no sistema de comunicação.

3.4 CONSIDERAÇÕES FINAIS

Este capítulo mostrou os conceitos e os tipos de aplicações veiculares, seus requisitos específicos e de comunicação. Além de apresentar a relação destas aplicações com as técnicas de inteligência artificial distribuída, como os sistemas multiagentes.

Os serviços de comunicação para aplicações distribuídas são essenciais para o funcionamento da formação veicular distribuída, pois buscam atender aos requisitos de comunicação da aplicação desejada.

CAPÍTULO 4

A PROPOSTA DE SERVIÇOS INTELIGENTES DE COMUNICAÇÃO PARA APLICAÇÕES VEICULARES

Neste capítulo apresentamos uma abordagem para a construção de serviços inteligentes de comunicação para aplicações veiculares – a qual se baseia no uso de regras para adaptar os serviços de comunicação à dinâmica das redes veiculares. A partir desta abordagem, propomos o *I-CAR* (*Intelligent Communication Mechanism for Multi-Agent Vehicular Applications*) que é um conjunto de serviços inteligentes de comunicação multiagente baseado em regras sobre redes veiculares. Tal mecanismo busca permitir que um agente veicular possa tomar decisões sobre as ações da comunicação que serão realizadas durante a execução de uma aplicação veicular.

A abordagem baseada em regras e o desenvolvimento do *I-CAR* são apresentados nas próximas seções. Para tanto, na Seção 4.1 descreve a estrutura da abordagem proposta e, na Seção 4.2 está a definição dos serviços a serem providos pelo *I-CAR*, com os seus respectivos comportamentos.

4.1 UMA ABORDAGEM PARA O DESENVOLVIMENTO DE SERVIÇOS INTELIGENTES DE COMUNICAÇÃO

Nesta seção, apresentamos uma abordagem para o desenvolvimento de serviços de comunicação para redes veiculares baseados em regras. Esta abordagem permite o uso de regras, considerando as características do ambiente de comunicação nos seguintes aspectos:

- *Dinâmica* – as regras podem ser usadas para perceber o estado do ambiente em relação a capacidade de comunicação dos nós, permitindo, por exemplo, que cada

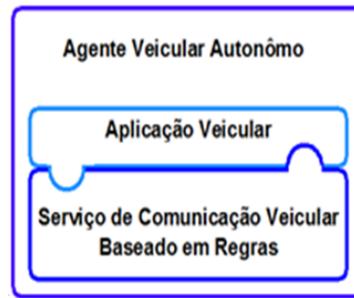


Figura 4.1. Mecanismo de Comunicação e o Agente Veicular

agente veicular tenha uma visão da topologia da rede e saiba com quais agentes o mesmo é capaz de comunicar. Além disto, permitir a adaptabilidade dos serviços à dinâmica do ambiente, permitindo introduzir novas regras em relação ao serviço de comunicação provido.

- *Heterogeneidade* – neste aspecto, as regras podem ser aplicadas para de modo que cada agente veicular tenha suas próprias regras e visão do estado do ambiente de comunicação. Isto é importante para garantir que cada agente tome suas decisões em relação a sua atuação no ambiente.
- *Requisitos da comunicação* – agentes veiculares podem usar regras para acionar comportamentos que garantam o atendimento de diferentes requisitos de comunicação. Para aplicações com requisitos temporais (e.g. formação veicular), por exemplo, regras podem ser usadas para decidir a melhor estratégia de encaminhamento das mensagens, garantindo assim o cumprimento dos limites temporais estabelecidos pela aplicação.
- *Demanda por serviços de comunicação* – na implementação dos serviços de comunicação, a base de regras pode ser reutilizada por diferentes serviços, permitindo que diferentes demandas de comunicação possam ser atendidas combinando o acionamento de diferentes regras previamente implementadas para uma ou mais aplicações veiculares distribuídas. Aplicações como a formação veicular, por exemplo, podem precisar de serviços de comunicação como monitoramento e detecção de defeitos, neste caso, regras básicas de transmissão de mensagens podem ser reutilizadas.

4.1 UMA ABORDAGEM PARA A CRIAÇÃO DE SERVIÇOS INTELIGENTES DE COMUNICAÇÃO33

Para desenvolver um serviço de comunicação baseado em regras, considera-se que este será encapsulado em um agente veicular, o que já possui as aplicações veiculares e os dispositivos e protocolos de rede necessários para a comunicação, como mostra a Figura 4.1.

A Figura 4.2 mostra uma estrutura baseada nas características de agentes inteligentes apresentadas na Seção 3.3. Esta consiste em um conjunto de elementos que deve compor um serviço de comunicação baseado em regras, tais são: o **estado do ambiente** que corresponde a um conjunto de propriedades que reflete a situação da comunicação vista pelo próprio agente; as **ações** que correspondem a um conjunto de tarefas que tratam exclusivamente dos recursos do sistema, podendo ou não interferir no ambiente; **os serviços de comunicação** são compostos de um conjunto de **regras** e ações inter-relacionadas e buscam prover serviços como *broadcast*, *unicast*, entre outros; as **interfaces** que servem para a interação entre as aplicações veiculares do agente e o conjunto de serviços de comunicação provido.

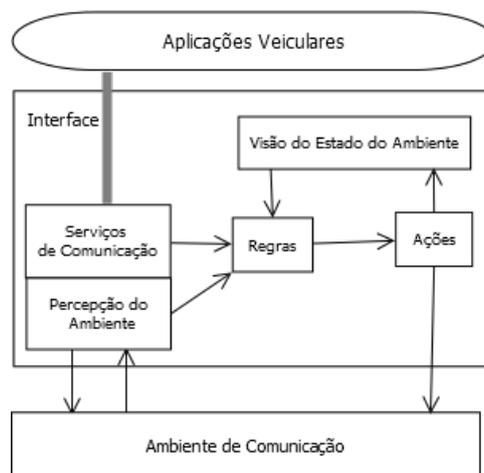


Figura 4.2. Estrutura do Mecanismo de Comunicação

O serviço de comunicação baseado na estrutura apresentada na Figura 4.2, percebe o ambiente de comunicação em torno do agente, atualiza o estado deste ambiente e, por meio de regras, toma suas próprias decisões sobre quais as ações deverão ser executadas.

4.1 UMA ABORDAGEM PARA A CRIAÇÃO DE SERVIÇOS INTELIGENTES DE COMUNICAÇÃO³⁴

Nas próximas seções serão detalhados os elementos que compõem esta estrutura de um serviço de comunicação baseado em regras: em 4.1.1 fala sobre a percepção do ambiente; em 4.1.2 apresenta a definição da visão do estado do ambiente de comunicação e nas Seções 4.1.3, 4.1.4 e 4.1.5 estão apresentados, respectivamente, os conceitos de ação, regra e serviço.

4.1.1 Percepção do Ambiente de Comunicação

Assumimos que cada agente mantém sua visão do estado do ambiente de comunicação, por meio de suas próprias regras de atualização, as quais devem ser construídas a partir de um conjunto de informações do ambiente (também chamado de conjunto de atributos do ambiente), como por exemplo, taxa de atraso de mensagens, *timeout* para recebimento de mensagens, tipo de nó de encaminhamento, taxa de alcance de sinal, entre outros.

Para isso, a percepção do ambiente atualiza a visão do estado do ambiente, a partir da coleta de informações sobre a comunicação. Esta coleta é realizada por regras e ações: 1) que percebem as mensagens que transitam na rede; 2) que buscam informações dos recursos de comunicação existentes para a comunicação. A percepção pode ser composta por serviços de monitoramento e de detecção de defeitos (também baseados em regras) para manter as informações atualizadas.

Vale ressaltar que, os dados a serem coletados estão implicitamente informados nas regras e ações associadas à percepção do ambiente. Por exemplo, dados sobre o tempo de viagem das mensagens trocadas entre dois nós podem ser coletados, calculados e atualizados por um conjunto de regras e ações.

4.1.2 Visão do Estado do Ambiente de Comunicação

A visão do estado do ambiente de comunicação possui informações sobre a situação da comunicação de um agente em relação ao ambiente o qual está inserido. Esta situação

consiste em um conjunto de propriedades gerais sobre a comunicação, tais são: a relação entre um agente local (a) e um agente remoto (b); a relação de vizinhança entre um agente local (a) e um agente remoto (b); estado da comunicação de um agente local (a) em relação ao grupo.

Estado entre o agente local (a) e um agente remoto (b)

O agente local (a) possui uma visão própria do estado da comunicação com cada agente remoto (b) de um grupo.

Para isso, são definidas quatro situações que especificam o modo com o qual o agente remoto (b) enviou e/ou recebeu informações do agente (a) em um instante de tempo recente, tais são:

- ***Communicating* (Comunicável)** - O agente (a) consegue transmitir e receber informações do agente (b). Assim, o agente (b) é um agente *Communicating* em relação ao agente (a).
- ***Non-Communicating* (Incomunicável)** - O agente (a), não recebe e não transmite mensagens para (b). Assim, o agente (b) é um agente *Non-Communicating* em relação ao agente (a).
- ***Receiving* (Receptor)** - O agente (a), apenas consegue transmitir mensagens para o agente (b). (a) reconhece que transmitiu, mas, não possui garantias de que a mensagem foi realmente recebida por (b). Mesmo assim (a) considera que, o agente (b) é um receptor de informações de (a).
- ***Transmitting* (Transmissor)** - A partir da verificação de um conjunto de atributos do ambiente, o agente (a) percebe que apenas recebe mensagens de (b), assim, o agente (b) é um transmissor de informações para o agente (a).
- ***Faulty* (Falho)** - o agente local (a) considera falho um agente remoto (b), quando (a) percebe que (b) falhou.

1	se o agente (<i>a</i>) consegue transmitir para o agente (<i>b</i>)
2	e o agente (<i>a</i>) consegue receber informações do agente (<i>b</i>)
3	e o atraso de mensagens entre os agentes (<i>a</i>) e (<i>b</i>) é menor que 100ms então
4	o agente (<i>b</i>) é um agente Communicating em relação ao agente (<i>a</i>)
5	fim

Figura 4.3. Exemplo de uso de atributos do ambiente para definir a situação da comunicação entre dois agentes

Cada situação é definida pela percepção da transmissão e recebimento de mensagens e, caso necessário, podem ser adicionados outros atributos do ambiente. O exemplo apresentado na Figura 4.3, além de utilizar o recebimento e transmissão, adiciona a informação do atraso de mensagens como uma informação adicional para identificar se o estado entre dois agentes é *Communicating*.

Modelo de Transição de Estado

Nesta seção é apresentada a transição do estado de um agente remoto (*b*) em relação ao agente local (*a*). Quando o agente remoto (*a*) percebe um novo evento de transmissão, recepção, *timeout*, entre outros, ele atualiza a sua visão sobre o estado de (*b*). A Tabela 4.1 mostra o estado inicial do agente (*b*), o evento que dispara a mudança do estado do agente remoto (*b*) e o estado final após a alteração realizada.

O estado da vizinhança entre dois agentes

A condição da vizinhança é uma das propriedades que compõe a situação da comunicação entre o agente local (*a*) e o agente remoto (*b*), o qual determina se um agente consegue enviar e receber informações com ou sem a presença de agentes intermediários para encaminhar as mensagens transmitidas. Para isso, são definidas duas situações de adjacência entre um par de agentes:

- **Não-adjacente** – chama-se um agente remoto (*b*) de não-adjacente ao agente local (*a*) quando durante a comunicação entre eles é necessário utilizar agentes in-

Tabela 4.1. Tabela de Transição dos Estados da Comunicação Entre Dois Agentes

Id	Estado Anterior do agente remoto de b	Descrição do Evento	Estado do agente remoto (b) após o evento
1	Non-Communicating	O agente (a) recebe mensagem de (b)	Transmitting
2	Non-Communicating	O agente (a) transmite mensagem para (b)	Receiving
3	Non-Communicating	O agente (a) não transmite para e nem recebe mensagem de (b)	Faulty
4	Faulty	O agente (a) recebe mensagem de (b)	Transmitting
5	Transmitting	O agente (a) transmite mensagem para (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Communicating
6	Transmitting	O agente (a) recebe mensagem de (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Transmitting
7	Transmitting	A validade do estado do agente (b) expirou em (a)	Non-Communicating
8	Receiving	O agente (a) recebe mensagem de (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Communicating
9	Receiving	O agente (a) transmite mensagem para (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Receiving
10	Receiving	A validade do estado do agente (b) expirou em (a)	Non-Communicating
11	Communicating	O agente (a) transmite mensagem para (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Communicating
12	Communicating	O agente (a) recebe mensagem de (b) e o tempo de validade do estado de (b) NÃO expirou no agente local (a)	Communicating
13	Communicating	O tempo de validade do estado de (b) expirou no agente local (a)	Non-Communicating
14	Receiving	O tempo de validade do estado de (b) expirou no agente local (a)	Non-Communicating
15	Transmitting	O tempo de validade do estado de (b) expirou no agente local (a)	Non-Communicating

intermediários para encaminhar as mensagens. Um agente remoto (*b*) não-adjacente, pode ser transmissor, receptor, comunicável e incomunicável em relação ao agente local (*a*).

- **Adjacentes** – chama-se um agente remoto (*b*) de adjacente, aquele que consegue se comunicar com o agente local (*a*) sem o uso de encaminhamento de mensagens. Um agente remoto (*b*) adjacente, pode ser transmissor, receptor e comunicável em relação ao agente local (*a*).

Considera-se que, provavelmente, um agente remoto (*b*) que é transmissor e adjacente ao agente local (*a*), conseguirá receber informações do agente (*a*), já que consegue, em nível de enlace, se comunicar com (*b*) para receber as mensagens. Assim como, se um agente remoto (*b*) é receptor e adjacente ao agente local (*a*), provavelmente, conseguirá enviar informações ao agente (*a*), já que consegue, em nível de enlace, se comunicar com (*b*).

Estado da comunicação de um agente local em relação ao grupo

Nesta seção iremos discutir sobre o estado de um agente local (a) sobre o ponto de vista do grupo de agentes envolvidos. Esta especificação ajuda ao próprio agente, a partir de suas regras, perceber a sua situação em relação ao ambiente de comunicação e possa disparar ações corretivas, preventivas e/ou evolutivas para melhorar o desempenho na comunicação. Para isso foi definido o estado da comunicação no agente local (a) em relação ao grupo de agentes envolvidos na comunicação.

Um agente local (a) considera o seu estado como *Communicating* em relação ao grupo de agentes quando (a) recebe e envia informações de e para pelo menos um dos agentes remotos do grupo. O agente também está *Communicating* se recebe de um e transmite para outro agente de um mesmo grupo.

O agente local (a) considera parcialmente comunicável em relação ao grupo de agentes, quando: é capaz, apenas, de receber (*receiving*) informações de outros agentes do grupo; é capaz, apenas, de transmitir (*transmitting*) informações para outros agentes do grupo.

Além disso, o agente local (a) é incomunicável para o grupo de agentes quando não consegue receber e nem transmitir informações a qualquer agente do grupo.

Para o grupo de agentes, o agente local (a) é falho quando: o defeito é detectado no processo local de transmissão e/ou de recepção; (a) está incomunicável para todos os agentes do grupo, por um período de tempo considerado longo¹ pelo agente; todos ou grande parte dos membros (*e.g.* a maioria ou um valor constante especificado pelo usuário) do grupo o consideram incomunicável ou falho.

Validade do Estado do Ambiente de Comunicação

A forma de interação entre os agentes podem ser modificada devido a frequência e o tipo de mudanças que podem ocorrer em um ambiente dinâmico de comunicação. Por

¹Período de tempo longo pode ser estabelecido pelo usuário ou por parâmetros de qualidade de serviço, por exemplo.

isso, é importante que a visão do estado da comunicação tenha uma validade. Esta validade indica, por quanto tempo este estado pode estar verdadeiro no sistema.

A validade pode ser definida para o estado de cada agente remoto (*b*) em relação a um agente local (*a*) ou pode ser determinada de forma global para todos os agentes.

O cálculo para determinar a validade da visão do estado do ambiente de comunicação pode ser determinado, por exemplo: pelos requisitos de comunicação da aplicação veicular; pelos parâmetros de qualidade de serviço da aplicação a ser executada; pelo atraso das mensagens na rede; pela carga de mensagens na rede; dados geo-referenciados; entre outros.

4.1.3 Ações

Um ambiente de comunicação veicular é composto por um conjunto de recursos heterogêneos e dinâmicos, como por exemplo: protocolos e dispositivos de rede, sensores, serviços de encapsulamento de dados, memórias, entre outros. Estes recursos permitem a atuação do agente no ambiente de comunicação.

Esta atuação permite a configuração e/ou reconfiguração dos recursos para executar a ação desejada. A Figura 4.4 mostra um exemplo de uma ação.

```

1 enviar-mensagem(agente origem, agente destino, mensagem){
2 encapsula mensagem nos protocolos de rede;
3 configura os protocolos necessário e aptos;
4 transmite mensagem;
5 }

```

Figura 4.4. Exemplo de uma ação para o envio de mensagem entre dois agentes

Uma ação corresponde a um conjunto de procedimentos que permite o uso dos recursos desejados. Uma ação pode ser composta por outras ações e é disparada a partir da verificação de uma hipótese verdadeira. Além disso, as ações podem ser diferentes entre os agentes de acordo com as características do agente.

Na próxima seção iremos falar sobre as regras e suas relações com as ações no meca-

nismo de comunicação.

4.1.4 Regras

O mecanismo possui um conjunto de regras que compõe o conjunto de serviços de comunicação. Estas regras estão baseadas na visão do estado da comunicação de um agente e é composta por uma hipótese que quando torna-se verdadeira, uma outra pode ser verificada ou uma ação pode ser disparada.

1	se	<i>(o agente remoto c é transmissor de a)</i>	então
2		Dispare ação	
3	fim		

Figura 4.5. Exemplo de uma regra

Uma hipótese está baseada na visão do estado do ambiente de comunicação de um agente. Na Figura 4.5 está o exemplo de uma regra que verifica uma hipótese sobre o estado da comunicação de um agente *c* em relação a um agente *a* e caso seja verdadeira, uma ação é disparada.

Cada agente possui um conjunto de regras que compõe a respectiva base de conhecimento. Esta base de conhecimento contribui para a composição de serviços e da percepção do ambiente de comunicação.

4.1.5 Serviços de Comunicação

O mecanismo pode ser formado por um ou vários serviços de comunicação. Estes serviços determinam o fluxo de execução das regras e é composto por um conjunto de regras e ações que podem ser combinadas e/ou encadeadas com outras regras e ações já existentes. Estas combinações e/ou encadeamentos determinam o comportamento do serviço durante a execução.

A Figura 4.6 mostra um exemplo de encadeamento e combinação de regras gerando

```

1 se regra 1 então
2   | se regra 2 então
3   |   | dispare ação 1
4   |   | senão
5   |   |   | dispare ação 2
6   |   |   | fim
7   |   | fim
8   |   | se regra 3 então
9   |   |   | se regra 4 então
10  |   |   |   | dispare ação 3
11  |   |   |   | senão
12  |   |   |   |   | dispare ação 4
13  |   |   |   |   | fim
14  |   |   |   | fim
15  |   | fim
16 fim

```

Figura 4.6. Exemplo do encadeamento e combinações de regras para definir um serviço um fluxo de execução que determina o comportamento de um serviço proposto qualquer.

Vale ressaltar que, o comportamento de um mesmo serviço de comunicação entre os agentes pode ser diferenciado desde que não fuja às definições gerais sobre as propriedades da visão do estado do ambiente de comunicação e atendam aos objetivos proposto.

Os serviços de comunicação podem ser, por exemplo: serviços de monitoramento; serviços de *unicast*; serviços de *broadcast*; detectores de defeitos; serviços de *membership* etc.

4.2 O I-CAR

Utilizamos a abordagem baseada em regras discutida na Seção 4.1 para desenvolver o *I-CAR* (*Intelligent Communication Mechanism for Multi-Agent Vehicular Applications*) que é um conjunto de serviços inteligentes de comunicação multiagente baseado em regras sobre redes veiculares. Os serviços implementados foram o de monitoramento de estado da comunicação, o *broadcast*, o *unicast* e o detector de defeitos.

Consideramos o ambiente da aplicação da formação veicular distribuída, um grupo de veículos autônomos que interage entre si, por meio de dispositivos de comunicação sem

de comunicação será mantido pelos serviços de monitoramento e detecção de defeitos, também providos pelo *I-CAR*. Assim, o comportamento geral do *I-CAR* pode ser descrito:

1– O agente veicular aciona o mecanismo de comunicação e os serviços de monitoramento, de recebimento geral de mensagens e o de detecção de defeitos são ativados.

2– Caso necessário, a aplicação de formação veicular do agente solicita um serviço de transmissão de mensagens, como o *unicast* e/ou o *multicast*.

3– Ao longo do tempo o serviço de comunicação mantém o estado da comunicação a partir de sua própria percepção do ambiente de comunicação. Para isso, atualiza sua visão sobre o estado da comunicação a partir dos serviços de monitoramento, de recebimento e de detecção de defeitos.

4– Caso deseje, uma aplicação do agente pode solicitar um serviço de comunicação.

5– Cada serviço ativa um conjunto de regras associadas a ele e a cada mudança de estado da comunicação, as regras associadas ao serviço são verificadas e as ações correspondentes são disparadas.

Nas próximas seções apresentaremos a modelagem dos serviços providos pelo *I-CAR*. Inicialmente, discutiremos sobre o serviço de comunicação fim-a-fim *unicast*, descrevendo as estratégias para transmissão e encaminhamento de mensagens que definem as regras para a troca de informações entre os agentes do grupo de veículos em formação. Em seguida mostraremos as regras dos serviços de: monitoramento que permite detectar e atualizar a visão do estado do ambiente de comunicação em cada agente envolvido; do *broadcast* que busca transmitir mensagens a um grupo de veículos envolvidos na aplicação; de detecção de defeitos que busca verificar se um agente está falho ou não.

4.2.1 Serviço para a Comunicação Fim-a-Fim - Unicast

Este serviço foi criado para provê a transmissão fim-a-fim entre dois veículos em formação. Neste serviço utilizamos a transmissão de mensagens entre agentes, por meio de regras baseadas no estado da comunicação atualizado pelo mecanismo, como descrito

em 4.2.1.

Estratégia Geral para a Transmissão de Mensagens

Serviço geral para a transmissão de mensagem tem o objetivo de prover um mecanismo básico de transmissão de dados. Este permite a transmissão fim-a-fim de mensagens, sem conexão e sem garantias de entrega.

Este serviço utiliza a abordagem de difusão (*broadcast*) de mensagem e foi adotada para que todos os agentes que estivessem no alcance do agente de origem α recebessem suas informações e atualizassem a situação da comunicação de α na visão do estado de cada um.

Contudo, muitas vezes, para que uma mensagem em uma transmissão fim-a-fim chegue ao seu destino é necessário fazer uso de recursos de encaminhamento de mensagem, isso ocorre, principalmente, devido aos limites de alcance do sinal de comunicação do agente de origem em relação ao agente destino. Por isso, nesta versão do *I-CAR* adotamos uma estratégia para o encaminhamento de mensagens que será descrita a seguir.

Estratégia Para Encaminhamento de Mensagens

Nesta estratégia para a transmissão de mensagem é considerado que cada agente envolvido (agentes de origem e de encaminhamento) pode decidir por qual agente está apto a encaminhar a mensagem até o destino. Os grafos direcionados apresentados na Figura 4.8 mostram um exemplo de transmissão de uma mensagem do agente α para o ω .

Os grafos apresentados nas Figuras 4.8(a) mostra a visão de α sobre o envio da mensagem para ω . Neste caso, ele verifica que na sua visão do estado do ambiente que o agente β é o mais apto (naquele instante de tempo) para transmitir uma mensagem para ω , mesmo que este β não esteja adjacente a ω . Assim, o agente α envia a mensagem para

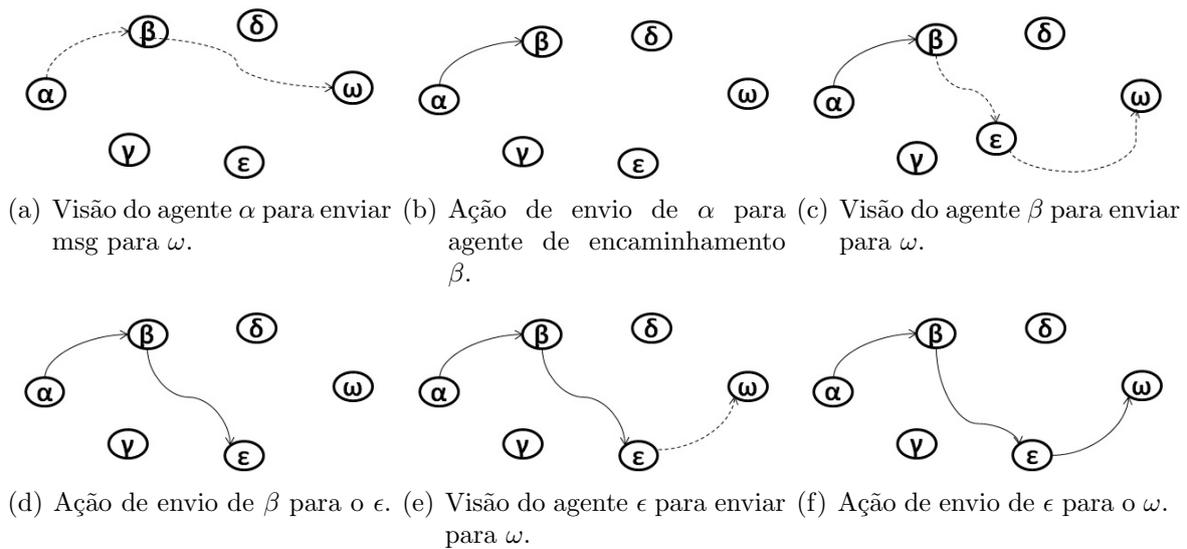


Figura 4.8. Transmissão de mensagem do agente α para o ω

o agente β como mostra a Figura 4.8(b).

Na Figura 4.8(c), o agente β verifica que na sua visão do ambiente de comunicação o agente ϵ é o mais apto a encaminhar a mensagem para ω e envia a mensagem para ϵ como mostra a Figura 4.8(d). Do mesmo modo, como mostram as Figuras 4.8(e) e 4.8(f), o agente ϵ verifica a situação do ambiente de comunicação e identifica que o agente ω é mais apto a receber a mensagem, então envia a mensagem para ω .

Para realizar a escolha de um agente apto a transmitir uma mensagem entre dois agentes, vimos necessidade de obter informações sobre a visão da comunicação de cada agente do ambiente. Para isso, em cada mensagem transmitida por um agente de origem α , são encapsuladas informações sobre o ambiente de comunicação visto por este agente (α). Assim, cada agente que receber a mensagem irá guardar em sua instância do *I-CAR*, a situação da comunicação vista pelo agente de origem. Esta informação é armazenada na matriz de visão do estado da comunicação de agentes remotos.

Assim, as regras desenvolvidas não só verificam o estado do agente destino em relação ao agente de origem, mas também verifica a situação do agente de encaminhamento em relação ao destino.

Ações para a Transmissão de Mensagens

As ações disparadas pelo serviço de transmissão de mensagens realizam a coleta das informações sobre a visão do estado do agente de origem de mensagem (caso o agente local seja o de origem), encapsulam as informações, tratam dos dispositivos (protocolos) de transmissão de mensagens e executa o envio por *broadcast*, tais ações são:

- enviar-mensagem-adjacente(a, b) - esta ação envia por meio de *broadcast* a mensagem do agente de origem para o agente destino.
- enviar-mensagem-non-adjacente(a, b, c) - esta ação envia a mensagem por meio de *broadcast* para o agente de encaminhamento (c) indicando que o agente destino é o agente (b).

A seguir apresentaremos as regras para transmissão de mensagens baseadas no estado do ambiente de comunicação de cada agente envolvido.

Regras para a Transmissão de Mensagens

As regras para envio das mensagens estão baseadas no estado dos agentes remotos relacionados ao agente local (a). O agente (a) verifica o estado da sua comunicação em relação ao agente remoto destino (b) e busca alternativas de transmissão direta ou por encaminhamento de mensagens. Nestas regras identificamos os agentes envolvidos da seguinte forma:

- a - agente local de origem;
- b - agente remoto de destino;
- c - agente de encaminhamento, caso a e b não sejam adjacentes.

Esta regra foi construída de forma hierárquica, onde a situação verificada pela regra anterior é considerada de melhor desempenho em relação a próxima regra. A Figura 4.9

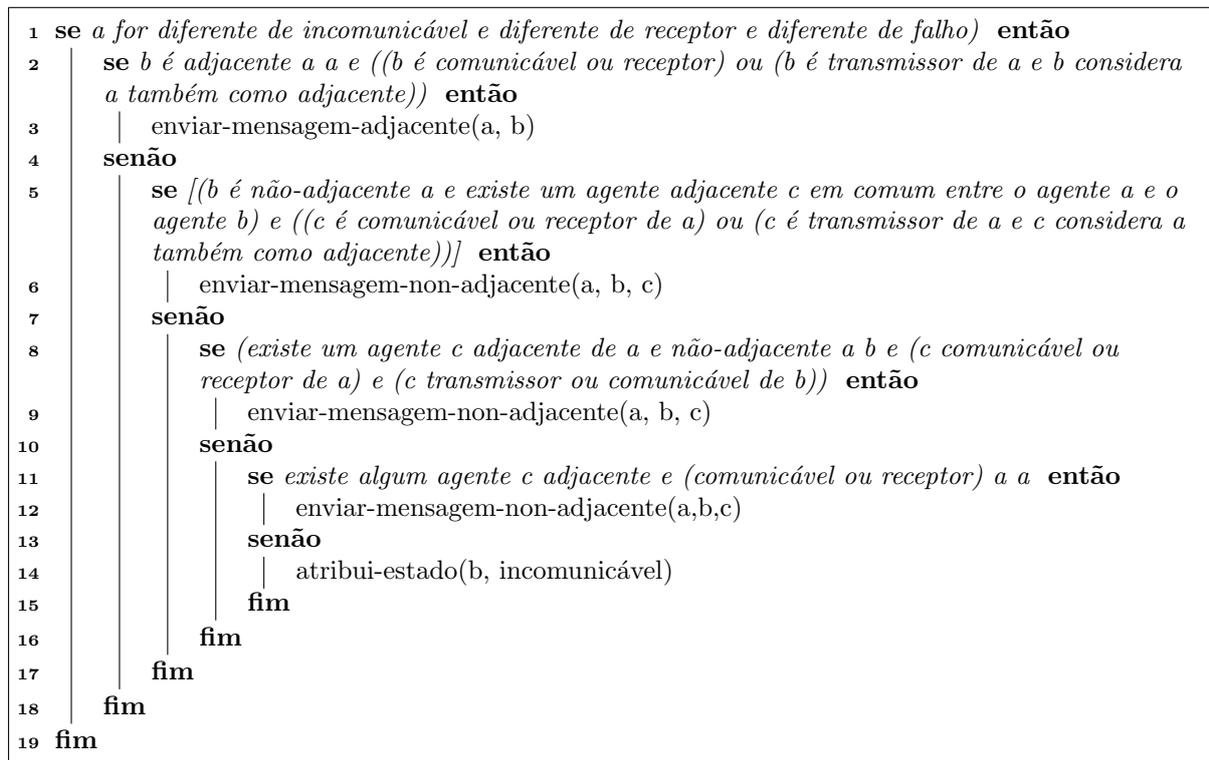


Figura 4.9. Regras para disparar ações para envio de mensagens

mostra o algoritmo que descreve as regras para a transmissão de mensagens que serão discutidas a seguir.

A primeira situação verificada na regra da Linha 1 do algoritmo da Figura 4.9 só permite a transmissão de mensagens do agente (a) se este estiver em um estado diferente de receptor, incomunicável e falha, pois, nestes estados o agente de origem está em uma situação que não consegue enviar mensagens para nenhum outro agente do ambiente (conforme descrito na Seção 4.1.2).

Na regra apresentada na Linha 2, o agente de origem (a) verifica se o agente (b) é adjacente, caso seja, este verifica a situação de (b) em relação a ele. Se (b) for comunicável ou receptor significa que (b) recentemente recebeu mensagens de (a) e é possível que (b) consiga receber mais uma mensagem de (a) e a ação de envio para agentes adjacentes é disparada. Contudo, (b) pode ser transmissor, neste caso, (a) verifica na matriz de estado de (b) se (b) considera (a) como adjacente, caso seja, isto indica que ambos se consideram

adjacentes e, por isso, existe a possibilidade de (b) receber informações diretamente de (a) mesmo sendo transmissor, assim, a ação de envio de mensagens para agentes adjacentes é disparada.

Nas transmissões entre agentes não-adjacentes desenvolvemos regras que escolhem o agente que irá encaminhar a mensagem. O critério de escolha está baseado na situação do agente (a) em relação ao (b) e a relação de (c) com (a) e com (b) , como mostrado nas linhas 5 e 14 da Figura 4.9. Caso, não exista um agente nesta situação (a) verifica, se existe um agente adjacente a (a) e não adjacente a (b) , mas que se comunique (comunicável ou transmissor de (b) - ver linhas 8 e 11) com (b) .

A situação descrita na Linha 5 da Figura 4.9, verifica se (b) não é adjacente a (a) e se existe um agente de encaminhamento (c) que é adjacente a (a) e a (b) e este (c) é comunicável ou receptor a (a) e transmissor ou comunicável a (b) . Neste caso, o agente (c) está apto a receber uma mensagem de (a) e enviá-la para (b) .

Se a hipótese da regra da Linha 5 for falsa, então na hipótese da Linha 8 verifica se existe um agente (c) adjacente a (a) , mas que não é adjacente a (b) , mas consegue receber mensagens de (a) e consegue se comunicar (transmissor ou comunicável a (b)) com (b) .

Caso a hipótese da Linha 8 seja falsa, (a) busca qualquer agente (c) adjacente a ele e que possa receber (receptor ou comunicável) uma mensagem dele. Se nenhuma dessas hipóteses forem verdadeiras atribua a (b) o estado de incomunicável em relação a (a) .

Estas regras são utilizadas para decidir por qual caminho será enviada uma mensagem entre pares de agentes e estão baseadas nas informações sobre o estado do ambiente e, para esta implementação, serão utilizadas nos serviços de monitoramento, detecção de defeitos, *unicast* e *multicast*.

4.2.2 Serviço de Monitoramento do Estado da Comunicação

O monitor de estados da comunicação utiliza mensagens trocadas pela aplicação distribuída e também geradas pelo próprio monitor para monitorar alterações no estado de comunicação de cada agente veicular. Tal monitoramento consiste em observar as alterações do ambiente e atualizar a visão do estado da comunicação do agente.

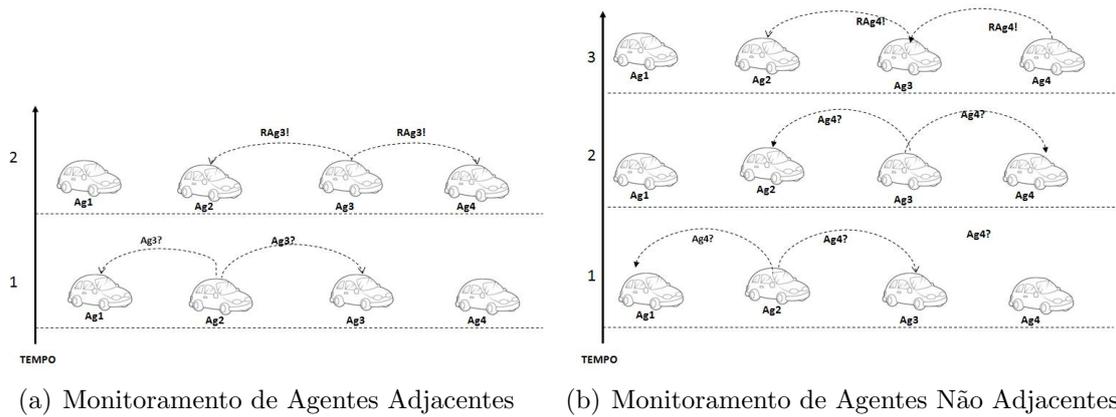


Figura 4.10. Dados Analisados

Inicialmente, o monitor utiliza as mensagens da aplicação veicular para atualizar o estado de um agente remoto (b) em um agente local (a). Caso, o estado da comunicação do agente (b) em relação a (a) esteja desatualizado o mecanismo dispara a solicitação de monitoração para o agente (b) e um tempo de espera é acionado, utilizando as estratégias apresentadas na Seção 4.2.2 e como mostra a Figura 4.10.

A Figura 4.10(a) mostra no tempo 1 o veículo *Ag2* enviando a solicitação do estado da comunicação para monitorar o estado da comunicação do agente veicular remoto *Ag3*, adjacente a *Ag2*. Neste caso, todos os agentes adjacentes a *Ag2* recebem a mensagem de solicitação e atualizam a própria visão do estado da comunicação de *Ag2*.

Na Figura 4.10(b) está apresentado no tempo 1, o agente *Ag2* enviando a solicitação do estado da comunicação do agente remoto não adjacente *Ag4* e, por isso, a mensagem de solicitação do estado precisa ser encaminhada. Assim, no tempo 2 da Figura 4.10(b) mostra o agente *Ag3* encaminhando a mensagem de solicitação para *Ag4*. Neste Caso, os agentes que estão adjacentes tanto de *Ag2*, quanto de *Ag3* recebem a mensagem de

solicitação de $Ag2$ e atualizam a sua própria visão do estado de $Ag2$ e aqueles que estão adjacentes a $Ag3$ recebem a mensagem de $Ag3$ e atualizam a sua visão de $Ag3$.

Ao receber a solicitação, os agentes monitorados enviam uma resposta sobre seu estado da comunicação, como mostra a Figura 4.10. Na Figura 4.10(a) no passo 2 o agente monitorado $Ag3$ verifica que o agente solicitante $Ag2$ é adjacente a ele, transmite a resposta a solicitação e todos os agentes adjacentes à $Ag3$ atualizam o estado da comunicação.

A Figura 4.10(b) mostra que o $Ag4$ verifica que $Ag2$ não é adjacente a ele e precisa encaminhar a resposta, assim, no tempo 2, $Ag4$ transmite para $Ag3$ que encaminha a mensagem de resposta para $Ag2$, no tempo 3, e os seus adjacentes atualizam a sua visão do estado de $Ag4$ e $Ag3$ de acordo com suas respectivas adjacências.

O envio das mensagens pelo próprio monitor é definido a partir da estratégia de monitoração adotada por cada veículo, ao monitorar cada um dos demais veículos envolvidos na aplicação. Esta estratégia é definida por um conjunto de regras utilizadas pelo agente para analisar o estado de comunicação anterior do veículo monitorado, para obter a resposta pretendida de forma eficiente.

As regras formam uma base de conhecimento que é adicionada ao agente veicular envolvido na aplicação, sendo utilizadas tanto para decidir pela melhor forma de monitorar cada um dos agentes, como para definir o estado de comunicação destes agentes como resultado da monitoração. As regras foram agrupadas em diferentes subconjuntos, baseados em sua utilização.

Neste contexto, nas próximas seções apresentamos o conjunto de ações e regras utilizadas pelo mecanismo como estratégias de monitoramento.

Ações dos Agentes para a Monitoração

Os agentes veiculares executam ações para enviar, responder e encaminhar mensagens de monitoração, assim como para atualizar o estado de comunicação dos diversos agentes monitorados. Mensagens enviadas pela própria aplicação são também percebidas

pelo monitor e utilizadas para atualizar os estados de comunicação, reduzindo assim a quantidade de mensagens enviadas pelo monitor. As ações executadas pelo monitor com relação ao envio de mensagens são:

- *Enviar-solicitacao-monitoramento-agente-adjacente(agente origem, agente destino)* - encapsula a mensagem de solicitação de monitoramento e utiliza a ação *enviar-mensagem-adjacente(a, b)* (ver Seção 4.2.1) para a transmissão da solicitação desejada.
- *Enviar-solicitacao-monitoramento-agente-não-adjacente(agente origem, agente destino, agente encaminhamento)* - transmite a solicitação de monitoração do agente de destino para o agente de encaminhamento, uma vez que o agente de origem e de destino não são adjacentes, utilizando a ação *enviar-mensagem-non-adjacente(a, b, c)* descrita na Seção 4.2.1.
- *Inserir-solicitacao-monitoramento(agente monitorado)* - insere a informação de que foi enviada uma solicitação de monitoração ao agente monitorado em uma lista no agente local.
- *Calcular-timeout-monitoramento-agente-adjacente(agente monitorado)* - calcula o tempo máximo de espera para a resposta da solicitação de monitoramento do agente monitorado.
- *Calcular-timeout-monitoramento-agente-não-adjacente(agente origem, agente monitorado)* - calcula o tempo máximo de espera para a resposta da solicitação de monitoramento do agente monitorado baseando nas condições de adjacência dos agentes.
- *Encaminhar-mensagem-solicitacao-monitoramento-agente-destino-adjacente(d, b, a)* - a mensagem de solicitação de monitoramento enviada pelo agente d é encapsulada e encaminhada pelo agente a para o agente monitorado b . Para isso, utiliza a ação *enviar-mensagem-adjacente(a, b)* (ver Seção 4.2.1) para a enviar a solicitação desejada.

- *Encaminhar-mensagem-solicitacao-monitoramento-agente-destino-nao-adjacente(d, b, a, c)* - encaminhamento da mensagem de monitoração quando os agentes local e monitorado (a e b) não são adjacentes.
- *Enviar-mensagem-resposta-monitoramento-agente-adjacente(a, b)* - transmite a resposta do estado de comunicação do agente b .
- *Enviar-mensagem-resposta-monitoramento-agente-nao-adjacente(a, b, c)* - transmite a resposta de monitoração do agente b para o agente intermediário c , uma vez que a e b não são adjacentes.
- *Excluir-solicitacao-da-lista-de-espera(b)* - ao receber informações recentes de (b), a solicitação de monitoramento de (b) é excluída da lista de solicitações de monitoramento.

As ações descritas nesta seção são disparadas pelas regras que compõem a base de conhecimento do monitor instanciado pelo agente, o qual é descrito na próxima seção.

Base de conhecimento do monitor de estados de comunicação

A base de conhecimento do monitor é composto por regras, que são ativadas quando a informação de estado de algum agente veicular remoto se tornar desatualizada, ou quando mensagens (do próprio monitor ou da aplicação) forem recebidas. As regras para o envio de mensagem de monitoração estabelecem uma estratégia para obter o estado do agente monitorado, a partir da informação de estado atual. Agentes adjacentes podem ser monitorados diretamente, enquanto agentes não adjacentes só podem ser monitorados através de outros agentes.

As regras são agrupadas em conjuntos de acordo com o seu objetivo. Temos regras para enviar a solicitação de monitoramento para o grupo, regras para encaminhar a solicitação de monitoramento, regras para enviar mensagem de resposta a solicitação de monitoramento, regras para encaminhar a mensagem de resposta e regras para receber

mensagem de resposta de solicitação de monitoramento. A base de conhecimento do monitor também possui um conjunto de regras para o recebimento de quaisquer mensagens da aplicação. Nas seções a seguir descrevemos o escopo e funcionamento de cada conjunto de regras.

Regras para o envio de mensagem de solicitação de monitoramento para agentes

O conjunto de regras que estabelece a estratégia para a solicitação de monitoramento é apresentado na Figura 4.11. Estas regras permitem ao agente monitor decidir sobre a melhor estratégia para monitorar agentes remotos em função do estado atual de comunicação destes agentes com relação ao agente local.

Para este conjunto de regras assumimos que:

- a - agente local, que gera a solicitação de monitoramento;
- b - agente remoto, o qual necessita ser monitorado;
- c - agente de encaminhamento de mensagens, adjacente ao agente local, quando o agente a ser monitorado não é adjacente; este encaminha a mensagem até o destino.

As regras indicam a ação a ser executada em função de o agente monitorado ser adjacente ou não, do estado atual de comunicação, de existir alguma solicitação de monitoração pendente para este agente e do resultado da última monitoração realizada. As alternativas para a solicitação de monitoramento são explicitadas pelas ações: *Enviar-solicitacao-monitoramento-agente-adjacente(a, b)*, *Enviar-solicitacao-monitoramento-agente-nao-adjacente(a, b, c)* e *Enviar-solicitacao-monitoramento-estado-agente-todos-agentes(a, b, c)*.

As regras utilizadas da linha 2 até a 12 da Figura 4.11 fazem parte das estratégias de transmissão de mensagem descritas na Seção 4.2.1. Quando nenhuma das estratégias de envio é possível, é selecionado um agente (c) que seja adjacente ao agente local a e comunicável ou receptor de (a) para tentar encaminhar a mensagem até (b) e a ação para

```

1 se a for não comunicável para todos os agentes
2 ou (existe agente remoto b no grupo que está com o tempo de atualização do estado expirado e
  não há solicitação de monitoramento de b em aberto) então
3   se b é adjacente a a e ((b é comunicável ou receptor) ou (b é transmissor de a e b considera
  a também como adjacente)) então
4     | Enviar-solicitacao-monitoramento-agente-adjacente(a,b)
5   senão
6     se [(b é não-adjacente a e existe um agente adjacente c em comum entre o agente a e o
  agente b) e ((c é comunicável ou receptor de a) ou (c é transmissor de a e c considera a
  também como adjacente))] então
7       | Enviar-solicitacao-monitoramento-agente-nao-adjacente(a,b,c)
8     senão
9       se (existe um agente c adjacente de a e não-adjacente a b e (c comunicável ou
  receptor de a) e (c transmissor ou comunicável de b)) então
10        | Enviar-solicitacao-monitoramento-agente-nao-adjacente(a,b,c)
11      senão
12        se ((c é transmissor de a) e (c é transmissor ou comunicável de b) e c
  considera a também como adjacente) então
13          | Enviar-solicitacao-monitoramento-agente-nao-adjacente(a,b,c)
14        senão
15          se existe algum agente c adjacente e (comunicável ou receptor) a a então
16            | Enviar-solicitacao-monitoramento-agente-nao-adjacente(a,b,c)
17          senão
18            | atribui-estado(b, incomunicável)
19          fim
20        fim
21      fim
22    fim
23  fim
24 fim
25 se (a) transmitiu uma mensagem para (b) recentemente e corretamente então
26   atribuir-estado(b, incomunicavel)
27   Inserir-solicitacao-monitoramento
28   se c!= b então
29     | Calcular-timeout-monitoramento-agente-nao-adjacente
30   senão
31     | Calcular-timeout-monitoramento-agente-adjacente
32   fim
33 fim

```

Figura 4.11. Regras para enviar solicitação de monitoramento

o enviar a mensagem a todos é disparada. Caso nenhum agente (*c*) para encaminhamento seja selecionado, (*b*) torna-se incomunicável.

Logo após o envio com sucesso da mensagem de solicitação de monitoramento (*a*) insere (*b*) em uma lista de solicitações de monitoramento, torna (*b*) incomunicável.

A ação *Calcular-timeout-monitoramento-agente-adjacente* é acionada quando os agentes de origem e o monitorado são adjacentes ao agente local para calcular o tempo máximo de espera para a atualização do estado do agente, considerando apenas um atraso no cálculo. Já a ação *Calcular-timeout-monitoramento-agente-não-adjacente* é acionada quando os agentes não são adjacentes entre si e calcula o tempo máximo considerando os possíveis encaminhamentos que serão realizados até chegar ao agente destino (*b*).

Regras para enviar mensagem de resposta a solicitação de monitoramento

Ao receber uma mensagem solicitando o seu monitoramento, e após utilizar o recebimento desta mensagem para atualizar o estado dos agentes envolvidos na solicitação, o agente monitorado envia uma resposta de monitoramento ao agente solicitante. Caso não sejam adjacentes, esta mensagem será enviada via encaminhamento. As regras para este envio são similares às apresentadas na Seção 4.2.1 e estão descritas na Figura 4.12. Para este conjunto de regras assumimos que:

- *a* - agente monitorado e local, que gera a resposta da solicitação de monitoramento;
- *b* - agente solicitante;
- *c* - agente de encaminhamento de mensagens, adjacente ao agente local, quando o agente a ser monitorado não é adjacente.

Caso, nenhuma das regras de envio seja possível é selecionado um agente (*c*) que seja comunicável ou receptor do agente local. Se não existe qualquer agente de encaminhamento possível o agente (*b*) torna-se incomunicável.

```

1 se a mensagem recebida é uma solicitação de monitoramento e a é o agente monitorado então
2   se b é adjacente de a e ((b é comunicável ou receptor) ou (b é transmissor de a e b considera
   a também como adjacente)) então
3     | enviar-resposta-monitoramento-agente-adjacente(a,b)
4   senão
5     se [(b é não-adjacente a e existe um agente adjacente c em comum entre o agente a e o
   agente b) e ((c é comunicável ou receptor de a) ou (c é transmissor de a e c considera a
   também como adjacente))] então
6       | Enviar-resposta-monitoramento-agente-nao-adjacente(a,b,c)
7     senão
8       se (existe um agente c adjacente de a e não-adjacente a b e (c comunicável ou
   receptor de a) e (c transmissor ou comunicável de b)) então
9         | Enviar-resposta-monitoramento-agente-nao-adjacente(a,b,c)
10      senão
11        se ((c é transmissor de a) e (c é transmissor ou comunicável de b) e c
   considera a também como adjacente) então
12          | Enviar-resposta-monitoramento-agente-nao-adjacente(a,b,c)
13        senão
14          se existe algum agente c adjacente e (comunicável ou receptor) a a então
15            | Enviar-resposta-monitoramento-agente-nao-adjacente(a,b,c)
16          senão
17            | atribui-estado(b, incomunicável)
18          fim
19        fim
20      fim
21    fim
22  fim
23 fim

```

Figura 4.12. Regras para enviar resposta de solicitação de monitoramento

Regras para o encaminhamento de solicitação de monitoração

Estas regras representam as decisões do agente quando receber uma mensagem de solicitação de monitoramento endereçada a um outro agente. As regras levam em consideração diversos aspectos do estado do ambiente de comunicação. Este conjunto de regras é apresentado na Figura 4.13.

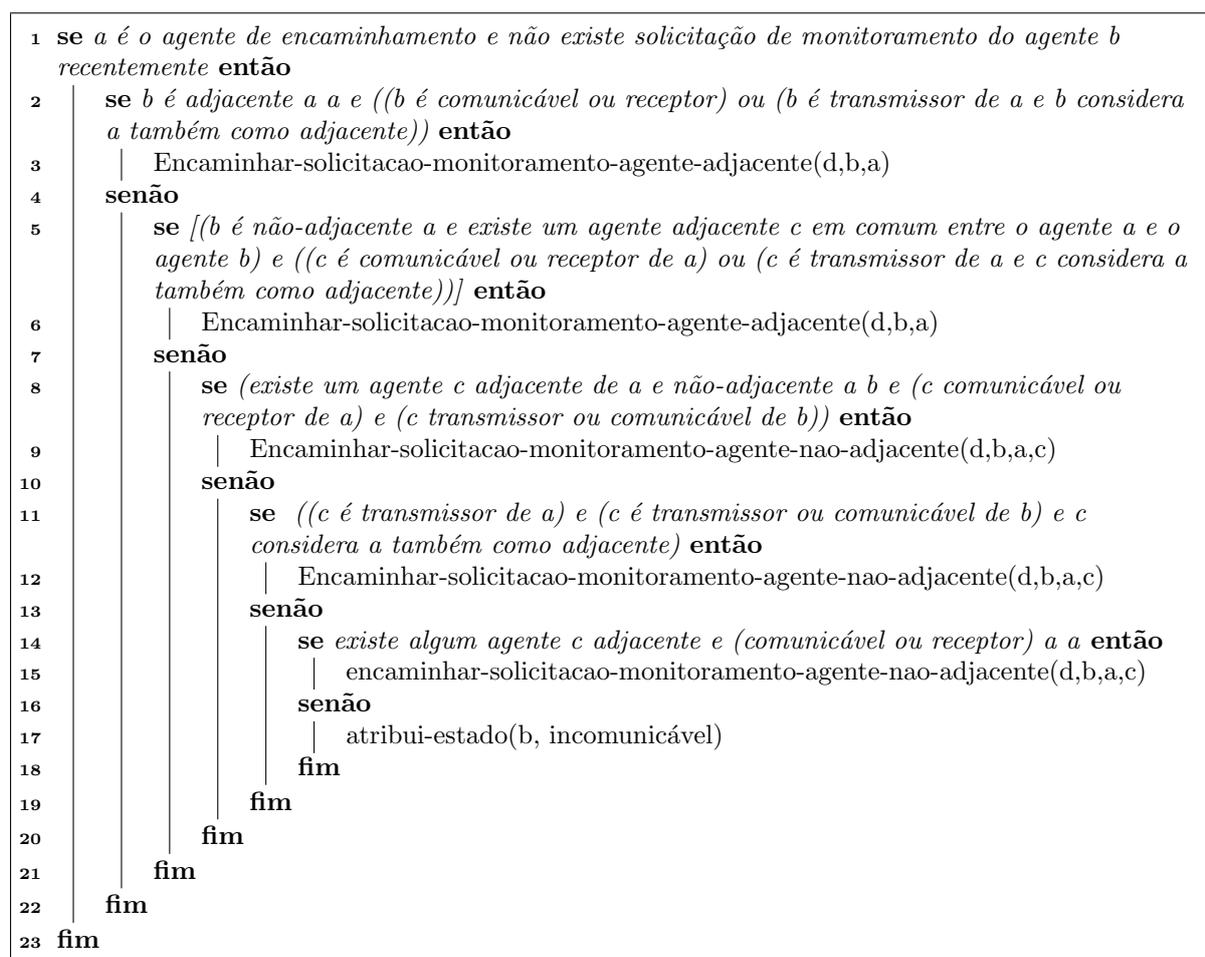


Figura 4.13. Regras para encaminhar uma solicitação de monitoramento

Nestas regras identificamos os agentes envolvidos da seguinte forma:

- *a* - agente local ao receber uma mensagem de solicitação de monitoramento a ser encaminhada ao agente a ser monitorado;
- *b* - agente remoto, o qual necessita ser monitorado;

- c - agente para o encaminhamento da mensagem de monitoração, quando os agentes a e b não são adjacentes;
- d - agente que solicitou a monitoração, origem da mensagem de solicitação de monitoração;
- e - agente que transmitiu a mensagem para a , diferente do agente d .

O recebimento da mensagem de solicitação de monitoração é também utilizada pelo agente local para atualizar o estado de comunicação do agente que fez a solicitação. Se a mensagem recebida não foi enviada pelo agente (d), o agente (e) que transmitiu a mensagem também terá o seu estado de comunicação atualizado.

Regras para o encaminhamento de mensagem de resposta a solicitação de monitoramento

Para encaminhar uma mensagem de resposta à solicitação de monitoração ao agente solicitante, o agente local verifica, na sua própria visão do agente, qual o próximo veículo para encaminhar a mensagem de resposta. Caso este agente e o local não sejam adjacentes, a mensagem será encaminhada para um novo agente de encaminhamento. Estas regras são similares às regras para encaminhamento da mensagem de solicitação, apresentadas na Seção 4.2.2.

Nestas regras identificamos os agentes envolvidos da seguinte forma:

- a - agente local - recebeu a mensagem de resposta de monitoramento para ser encaminhada;
- b - agente remoto - destino da resposta ao monitoramento;
- c - novo agente de encaminhamento, caso a e b não sejam adjacentes;
- d - agente transmissor da mensagem, que enviou a mensagem para a , caso a e e não sejam adjacentes;

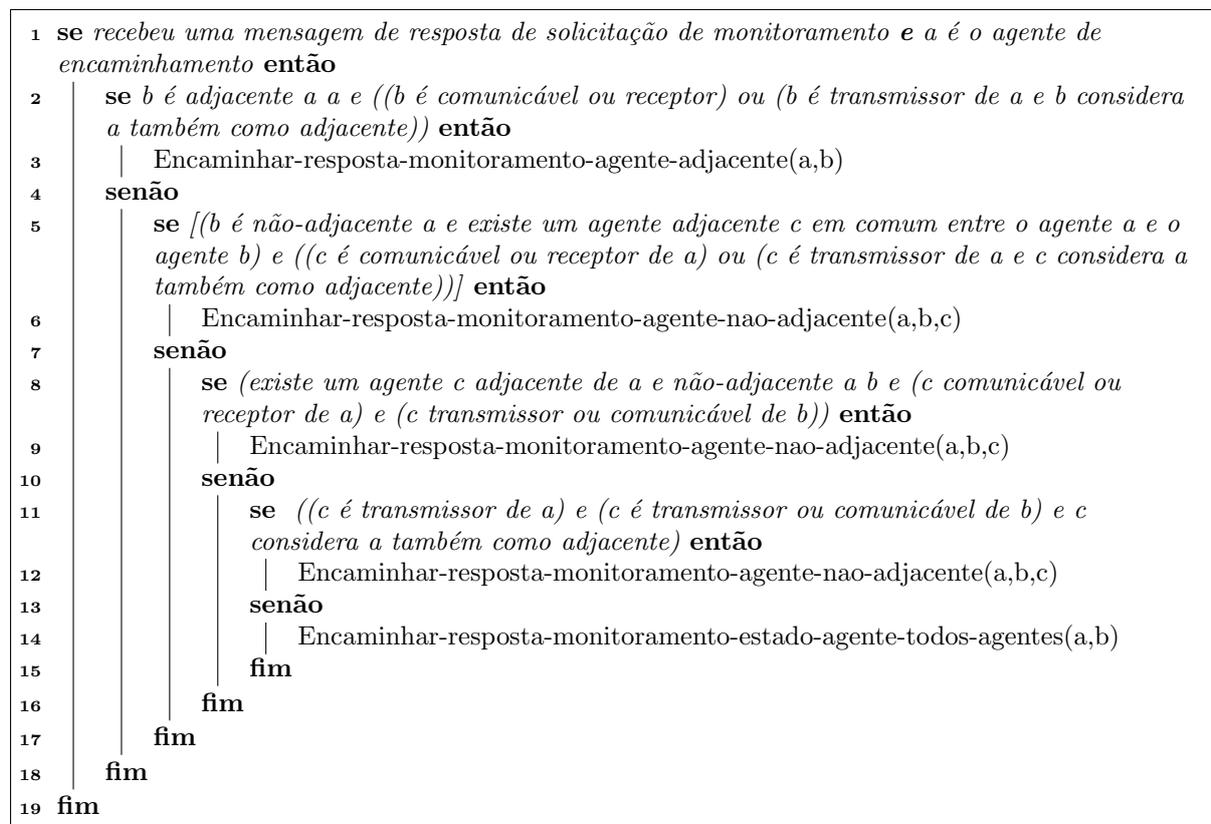


Figura 4.14. Regras para encaminhar resposta de solicitação de monitoramento

- e - agente monitorado, que respondeu à solicitação.

4.2.3 Serviço Geral de Recebimento de Mensagens

Este serviço atualiza os dados do estado do ambiente ao receber uma mensagem de um agente envolvido em uma transmissão. Para isso, ao receber uma mensagem, o mecanismo identifica os agentes de origem e de encaminhamento, verifica a situação de cada um deles em relação ao agente que recebeu a mensagem e atualiza os dados necessários.

A verificação do estado de um agente é baseada na definição dos estado do ambiente de comunicação descrito na seção 4.1.2. Desse modo foram criadas ações e regras que definem o comportamento do serviço proposto e serão discutidas nas próximas seções.

Ações do Serviço de Recebimento de Mensagens

As ações para o serviço de recebimento de mensagens do *I-CAR* foram desenvolvidas para manipular a estrutura de dados que armazena o estado do ambiente de comunicação. Esta manipulação consiste, principalmente, em alterações nos dados referentes ao estado do ambiente de comunicação, tais ações são:

- *Atribuir-Estado-Agente*[$agente, estado$] - atribui um estado indicado (transmissor, receptor, comunicável, incomunicável ou falho) a um agente especificado.
- *Atribuir-adjacência-timeout*(a, b) - atribui a condição de adjacência e calcula o tempo máximo de espera para a próxima atualização do estado do agente de origem. Esta atribuição é realizada quando os agentes envolvidos são adjacentes entre si.
- *Atribuir-adjacência-timeout-com-encaminhamento*(a, b, c) – similar à ação *Atribuir-adjacência-timeout*, é executada quando os agentes de origem e destino da mensagem não são adjacentes e são utilizados agentes de encaminhamento. O cálculo do

timeout para a próxima atualização do estado do agente de origem considera que este não é adjacente ao agente local.

- *Atualizar-visao-estado-do-agente-de-origem-da-mensagem* – consideramos que na mensagem recebida contém informações sobre o ambiente de comunicação visto pelo agente de origem, assim, esta ação identifica os dados do estado da comunicação do agente de origem que estão encapsulados na mensagem e atualiza no agente local.

As regras para disparar as ações de recebimento do estado do ambiente de comunicação serão discutidas a seguir.

Regras Para Definir o Estado do Ambiente Ao Receber Uma Mensagem

Com base na definição do estado da comunicação e do modelo de transição descritos na Seção 4.1.2, um conjunto de regras foi desenvolvido com o objetivo de modificar, quando necessário, o estado do ambiente de comunicação.

Em cada recebimento de mensagem, o agente verifica e atualiza o seu estado da comunicação em relação ao grupo e o estado dos agentes remotos associados à mensagem. Tais agentes são o que originou e o último que encaminhou a mensagem. Esta atualização define se os agentes remotos envolvidos estão comunicáveis, transmissores, receptores, incomunicáveis ou falhos e se são adjacentes ou não.

As regras para o recebimento de mensagens no *I-CAR* estão apresentadas na Figura 4.15 e possui os agentes envolvidos da seguinte forma:

- *a* - agente local
- *b* - agente destino
- *c* - o último agente que transmitiu a mensagem, este enviou a mensagem para *a*, caso *a* e *b* não sejam adjacentes.

- d - é o agente de origem, aquele que originou a mensagem.

```

1 se  $a$  recebeu uma mensagem de  $d$  recentemente e corretamente então
2   se o tempo de atualização do  $d$  está expirado ou  $d$  está incomunicável ou falho em relação a
    $a$  recentemente então
3     | Atribuir-Estado( $d$ , Transmissor)
4   fim
5 se  $d$  é receptor em relação a  $a$  recentemente então
6   | Atribuir-Estado( $d$ , Comunicável)
7   fim
8 se o tempo de atualização do  $c$  está expirado ou  $c$  está incomunicável ou falho em relação a
    $a$  recentemente então
9   | Atribuir-Estado( $c$ , Transmissor)
10  senão
11   | se  $c$  está receptor em relação a  $a$  recentemente então
12     | Atribuir-Estado( $c$ , Comunicável)
13   fim
14  fim
15 se  $c \neq d$  então
16   | Atribuir-adjacência-timeout( $a$ , $b$ )
17  senão
18   | Atribuir-adjacência-timeout-com-encaminhamento( $a$ , $b$ , $c$ )
19  fim
20 se  $timestamp_{mensagem} \geq timeStamp_{do estado de}$   $d$  então
21   | Atualize-matriz-estado( $d$ )
22  fim
23 se Existe solicitação de monitoramento para ( $d$ ) então
24   | Excluir-solicitação-da-lista-de-espera( $d$ )
25  fim
26 se Existe solicitação de monitoramento para ( $c$ ) então
27   | Excluir-solicitação-da-lista-de-espera( $c$ )
28  fim
29 fim

```

Figura 4.15. Regras de atribuição de estado dos agentes na recepção de mensagens

Cada mensagem recebida pelo agente contém informações sobre a visão do estado da comunicação do agente remoto que originou a mensagem. Tal visão consiste em uma matriz de estado contendo a relação do agente que originou a mensagem com cada agente do grupo. Esta matriz é atualizada em cada agente que recebe esta mensagem.

Nestas regras são verificados os estados dos agentes (c) e (d) e atualizados de acordo com a tabela de transição. Vale ressaltar que, a matriz de estado do agente (d) no agente (a) é atualizado somente quando o instante de tempo da mensagem é maior do que o último instante de tempo da última atualização.

Além disso, ao receber uma mensagem é verificada a existência de solicitações de monitoramento dos agentes envolvidos, caso tenha alguma, a mesma é excluída, já que o estado desses agentes foi atualizado.

Regras para Atualizar o Estado do Agente Local em Relação ao Grupo de Agentes

Estas regras foram criadas para atualizar o estado do agente local em relação ao grupo de agentes envolvidos na aplicação, como apresentado na Seção 4.1.2.

O agente local verifica o seu estado em relação ao grupo quando uma mensagem é recebida ou transmitida ou quando houve alterações no estado dos agentes remotos.

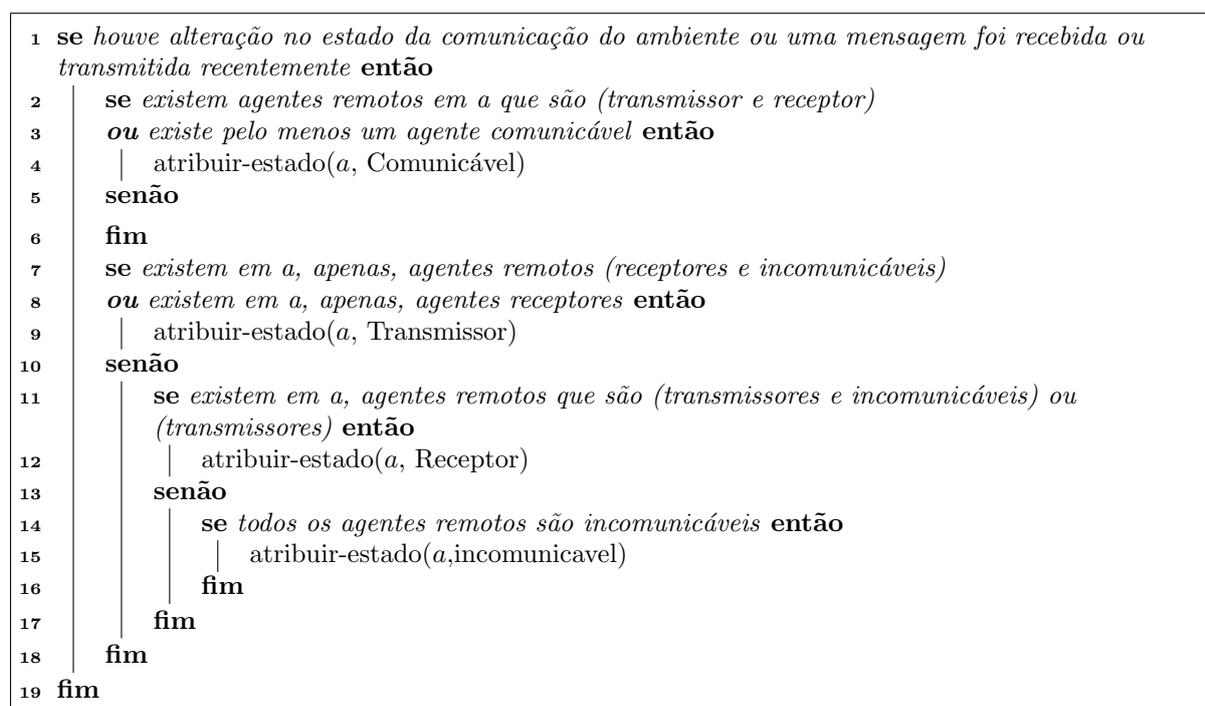


Figura 4.16. Regras de atribuição de estado do agente local

As alterações do estado para falho no agente local em relação ao grupo serão discutidas na seção de detecção de defeitos.

4.2.4 Serviço para a Comunicação um Agente para Todos - Broadcast

Este serviço foi desenvolvido para prover à aplicação veicular o envio de mensagens de um agente para muitos. Neste serviço, consideramos que a formação veicular é realizada por um grupo fechado e conhecido pelos seus membros. As principais ações utilizadas pelos serviços são:

- *enviar-mensagem-grupo*(*agente destino, agente encaminhamento, mensagem*) – encapsula informações do estado do agente local e transmite a mensagem solicitada pelo agente;
- *Verifica-agente-mais-distante* – esta ação busca o agente remoto (*b*) relacionado com o agente local (*a*) que possui maior quantidade de agentes para encaminhamento para transmitir uma mensagem de (*a*) para (*b*).

```

1 se a aplicação solicitou o serviço de envio de mensagens para o grupo então
2   | Verifica-agente-mais-distante
3   | se existe um agente remoto com maior quantidade de agentes de encaminhamento no grupo
4   |   | enviar-mensagem-grupo(agente destino, agente encaminhamento, mensagem)
5   |   | então
6   |   | atribui(Agente local, comunicavel)
7   |   | fim
8 fim

```

Figura 4.17. Regras para enviar mensagens de aplicações do agente para o grupo

A Figura 4.17 apresenta as regras para o serviço de envio de mensagens para um grupo de agentes, onde consideramos que um agente remoto (*b*) é o agente mais distante de (*a*) quando possui a maior quantidade de agentes de encaminhamento em relação aos outros membros do grupo na visão do estado de (*a*).

4.2.5 Serviço de Detecção de Defeitos

Em uma formação veicular onde os agentes são conhecidos eles devem se comunicar periodicamente, contudo, devido as características da rede é possível que alguns agentes

fiquem incomunicáveis por algum tempo, pois, podem acontecer interferências durante a comunicação, como por exemplo, a degradação do sinal de comunicação devido as condições das vias ou da mobilidade do veículo. Neste sentido, é possível que decisões sejam tomadas com o objetivo de diminuir o impacto da degradação do sinal de comunicação.

Contudo, em alguns casos, a comunicação realmente falhou e um conjunto de outras ações poderão ser executadas para, por exemplo, manter a comunicação mesmo na presença de falhas. Assim, para esta implementação, consideramos um agente remoto b com falha de comunicação por *crash* quando um agente, após, a verificação e atualização, fica incomunicável por um período de tempo estabelecido pelo usuário.

Além disso, consideramos um agente local (a) com falha de comunicação por *crash* quando todos os agentes remotos estiverem incomunicáveis em relação ao agente local (a) ou quando ocorre um erro local durante o recebimento ou transmissão de mensagens.

Desse modo, o serviço de detecção de defeitos é disparado pelo monitor, quando este não consegue receber resposta de uma solicitação de monitoramento de um agente monitorado em tempo hábil.

Consideramos, apenas, a definição de um agente remoto falho por meio de um tempo de espera de mensagem. Para o agente local, além do tempo de espera, consideramos a relação dele com todos os membros do grupo, não definindo quantidade de membros ou maioria, como citado na seção 4.1.2.

Assim, desenvolvemos a ação *calcularTimeoutDeteccao(agente)* que calcula o instante de tempo máximo para que o agente local (a) espere até considerar o agente remoto ou local como falho. Nesta implementação o instante de tempo máximo é calculado pela soma da quantidade de tempo máximo de espera definida pelo usuário, com o atraso médio da transferência da mensagem entre dois agentes e com o instante de tempo da detecção.

As regras descritas na Figura 4.18 descrevem o funcionamento da detecção de defeitos nos agentes remotos e locais. Estas regras estão baseadas na definição do estado falho de

```

1 se existe um agente (b) incomunicável e com o seu timeout de monitoramento expirado então
2   |   calcularTimeoutDetecção(b)
3   |   se timeoutDetecao e b ainda está incomunicável para a então
4   |     |   atribuir-estado(b, falho)
5   |   fim
6 fim
7 se a em relação ao grupo está incomunicável e todos os agentes remotos relacionados a a estão
   incomunicáveis então
8   |   calcularTimeoutDetecção(a)
9   |   se timeoutDetecao então
10  |     |   atribuir-estado(a, falho)
11  |   fim
12 fim

```

Figura 4.18. Regra para detecção de Falhas do agente

um agente local e de um remoto apresentadas na seção 4.1.2 e 4.1.2, respectivamente.

Vale ressaltar que, cada agente possui sua própria observação sobre quais agentes estão falhos. Além disso, se o agente considerado falho retornar a comunicação, seu estado muda de acordo com as regras para atualizar o estado do ambiente descritas em 4.2.3.

4.3 CONSIDERAÇÕES FINAS

Neste capítulo apresentamos uma abordagem para o desenvolvimento de serviços de comunicação para aplicações veiculares. Esta propõe o uso de regras sobre as informações do ambiente de comunicação para realizar a adaptação de serviços em relação a dinâmica e a heterogeneidade do ambiente de comunicação, ao mesmo tempo que busca atender aos requisitos e as demandas de uma aplicação veicular.

São vários os desafios em relação a essa abordagem, um deles é construir um serviço de comunicação considerando aspectos de desempenho relacionados ao fluxo de execução das regras. Além disso, determinar os atributos do ambiente para a definição da visão do estado do ambiente de comunicação.

Nas próximas seções será apresentado o *I-CAR* que é um conjunto de serviços para a formação veicular distribuída baseado na abordagem proposta.

CAPÍTULO 5

IMPLEMENTAÇÃO, SIMULAÇÃO E RESULTADOS

Neste capítulo descrevemos a implementação, simulação e a avaliação do mecanismo de comunicação proposto. Na Seção 5.1 apresentaremos as soluções para a implementação dos estados, regras e serviços do I-CAR. Na Seção 5.2, discutiremos sobre a simulação e as ferramentas adotadas para executar os experimentos. Já na Seção 5.3 apresentaremos os dados coletados durante a simulação e os resultados encontrados.

5.1 IMPLEMENTAÇÃO DO I-CAR

Metodologias modernas para análise e projetos de softwares orientados a agente considera o uso de técnicas de orientação a objetos como uma forma de desenvolvimento destes sistemas [43]. Assim, para a implementação do *I-CAR* adotamos tecnologias de desenvolvimento orientado a objetos, como a linguagem de programação *C++* e a *UML* (*Unified Modeling Language*) para modelagem do sistema.

A seguir apresentaremos o diagrama de classes do *I-CAR* com as principais soluções desenvolvidas na implementação dos serviços.

5.1.1 Implementação dos Serviços do I-CAR

A classe *ICAR*, mostrada na Figura 5.1, é a *container* do sistema e pode ter nenhum ou muitos serviços de comunicação. Os serviços de comunicação a serem providos pelo *I-CAR*, devem herdar as características da classe *CommService* que é uma classe abstrata que representa um serviço de comunicação do *I-CAR* e assegura a interação de qualquer serviço no mecanismo.

CommService é uma classe que possui pelo menos um comportamento associado a

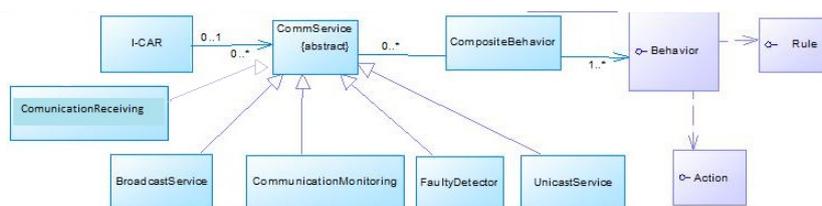


Figura 5.1. Diagrama de classes dos Serviços do I-CAR

ela, o qual é representado pela classe *CompositeBehavior* que é responsável por agregar, combinar e compor o(s) comportamento(s) associado(s) a ele. Tais comportamentos são representados pela interface *Behavior* a qual permite combinar regras e ações para formar o comportamento, como discutido na Seção 5.1.

As classes *Broadcast*, *CommunicationReceiving*, *CommunicationMonitoring*, *FaultyDetector* e *Unicast* representam, respectivamente, os serviços de *broadcast*, recebimento, monitoramento, detector de falhas e *unicast* descritos na Seção 4.2.

5.1.2 Implementação da Visão do Estado da Comunicação dos Veículos

O diagrama de classe apresentado na Figura 5.2 corresponde a estrutura de dados que suporta as informações sobre o estado da comunicação dos agentes remotos e do próprio agente local como discutido na Seção 4.1.2.

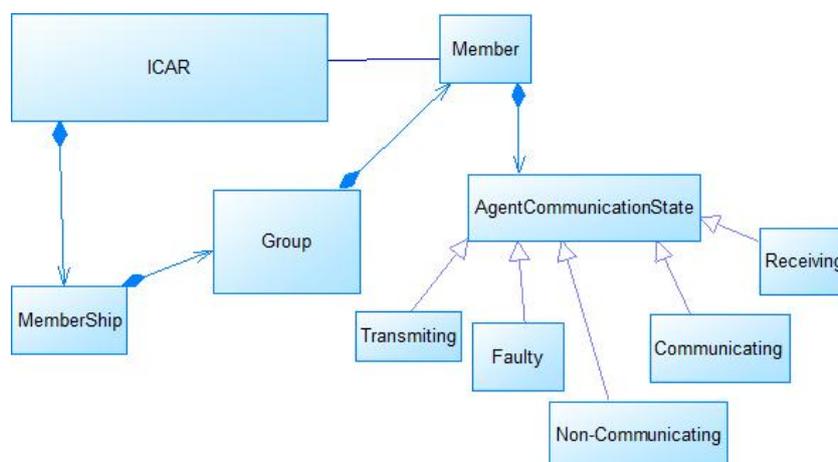


Figura 5.2. Diagrama que representa a estrutura de dados do estado da comunicação

Uma das classes que fazem parte desta estrutura é a *Membership* que pode ter nenhum ou vários grupos de veículos associados a ela e é responsável por gerenciar os grupos de agentes dos sistemas. A classe *Group*, por sua vez, representa um grupo de veículos, o qual na nossa aplicação corresponde ao grupo dos agentes envolvidos na formação veicular e possui pelo menos um *Member*.

A classe *Member* representa um membro (um veículo) do grupo do sistema e possui atributos como o *AgentCommunicationState* que representa o estado da comunicação do veículo. O objeto *Member* associado à classe *ICAR* representa o estado do agente local e o objeto associado à classe *Group* corresponde ao agente remoto do sistema.

Além do *AgentCommunicationState*, um conjunto de outros atributos são definidos na classe *Member*, dentre eles destacamos aqueles que contribuem para a atualização das informações do estado da comunicação do veículo, tais são:

- **Atraso estimado da mensagem (*CalculatedDelay*)** – é calculado pelo estimador indicado pelo usuário.
- **Atraso medido da mensagem (*MeasuredDelay*)** – é a diferença entre o instante de tempo da chegada da mensagem no agente destino e o instante de tempo da transmissão da mensagem no agente origem.
- **O instante de tempo máximo da validade do estado de um agente remoto ($TIMEOUT_{dve}$)** – este tempo máximo indica até quando a informação do estado de um agente remoto é válida para o agente local associado e é calculado por $TIMEOUT_{dve} = T + DVE + calculatedDelay$, onde T é o instante de tempo do cálculo.
- **O instante de tempo máximo de espera da resposta de monitoramento ($TIMEOUT_{tem}$)** – este tempo máximo indica até quando o monitor deve esperar a resposta sobre o estado de um agente remoto monitorado e é calculado por $TIMEOUT_{tem} = T + TEM + calculatedDelay$, onde T é o instante de tempo do cálculo.

- **O instante de tempo máximo de espera de identificação de falhas ($TIMEOUT_{ted}$)**
 - este tempo máximo indica até quando o detector de defeitos deve esperar uma resposta de um agente monitorado e é calculado por $TIMEOUT_{ted} = T + TED + calculatedDelay$, onde T é o instante de tempo do cálculo.

5.1.3 Implementação dos Tipos de Mensagens

As mensagens trocadas entre os veículos possuem um formato próprio para que os serviços de monitoramento e de detecção de defeitos possam manter os estados do ambiente atualizados. Criamos três tipos de mensagens, tais são: o tipo 0 indica que a mensagem possui informações da aplicação veicular; o tipo 1 é a mensagem de solicitação de monitoramento; o tipo 2 é a resposta da solicitação do monitoramento.

O tipo de mensagem 0 pode ser utilizado pela aplicação e/ou pelos serviços disponíveis no próprio mecanismo, a exemplo dos serviços de *unicast* e *broadcast*. Ao transmitir a mensagem do tipo 0 os seguintes dados são encapsulados: o instante de tempo que a mensagem foi transmitida; o identificador da mensagem; o identificador do agente de origem; o identificador do agente anterior; o identificador do próximo agente; o identificador do agente destino; a matriz do agente de origem contendo a identificação de cada agente remoto do grupo e o respectivo estado da comunicação e a condição de vizinhança em relação ao agente da origem da mensagem.

A mensagem do tipo 1 é utilizada pelo monitor, não pode ser utilizada por qualquer outro serviço embarcado no mecanismo e corresponde a solicitação de monitoramento de um agente. Ao transmitir a mensagem do tipo 1 os seguintes dados são encapsulados: o instante de tempo que a mensagem foi transmitida; o identificador da mensagem; o identificador do agente de origem; o identificador do agente solicitante que requisita o estado do agente monitorado; o identificador do próximo agente que irá encaminhar a mensagem; o identificado do agente monitorado; a matriz do agente solicitante com a identificação de cada agente remoto do grupo e o respectivo estado da comunicação e a condição de vizinhança em relação ao agente solicitante.

O tipo de mensagem 2 não pode ser utilizada pela aplicação e nem pelos serviços disponíveis no mecanismo e ao transmitir mensagens deste tipo os seguintes dados são encapsulados: o instante de tempo que a mensagem foi transmitida; o identificador da mensagem; o identificador do agente monitorado; o identificador do agente solicitante; o identificador do próximo agente que irá encaminhar a mensagem; o identificador do agente que encaminhou a mensagem anteriormente; a matriz do agente monitorado com a identificação de cada agente remoto do grupo e o respectivo estado da comunicação e a condição de vizinhança em relação ao agente monitorado.

5.2 SIMULAÇÃO

A mobilidade dos veículos na formação veicular distribuída pode alterar a forma de comunicação entre eles [26, 7]. Assim, para analisar o comportamento do *I-CAR* provendo os serviços de comunicação sobre o efeito da mobilidade na rede, adotamos o *framework Veins* (*Vehicles in Network Simulation*).

O *Veins* acopla de forma bidirecional as técnicas relacionadas às simulações de rede de comunicação e as técnicas do domínio da microsimulação¹ de tráfego em rodovias [34].

O acoplamento provido pelo framework *Veins* permite integrar o *OMNeT++* (*Objective Modular Network Testbed in C++*) para simulação do ambiente de comunicação e o *SUMO* (*Simulation of Urban Mobility*) para a simulação do tráfego dos veículos nas vias [22][39]. O *VEINS* utiliza a *interface TraCI* (*Traffic Control Interface*) para realizar a troca de informações entre os dois simuladores em tempo de execução [40].

¹Microsimulação de tráfego de vias consiste em um modelo de mobilidade considerado pelos nós das *VANETs*, pois, tal modelo consegue prover informações mais precisas sobre cada veículo e o relacionamento entre eles, por exemplo a posição exata dos veículos simulados [34].

5.2.1 Configuração da Mobilidade no SUMO

Dado um conjunto de veículos conhecidos entre si que precisam percorrer enfileirados uma trajetória pré-definida, adotamos o algoritmo de mobilidade provido pelo *Framework VEINS*, o qual, mantém, para cada veículo, os parâmetros necessários (e.g. velocidade e aceleração) para evitar colisões entre os veículos.

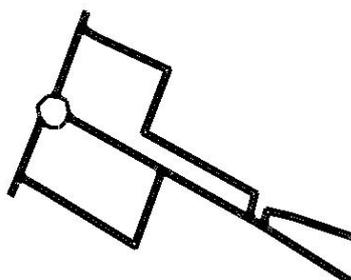


Figura 5.3. Trajetória Curvas Acentuadas

Assim, para um grupo de veículos em movimento, utilizamos do *VEINS* uma trajetória com rotatórias, retornos e curvas mais acentuadas que representa as ruas da cidade *Erlange* na Alemanha, extraídas do mapa disponível pelo projeto *OpenStreetMap*, como mostra a Figura 5.3. Nesta trajetória os veículos trafegam por 300s e a velocidade máxima adotada em cada via não ultrapassa os 14m/s (50km/h). A Figura 5.4 mostra um grupo formado por 4 veículos percorrendo parte da trajetória no *SUMO*.



Figura 5.4. Amostra de Parte da Trajetória Implementada no *SUMO*

Neste ambiente, a variação das distâncias entre os veículos acontece devido à necessidade de diminuir a velocidade de um veículo quando este passa por uma conexão entre vias (curvas, transversais e rotatórias). Ao modificar a velocidade de um veículo, os demais devem, quando necessário, alterar seus parâmetros de mobilidade (e.g. aceleração) para evitar colisões entre si. Esta variação da distância entre veículos pode interferir no modo de comunicação entre eles.

5.2.2 Configuração da Comunicação no OMNET

Na configuração do ambiente de comunicação utilizamos o módulo *Car* do *framework VEINS*, o qual, representa a abstração de um agente veicular implementado no *OMNET++*. Este possui um conjunto com os submódulos *UDP*, *NetworkLayer* e *WLAN* que definem os elementos da rede veicular utilizados por um veículo.

No módulo *Car* criamos um submódulo *VehicularFormationGComm* no qual encapsulamos toda a implementação do mecanismo de comunicação *I-CAR*. Este módulo possui uma interface para a interação com o submódulo *UDP* o qual possui a implementação do protocolo *UDP (User Datagram Protocol)* para a transmissão de dados fim-a-fim, como mostra a Figura 5.5.

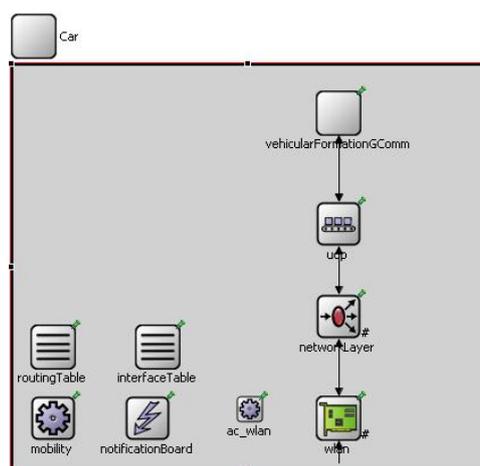


Figura 5.5. Módulo *VehicularFormationGComm* embarcado no módulo *Car* do *VEINS*

O módulo *NetworkLayer* possui interface com o módulo *UDP* e *WLAN* e a implementação do protocolo *IP (Internet Protocol)*. Nesta configuração não adotamos qualquer algoritmo de roteamento, deixamos a cargo dos serviços providos pelo módulo *VehicularFormationGComm*. Por isso, os submódulo *routingtable* e *interfacetable* não estão sendo utilizados.

O *wLan*, apresentado na Figura 5.5, possui a configuração da camada física e de enlace, a qual define os parâmetros que implementam o protocolo *IEEE 802.11b*, com taxa de transmissão de 11Mbit/s.

No cenário no *OMNET*, cada módulo *Car* representa um carro no *SUMO* e possui o módulo *mobility* que conecta a *Interface Traci* e recebe informações sobre a mobilidade dos veículos do *SUMO*, assim o *OMNET* adapta a posição de cada nó da rede em seu ambiente de simulação de acordo com a mobilidade do *SUMO*.

Vale ressaltar que, nesta seção descrevemos os aspectos da configuração da rede que achamos mais relevantes para o entendimento do ambiente de simulação, mas caso deseje mais detalhes do módulo *Car* pode ser encontrado em [34].

5.2.3 Parâmetros para Simulação do I-CAR

Além de configurar a mobilidade e a rede veicular e de encapsular o *I-CAR* no módulo *Car* do *Veins*, para executar a simulação, é necessário prover informações para os parâmetros de entrada de cada experimento a ser realizados, tais parâmetros são:

Número de Veículos (*NVEIC*)

Definimos dois conjuntos de veículos para a simulação, um com 4 e outro com 10 veículos, baseando-se no requisito que indica que o número máximo de veículos em uma formação veicular é 10 (ver Seção 3.3.1.3). Escolhemos 4 veículos de forma empírica, considerando ser um número pequeno de veículos.

Período da Carga da Aplicação (*AppLoad*)

Este parâmetro representa a carga de mensagens geradas pela aplicação e nele é definido o período em que cada veículo envia informações, sobre o movimento, para um ou mais agentes envolvidos na formação veicular.

De forma empírica, assumimos a frequência de transmissão de mensagens de cada veículo em formação, variando entre 0.5s, 1s, 1.5s ou 2s. Apenas consideramos como limite atender a um dos requisitos temporais da comunicação da formação veicular discutidos em [42, 21], o qual, determina que o tempo máximo de resposta da rede suportado pelo controle do movimento veicular é de 100ms (ver Seção 5.1.2).

Assim, consideramos a maior (com a frequência de 0.5s) carga da aplicação de: 8 mensagens por segundo para o conjunto com 4; veículos e 20 mensagens por segundo para um conjunto com 10 veículos.

Duração da Validade do Estado (*DVE*)

Este parâmetro é utilizado pelo monitor do mecanismo e define a duração da validade do estado de um agente remoto relacionado ao agente local. Após, este tempo a informação sobre o estado é considerada como desatualizada. Para cada agente remoto pode ser definida uma duração da validade do estado específico.

Definimos de forma empírica tomando como valor de referência o trabalho apresentado em [32], o qual determina o período para a próxima atualização do estado do ambiente entre [1.75, 2.25] segundos para uma aplicação que prover transmissão de dados de vídeo. Assim definimos o *DVE* variando entre 0.5s, 1s, 1.5s ou 2s.

Tempo de espera do Monitoramento (*TEM*)

Este parâmetro é utilizado pelo monitor do mecanismo e define a quantidade de tempo que um agente local espera por uma resposta de solicitação de monitoramento de um agente remoto. Nestas simulações consideramos a variação do TEM igual ao DVE.

Tempo de Espera da Detecção de Defeitos (*TED*)

Este parâmetro é utilizado pelo detector de defeitos e define o tempo que o agente espera para identificar o agente falho. Nestas simulações consideramos a variação do TED igual ao DVE.

Estimador de Atraso (*EA*)

Este parâmetro é utilizado pelo monitor e o detector de defeitos do mecanismo e define o algoritmo que será usado para estimar o atraso na entrega de mensagens entre dois agentes.

O atraso estimado é utilizado para calcular os tempos máximos do sistema, como por exemplo, o tempo de espera por uma resposta de solicitação de monitoramento.

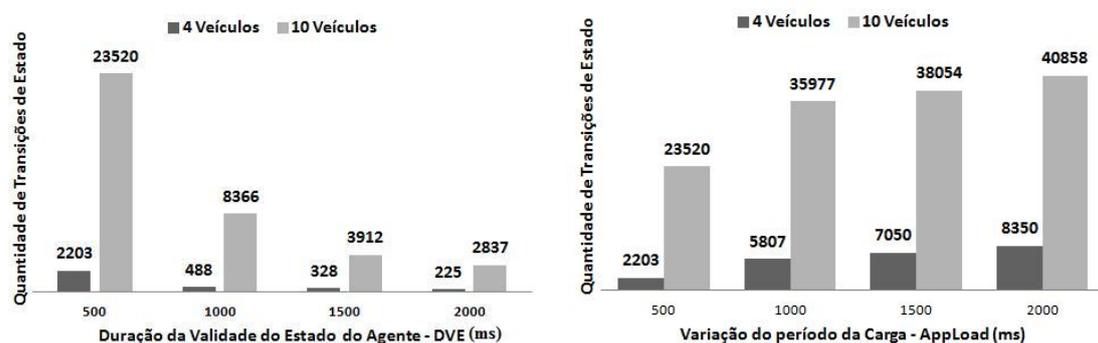
5.3 ANÁLISE DO FUNCIONAMENTO DO I-CAR

Nesta seção, apresentamos os resultados encontrados a partir da análise do efeito da mobilidade dos veículos em relação ao I-CAR. Para tanto, durante a simulação, coletamos informações sobre: a transição dos estados dos agentes remotos em um agente local; os atrasos medidos e estimados das mensagens fim-a-fim; a quantidade de mensagens da aplicação e de monitoramento que foram transmitidas, recebidas e encaminhadas; informações sobre a detecção de defeitos entre dois agentes e entre um agente e o grupo.

A partir de tais informações, analisamos o comportamento da transição dos estados, do atraso da comunicação fim-a-fim, a sobrecarga (*overhead*) das mensagens de monitoramento, a perda das mensagens, detecção de defeitos e de falsa suspeitas.

5.3.1 Transição entre os Estados

Verificamos o comportamento da transição entre os estados de um agente remoto quando variamos o tempo da validade do estado (*DVE*) e o período de requisição de transmissão de mensagem da aplicação (*AppLoad*), como mostram os gráficos das Figuras 5.6(a) e 5.6(b), respectivamente.



(a) Quantidade de transições realizadas variando o DVE (b) Quantidade de transições realizadas variando a carga da aplicação

Figura 5.6. Transição entre os estados de um agente remoto

O gráfico da Figura 5.6(a) apresenta a quantidade de transições geradas em relação a

variação do *DVE* com a frequência da transmissão das mensagens da aplicação de 0,5s por veículo. Foi aplicada a carga de mensagens da aplicação corresponde à maior utilizada entre os experimentos realizados. Quando a frequência da carga da aplicação é menor ou igual ao *DVE* permite que a mensagem da aplicação que encapsula o estado do agente de origem atualize o estado do agente remoto sem acionar o monitoramento.

A Figura 5.6(a) mostra que quanto maior o *DVE* do agente remoto, menor é a quantidade de transições realizadas, já que, com *DVE* maior o estado demora de perder a validade, diminuindo a quantidade de verificações e atualizações dos estados ao percorrer trajetória.

O gráfico da Figura 5.6(b) apresenta a quantidade de transições geradas em relação a frequência da transmissão das mensagens da aplicação representada pelo período do gráfico. Neste gráfico, apresentamos os resultados encontrados nos experimentos que possuíam a menor (0,5s) duração da validade do estado (*DVE*). Assim, quando diminuimos a carga da aplicação, aumentando a frequência de solicitação de transmissão de mensagens para uma mensagem a cada 1s, 1,5s e 2s, percebemos que a quantidade de transições aumentou, pois, a demora em receber informações da aplicação provoca o acionamento do serviço de monitoramento.

5.3.2 Atraso das Mensagens Fim-a-Fim

Avaliar o atraso das mensagens consiste em verificar em quanto tempo uma mensagem leva para partir da origem e chegar ao seu destino. Para isso, coletamos em cada recepção de mensagem fim-a-fim os atrasos medidos e estimados calculados pelo mecanismo (como descrito na seção 5.1.2). Após a coleta calculamos a média e o desvio padrão dos dois atrasos. A Tabela 5.1 mostra a média e o desvio padrão dos atrasos medidos e estimados de todos os experimentos com 4 e 10 veículos e que possui a maior carga da aplicação (período de 0,5s em cada veículo) assumida na simulação.

Os valores do desvio padrão medidos são característicos da rede sem fio, por conta de

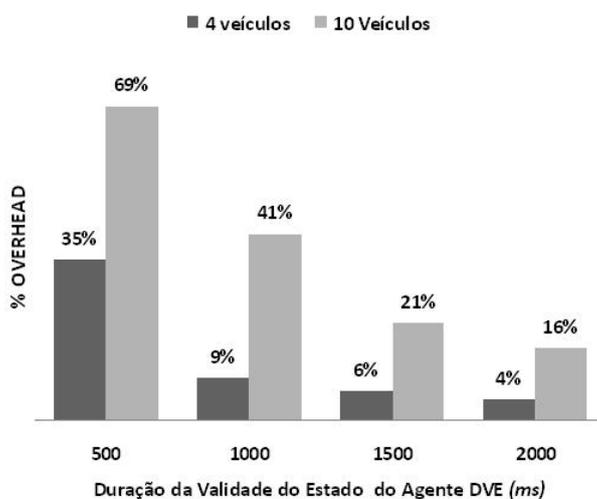
Tabela 5.1. Atraso Médio Fim-a-Fim

Atraso Fim-a-Fim (ms)					
nvec	DVE	Medido		Estimado	
		média	desvio	média	desvio
4	2000	0,98	0,57	0,93	0,42
4	1000	1,02	0,70	0,96	0,41
4	1500	1,06	0,51	0,89	0,30
4	500	1,23	0,86	1,07	0,31
10	1000	1,80	1,64	1,70	0,67
10	1500	1,84	1,65	1,73	0,59
10	500	2,02	1,89	1,97	0,88
10	2000	2,03	1,58	1,90	0,57

colisões de mensagens, latência de acesso ao meio, entre outros. Desse modo, observamos que tanto o atraso medido quanto o estimado atenderam ao requisito de comunicação da aplicação adotado como base para a simulação (ver seção 5.2.1).

5.3.3 Overhead de Mensagens de Monitoramento

A sobrecarga de mensagens de monitoração (*overhead*) da Figura 5.7 é calculada a partir da relação entre o total de mensagens de monitoração transmitidas (n_{mon}) e a quantidade total de mensagens transmitidas na rede (n_{tot}), isto é: $overhead = n_{mon}/n_{tot}$. Para este gráfico utilizamos os experimentos com maior carga de mensagens da aplicação (frequência de 0,5s por veículo) na rede.

**Figura 5.7.** *Overhead* de mensagens de monitoramento

Assim, observamos no gráfico da Figura 5.7 que a sobrecarga aumenta quando o número de veículos também aumenta, pois, para cada veículo serão, quando necessário, enviadas solicitações de monitoramento. Por outro lado, quando a duração da validade do estado (*DVE*) de um agente remoto aumenta, o número de mensagens transmitidas na rede reduz, diminuindo também o *overhead* de mensagens de monitoramento.

5.3.4 Mensagens Perdidas

O gráfico apresentado na Figura 5.8 mostra o percentual de perda de mensagens de cada veículo envolvido na simulação com e sem usar o *I-CAR*. Para usar a aplicação sem o *I-CAR* criamos um serviço de transmissão e recepção para as mensagens geradas pela formação veicular distribuída. Estes serviços não utilizam regras para verificar o estado do ambiente e não utilizam qualquer estratégias de encaminhamento de mensagens.

Os resultados apresentados correspondem aos experimentos que possuem a maior carga da aplicação, a menor duração da validade do estado e o maior número de veículos. Deste modo, a carga da rede é a maior entre todos os experimentos realizados.

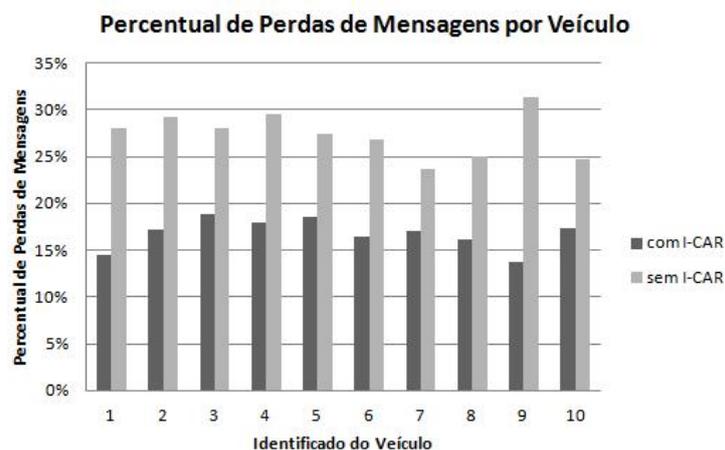
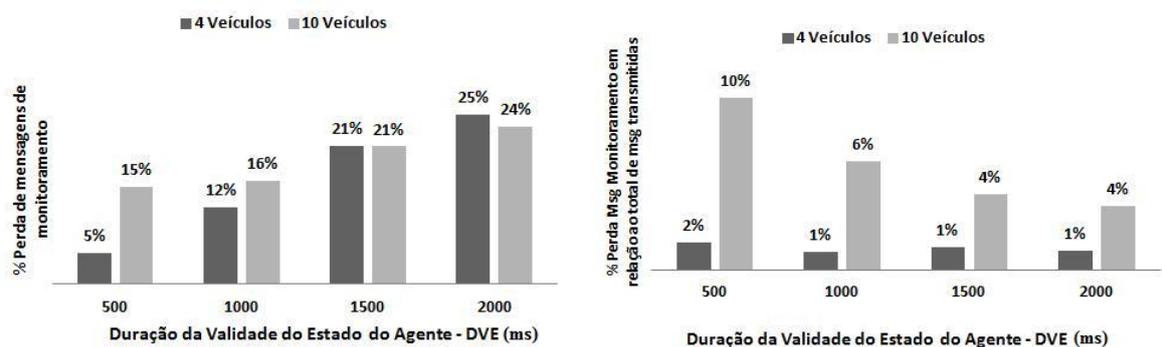


Figura 5.8. Mensagens Perdidas

Neste verificamos que as perdas de mensagem sem utilizar o *I-CAR* chegam ao máximo de 31% e ao utilizar o *I-CAR* de 19%. A diferença média da perda de mensagens entre sem e com o I-CAR é de 10%.

5.3.5 Mensagens de Monitoração Perdidas

A Figura 5.9 mostra que ao diminuir a quantidade de solicitações de monitoramento, aumentando o *DVE*, o percentual de perdas aumentou. Isso acontece, porque, diminuir a quantidade de mensagens na rede não garante que as perdas de mensagens diminuam. Neste caso, as perdas aconteceram devido à dinâmica do ambiente.



- (a) % Perda de msg de monitoramento em Relação ao total de msg de monitoramento transmitidas
- (b) % Perda de mensagens de monitoramento em Relação ao total de mensagens transmitidas

Figura 5.9. Mensagens de Monitoração Perdidas

As Figuras 5.9(a) e 5.9(b) mostram os gráficos que representam as perdas de mensagens de monitoração em relação ao total de mensagens de monitoramento transmitidas e ao total de todas as mensagens transmitidas, respectivamente.

5.3.6 Detecção de Defeitos

Verificamos o funcionamento do serviço de detecção de defeitos em relação à identificação de falsas suspeitas de falhas de um agente. Uma falsa suspeita de falha ocorre quando um agente local identifica que um agente remoto está falho por *crash* quando na verdade não está.

O gráfico 5.10 mostra o percentual de identificação de falsas suspeitas de falhas em relação ao total de transições de estado realizadas, variando o *DVE*. Neste gráfico observamos que quanto menor é a duração da validade do estado (*DVE*), menor é a quantidade

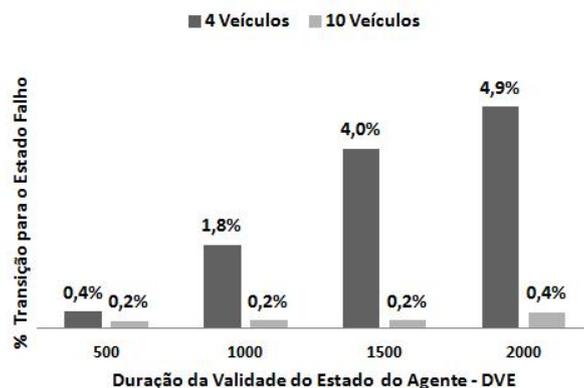


Figura 5.10. Percentual de falsas suspeitas de falhas

de falsas suspeitas devido a frequência com que o estado do ambiente é atualizado.

Em uma outra simulação inserimos uma falha por *crash* em um dos veículos e todos conseguiram identificar o agente como falho obedecendo ao $TIMEOUT_{tde}$ para mudar o estado para falho.

5.4 CONSIDERAÇÕES FINAIS

Neste capítulo, discutimos sobre a implementação, simulação e resultados obtidos sobre o *I-CAR*. Utilizamos tecnologias modernas de análise e projeto de software orientado a agentes para desenvolver todo o sistema. Adotamos ferramentas de simulação que aproximassem a simulação da realidade, contemplando os cenários de vias e de comunicação utilizando um *Framework* que interage com duas ferramentas de simulação muito utilizadas na literatura.

Avaliamos a interferência do movimento em relação ao *I-CAR* e observamos que o monitor e o detector de defeitos atualizam a visão do estado do ambiente de comunicação em cada agente envolvido na formação no tempo previsto pela simulação, aproveitando as mensagens que trafegam no ambiente de comunicação. Utilizando a estratégia de transmissão de mensagens baseada em regras conseguimos diminuir as perdas de mensagens do sistema.

Apesar do I-CAR ainda não prover garantias temporais rígidas, os resultados encontrados na simulação nos motivam a continuar buscando melhores soluções aprimorando a abordagem proposta neste trabalho.

CONCLUSÃO

Nas redes veiculares, os dispositivos embarcados em veículos se movimentam, podendo entrar e sair do alcance de comunicação uns dos outros, caracterizando um ambiente dinâmico, no qual as capacidades de comunicação se alteram a todo instante.

Sobre estas redes aplicações têm sido propostas e desenvolvidas sendo que algumas tratam do fornecimento de informações para uso do condutor, e outras visam garantir o controle autônomo, cooperativo, dos veículos, com objetivos específicos. Formação veicular é uma destas aplicações, na qual um grupo de veículos troca informações e tomam decisões com relação a sua movimentação, com o objetivo de gerar e manter uma formação desejada durante a trajetória.

Para a adequada solução do problema da formação veicular os veículos precisam trocar informações, com garantias de que as informações necessárias, estarão disponíveis para todos os veículos do grupo, no momento necessário.

Neste sentido, desenvolvemos uma abordagem para a construção de serviços de comunicação baseado em regras, que foi proposta para ser aplicada em sistemas multiagentes como a formação veicular distribuída.

Para isso, foi desenvolvida uma estrutura que permite que o projetista modele o serviço de comunicação com base em uma visão de estado da comunicação, um conjunto de ações e regras que definem o comportamento de um serviço. Esta abordagem provê uma percepção da estado da comunicação utilizando atributos do próprio ambiente, o que também é feito em vários serviços de comunicação, como foi apresentado em 2.5.

Além disso, a abordagem proposta busca permitir que o projetista trate a heterogeneidade e a dinâmica do ambiente de comunicação possibilitando, em fase de projeto, utilizar atributos variados e regras diferenciadas entre os agentes, com base nas características

individuais de cada agente, mesmo provendo o mesmo serviço de comunicação.

Assim, utilizando esta abordagem criamos o *I-CAR*, que é conjunto de serviços de comunicação veicular para atender aos requisitos de comunicação da formação veicular distribuída, neste conjunto contém os serviços: comunicação *unicast*; comunicação *broadcast*; o monitor de estados de comunicação; e para os serviços de detecção de defeitos.

A construção de um monitor de estados de comunicação baseado em regras foi fundamental para a manutenção da visão do agente em relação ao estado do ambiente de comunicação. Este aproveita as mensagens das aplicações que utilizam os serviços do mecanismo para otimizar a atualização do estado do ambiente. O monitor de estados de comunicação é utilizado pelos demais serviços de comunicação, e também pode ser utilizado pela aplicação.

A opção por construir estes serviços baseados em regras, é fundamental para que as decisões sejam tomadas levando em consideração diferentes aspectos da comunicação, como: o estado de comunicação conhecido entre os diversos agentes; aspectos sobre o fornecimento do serviço proposto; aspectos sobre as mensagens que são trocadas entre os diversos agentes.

Um aspecto em especial, é a adoção de uma estratégia de encaminhamento de mensagens baseada em regras efetuado no nível da aplicação. No *I-CAR* o roteamento é feito sob demanda, mas sem manter tabelas de roteamento e sem restrições no número de saltos. Além disto, o fato de o roteamento ser feito em sobreposição, permite que cada agente intermediário utilize as mensagens encaminhadas na atualização de sua visão do estado de comunicação dos agentes comunicantes. Estes aspectos dos serviços oferecidos pelo *I-CAR* são fortes argumentos justificando as soluções oferecidas.

Desenvolvemos uma implementação dos serviços propostos, obtendo diversos resultados de experimentos realizados. Os cenários para os experimentos realizados foram desenvolvidos utilizando um ambiente de simulação (*Framework VEINS*) que contemplou tanto o cenário das vias quanto o da comunicação, este foi essencial para analisar o comportamento do mecanismo sob o efeito da mobilidade do ambiente, juntamente com

a carga gerada pela aplicação da formação veicular.

As mensagens trocadas pelo *I-CAR* geram um *overhead* que não interfere no atendimento aos requisitos de comunicação da formação veicular previstos pelos serviços. Além disto, apesar da taxa de perda de mensagens de 19% em relação ao total de mensagens geradas, com o algoritmo de mobilidade utilizado, não observamos alterações na movimentação dos veículos.

Neste trabalho ainda não verificamos o comportamento do funcionamento do *I-CAR* em relação em outros tipos de aplicação, esta análise faz parte dos trabalhos futuros, os quais iremos discutir na próxima seção.

6.1 TRABALHOS FUTUROS

A partir dos resultados encontrados, surgem outros questionamentos em relação a abordagem proposta gerando novos trabalhos a serem desenvolvidos, tais como:

- Investigar como melhorar a confiabilidade dos serviços de comunicação *unicast* e *broadcast*;
- Desenvolver um serviço de comunicação *multicast*;
- Desenvolver um serviço de comunicação em grupo (*membership* mais o *multicast*);
- Um outro trabalho é construir novos cenários ainda mais parecidos com uma zona urbana e zona de auto-estrada, variando a quantidade de veículos, inserindo interferências nos sinais de comunicação, alterando o tipo de vias, colocando veículos que não fazem parte da formação, entre outros. Assim, teremos uma avaliação do comportamento do mecanismo mais aproximada da dinâmica do ambiente de execução real.
- A implementação de um ambiente de mobilidade que permita verificar o desempenho de um controlador do movimento individual e coletivo de uma aplicação

de formação veicular distribuída a exemplo [23, 10]. Desse modo, poderemos avaliar o impacto do mecanismo sobre o funcionamento de um sistema de controle do movimento.

- Desenvolver um adaptador dinâmico que calcule em tempo de execução a duração da validade do estado de um agente remoto (*DVE*). Este cálculo poderá ser feito a partir de um conjunto de parâmetros de qualidade de serviço da aplicação, como é realizado em [31], associado ao estado do ambiente de comunicação do mecanismo em tempo de execução. Isso permitirá que o agente utilize os recursos de comunicação de forma inteligente, permitindo a redução de atrasos e perdas de mensagens sem deixar de atender as expectativas da aplicação.
- Comparar o desempenho das regras para a transmissão de mensagens com os protocolos de encaminhamento já existentes para *VANETs*, como discutido na Seção 2.4.2, identificando as vantagens e desvantagens.
- Comparar, também, os serviços de *unicast* e *broadcast* propostos com outras abordagens utilizadas em aplicações veiculares como [2], entre outros. Desse modo, avaliar o comportamento dos serviços em um outro tipo de aplicação veicular trazendo vantagens e desvantagens.

6.2 CONSIDERAÇÕES FINAIS

Este trabalho propõe o tratamento da dinâmica da comunicação na formação veicular baseando-se em aspectos de agentes inteligentes e de sistema multiagentes, buscando fornecer autonomia ao agente para tomar a decisão sobre as ações a serem realizadas em relação a sua comunicação.

Os resultados obtidos ainda são iniciais, mas nos mostram a possibilidade de desenvolver tecnologias que venham permitir a interação entre os agentes de forma colaborativa, sem que o agente deixe de atender aos seus objetivos individuais e coletivos.

Além disso, acreditamos na possibilidade de usar tal abordagem proposta em outros tipos de aplicação, desde que se construam serviços baseados em regras e ações que atendam aos requisitos essenciais de cada uma com correte e pontualidade.

REFERÊNCIAS

- [1] M.M. Al-Doori, A.H. Al-Bayatti, and H. Zedan. Context aware architecture for sending adaptive hello messages in vanet. In *Proceedings of the 4th ACM International Workshop on Context-Awareness for Self-Managing Systems*, page 10. ACM, 2010.
- [2] A. Amoroso, L. Gandolfi, S. Grassilli, and M. Rocchetti. Frov: a distributed broadcast protocol for vanet experimental results. In *Ultra Modern Telecommunications Workshops, 2009. ICUMT '09. International Conference on*, pages 1 –8, Petersburg, Russia, oct. 2009.
- [3] P. Bijan, Md I. Khalil, and Md. A.N.B. Bikas. Vanet routing protocols: Pros and cons. *International Journal of Computer Applications*, 20(3):28–34, 2011.
- [4] G. Bittencourt. *Inteligencia artificial: ferramentas e teorias*. Univ. Federal de Santa Catarina, 2001.
- [5] V. Cabrera, F.J. Ros, and P.M. Ruiz. Simulation-based study of common issues in vanet routing protocols. In *Vehicular Technology Conference, 2009. VTC Spring 2009. IEEE 69th*, pages 1–5. IEEE, 2009.
- [6] E. Cambrozzi, J. Farines, R. Macêdo, and K. Werner. An adaptive failure detection system for vehicular ad-hoc networks. In *2010 IEEE Intelligent Vehicles Symposium (IV 2010)*, San Diego, USA, June 2010.
- [7] Y. Chen and Z. Wang. Formation control: a review and a new consideration. In *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, pages 3181 – 3186, Shaw Conference Center Edmont, Alberta Canadá, aug. 2005.
- [8] F. Cristian. Synchronous and asynchronous group communication. *Communications of the ACM*, 39(4):88–97, abril 1996.
- [9] E.H. Durfee and J.S. Rosenschein. Distributed problem solving and multi-agent systems: Comparisons and examples. *AAAI Technical Report*, 1001:48109, 1994.
- [10] J.A. Fax and R.M. Murray. Information flow and cooperative control of vehicle formations. *Automatic Control, IEEE Transactions on*, 49(9):1465 – 1476, sept. 2004.
- [11] L. Figueiredo, I. Jesus, J. Machado, J. Ferreira, and J.L. Carvalho. Towards the development of intelligent transportation systems. In *Intelligent Transportation Systems*,

2001. *Proceedings. 2001 IEEE*, pages 1206–1211, Oakland, California, USA, 2001. IEEE.
- [12] R. Ghabcheloo, A.P. Aguiar, A. Pascoal, and C. Silvestre. Synchronization in multi-agent systems with switching topologies and non-homogeneous communication delays. In *Decision and Control, 2007 46th IEEE Conference on*, pages 2327–2332, dec. 2007.
- [13] A. Gordillo, M. Calderon, and C.J. Bernardos. Enabling ip geomulticast services for vehicular networks. In *Intelligent Transport Systems Telecommunications, (ITST), 9th International Conference on*, pages 292–297, Nouveau Siecle Congress Centre, Lillie, France, 2009. IEEE.
- [14] M. Gramaglia, C.J. Bernardos, I. Soto, M. Calderon, and R. Baldessari. Ipv6 address autoconfiguration in geonetworking-enabled vanets: characterization and evaluation of the etsi solution. *EURASIP Journal on Wireless Communications and Networking*, 2012(1):19, 2012.
- [15] S. Halle. Automated highway systems: Platoons of vehicles viewed as a multiagent system. Master’s thesis, Faculté Des Sciences Et De Génie Université Laval Québec, June 2005.
- [16] S. Halle, B. Chaib-draa, and J. Laumonier. Car platoons simulated as a multiagent system. In *Proc. 4th Workshop on Agent-Based Simulation*, pages 57–63, Montpelier, France, 2003. Citeseer.
- [17] K. Jagadeesh, S. Siva Sathya, B. Govinda Laxmi, and B.B. Ramesh. A survey on routing protocols and its issues in vanet. *International Journal of Computer Applications*, 28(4):38–44, 2011.
- [18] D. Jiang and L. Delgrossi. IEEE 802.11 p: Towards an international standard for wireless access in vehicular environments. In *Vehicular Technology Conference, 2008. VTC Spring 2008. IEEE*, pages 2036–2040. IEEE, 2008.
- [19] Karl Henrik Johansson, Martin Tarngren, and Lars Nielsen. Vehicle applications of controller area network. In *Handbook of Networked and Embedded Control Systems*, Control Engineering, pages 741–765. Birkhauser Boston, 2005.
- [20] Jagadeesh Kakarla, S Siva Sathya, B Govinda Laxmiand, and Ramesh Babu B. A survey on routing protocols and its issues in vanet. *International Journal of Computer Applications*, 28(4):38–44, August 2011. Published by Foundation of Computer Science, New York, USA.
- [21] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil. Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions. *Communications Surveys Tutorials, IEEE*, 13(4):584–616, quarter 2011.

- [22] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner. Sumo (simulation of urban mobility). In *Proc. of the 4th Middle East Symposium on Simulation and Modelling*, pages 183–187, Sharjah – United Arab Emirates, 2002.
- [23] J.R.T. Lawton, R.W. Beard, and B.J. Young. A decentralized approach to formation maneuvers. *Robotics and Automation, IEEE Transactions on*, 19(6):933–941, 2003.
- [24] S. Liu and L. Cheng. A context-aware reflective middleware framework for distributed real-time and embedded systems. *Journal of Systems and Software*, 84(2):205–218, 2011.
- [25] N. A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers, Inc., 1996.
- [26] F.J. Martinez, C.K. Toh, J.C. Cano, C.T. Calafate, and P. Manzoni. A survey and comparative study of simulators for vehicular ad hoc networks (vanets). *Wireless Communications and Mobile Computing*, 2009.
- [27] N.F. Maxemchuk, P. Tientrakool, and T.L. Willke. Reliable neighborcast. *Vehicular Technology, IEEE Transactions on*, 56(6):3278–3288, 2007.
- [28] Renã Meier and Vinny Cahill. Exploiting proximity in event-based middleware for collaborative mobile applications. In *Distributed Applications and Interoperable Systems*, volume 2893 of *Lecture Notes in Computer Science*, pages 285–296. Springer Berlin / Heidelberg, 2003.
- [29] B.J. Moore and C. Canudas-de Wit. Formation control via distributed optimization of alignment error. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 3075 –3080, dec. 2009.
- [30] K. Ogata. *Engenharia de controle moderno*. Prentice Hall Brasil, third edition, 1998.
- [31] Alirio Sá and Raimundo Macêdo. Detectores de defeitos autônômicos para sistemas distribuídos. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, SBRC’2010*, pages 785–798, Gramado, Brasil, Maio 2010. SBRC / SBC.
- [32] P. Shankar, T. Nadeem, J. Rosca, and L. Iftode. Cars: Context-aware rate selection for vehicular networks. In *Network Protocols, 2008. ICNP 2008. IEEE International Conference on*, pages 1 –12, oct. 2008.
- [33] M.L. Sichitiu and M. Kihl. Inter-vehicle communication systems: a survey. *Communications Surveys Tutorials, IEEE*, 10(2):88 –105, quarter 2008.
- [34] C. Sommer, R. German, and F. Dressler. Bidirectionally coupled network and road traffic simulation for improved ivc analysis. *Mobile Computing, IEEE Transactions on*, 10(1):3 –15, jan. 2011.
- [35] P. Stone and M. Veloso. Multiagent systems: A survey from a machine learning perspective. *Autonomous Robots*, 8(3):345–383, 2000.

- [36] K.P. Sycara. Multiagent systems. *AI magazine*, 19(2):79–92, 1998.
- [37] Andrew S. Tanenbaum. *Computer Networks*. Prentice-Hall, second edition, 1988.
- [38] K. Tindell, A. Burns, and A.J. Wellings. Calculating controller area network (can) message response times. *Control Engineering Practice*, 3(8):1163–1169, 1995.
- [39] A. Varga et al. The omnet++ discrete event simulation system. In *The European Simulation Multiconference (ESM 2001)*, pages 319–324, Prague, Czech Republic, 2001.
- [40] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.P. Hubaux. Traci: an interface for coupling road traffic and network simulators. In *Proceedings of the 11th communications and networking simulation symposium*, pages 155–163, Ottawa, ON, Canada, 2008. ACM.
- [41] G. Weiss. *Multiagent systems: a modern approach to distributed artificial intelligence*. The MIT press, 2000.
- [42] T.L. Willke, P. Tientrakool, and N.F. Maxemchuk. A survey of inter-vehicle communication protocols and their applications. *Communications Surveys Tutorials, IEEE*, 11(2):3–20, quarter 2009.
- [43] M. Wooldridge. *An Introduction to MultiAgent Systems*. John Wiley & Sons, second edition, 2009.
- [44] M. Wooldridge and N.R. Jennings. Intelligent agents: Theory and practice. *Knowledge engineering review*, 10(2):115–152, 1995.
- [45] W. Xiang, P. Richardson, and J. Guo. Introduction and preliminary experimental results of wireless access for vehicular environments (wave) systems. In *Mobile and Ubiquitous Systems-Workshops, 2006. 3rd Annual International Conference on*, pages 1–8. IEEE, 2006.
- [46] F. Xiao, L. Wang, Jie Chen, and Yanping Gao. Finite-time formation control for multi-agent systems. *Automatica*, 45(11):2605 – 2611, nov. 2009.