UNIVERSIDADE ESTADUAL DO CEARÁ

CENTRO DE CIÊNCIAS E TECNOLOGIA

PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO

DANIEL SUCUPIRA LIMA

E-PROBT: UM PROTOCOLO PROBABILÍSTICO E TEMPORAL BASEADO EM TEORIA DOS JOGOS PARA MITIGAÇÃO DO PROBLEMA BROADCAST STORM EM VANETS

FORTALEZA – CEARÁ

2016

DANIEL SUCUPIRA LIMA

E-PROBT: UM PROTOCOLO PROBABILÍSTICO E TEMPORAL BASEADO EM TEORIA DOS JOGOS PARA MITIGAÇÃO DO PROBLEMA BROADCAST STORM EM VANETS

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Orientador: Prof. Dr. Joaquim Celestino Júnior

FORTALEZA – CEARÁ

2016

Dados Internacionais de Catalogação na Publicação

Universidade Estadual do Ceará

Sistema de Bibliotecas

Lima, Daniel Sucupira.
E-ProbT: um protocolo probabilístico e temporal baseado em Teoria dos Jogos para mitigação do problema broadcast storm em VANETs [recurso eletrônico] / Daniel Sucupira Lima. - 2016.
1 CD-ROM: il.; 4 ¾ pol.

CD-ROM contendo o arquivo no formato PDF do trabalho acadêmico com 81 folhas, acondicionado em caixa de DVD Slim (19 x 14 cm x 7 mm).

Dissertação (mestrado acadêmico) - Universidade Estadual do Ceará, Centro de Ciências e Tecnologia, Mestrado Acadêmico em Ciência da Computação, Fortaleza, 2016.
Área de concentração: Ciência da Computação.
Orientação: Prof. Dr. Joaquim Celestino Júnior.

1. VANETs. 2. Teoria dos Jogos. 3. Tempestade Broadcast. 4. Disseminação de informações. I. Título.

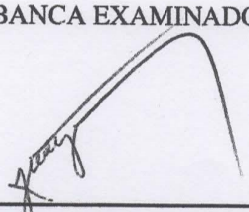
DANIEL SUCUPIRA LIMA

E-PROBT: UM PROTOCOLO PROBABILÍSTICO E TEMPORAL BASEADO EM TEORIA DOS JOGOS PARA MITIGAÇÃO DO PROBLEMA BROADCAST STORM EM VANETS

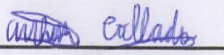
Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências e Tecnologia da Universidade Estadual do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Ciência da Computação

Aprovada em: 02 de fevereiro de 2016

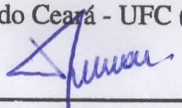
BANCA EXAMINADORA



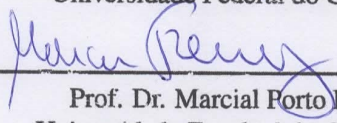
Prof. Dr. Joaquim Celestino Júnior (Orientador)
Universidade Estadual do Ceará – UECE



Prof. Dr. Arthur de Castro Callado
Universidade Federal do Ceará - UFC (Campus Quixadá)



Prof. Dr. José Neuman de Souza
Universidade Federal do Ceará - UFC



Prof. Dr. Marcial Porto Fernandez
Universidade Estadual do Ceará - UECE

AGRADECIMENTOS

Primeiramente, agradeço a Deus que esteve comigo em toda a minha caminhada acadêmica e estará comigo durante toda a minha vida.

Agradeço à meus pais e irmãs por todo apoio dado a mim no decorrer dos anos.

Agradeço à minha tia, Paula Franssinetti Sucupira Leandro, por tudo o que tem feito em minha vida. Cada passo nessa caminhada só foi possível graças à sua ajuda.

Agradeço aos meus amigos e aos companheiros do LARCES por estar comigo em diversos momentos.

Faço um agradecimento especial ao Prof. Dr. Joaquim Celestino Júnior. Guardarei pra sempre comigo os ensinamentos passados no decorrer desses anos. Obrigado pela sua orientação desde o período da graduação.

Agradeço ao Prof. Dr. André Ribeiro Cardoso por todos os ensinamentos passados durante os períodos de graduação e pós-graduação.

Agradeço também ao Mestre Filipe Maciel Roberto, que empenhou muitas horas para passar os seus conhecimentos sobre VANETs. Essa conquista é resultado da sua dedicação ao ensino.

"Uns confiam em carros e outros em cavalos, mas nós faremos menção do nome do Senhor nosso Deus."

(Salmos 20:7)

RESUMO

VANETs são redes auto-organizadas nas quais os nós constituintes são veículos. Elas podem ser classificadas como uma subcategoria das MANETs. Devido às suas características especiais, necessitam de protocolos criados especificamente para o seu cenário de atuação. Diversos tipos de aplicações podem ser criados para as mesmas, tais como, aplicações de segurança, gerenciamento de tráfego, manutenção de sistemas e conforto para motoristas e passageiros. Em geral, aplicações desenvolvidas para VANETs fazem uso do *broadcast* de informações. Porém, existem diversos problemas a serem levados em consideração. Um dos problemas centrais é a tempestade de *broadcast*. Neste trabalho é proposto um protocolo probabilístico e temporal baseado em Teoria dos Jogos e Estatística Média Móvel Exponencial Ponderada, intitulado E-ProbT, para a mitigação da tempestade *broadcast*. De acordo com as decisões sobre encaminhamento de pacotes, são usadas recompensas e penalidades para nós vizinhos, aplicando-se a Teoria dos Jogos. Realiza-se uma estimação das maiores probabilidades de encaminhamento dos nós vizinhos, utilizando-se a Estatística Média Móvel Exponencial Ponderada. O desempenho do *E-ProbT* será medido e comparado com o dos protocolos *Blind Flooding*, *Wheighted p-Persistence*, *AutoCast*, *Irresponsible Forwarding* e *ProbT*. Os resultados mostram o *E-ProbT* como uma nova abordagem efetiva para *broadcast* de informações.

Palavras-chave: VANETs. Teoria dos Jogos. Tempestade Broadcast. Disseminação de informações.

ABSTRACT

VANETs are self-organized networks in which their constituent nodes are vehicles. They can be classified as a subcategory of MANETs. Due to their special characteristics, they demand protocols designed specifically for their scenario of action. Various types of applications can be created for VANETs such as security, traffic management, systems maintenance and comfort for drivers and passengers. In general, applications developed for VANETs make use of broadcast information. However, there are many issues to be considered. One of the central problems is the broadcast storm. In this paper, a temporal probabilistic protocol based on Game Theory and the Statistic Exponentially Weighted Moving Average, entitled E-ProbT, is proposed to mitigate the broadcast storm problem. According to the packet forwarding decisions, rewards and penalties are defined for neighbors, applying the Game Theory. An estimation of the highers forwarding probabilities of the neighbors is executed utilizing the Statistic Exponentially Weighted Moving Average. The E-ProbT performance will be measured and compared to the performance of the protocols: blind flooding, Wheighted p-Persistence, AutoCast, Irresponsible Forwarding and ProbT.

Keywords: VANETs. Game Theory. Broadcast Storm. Dissemination of Information

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE CÓDIGOS-FONTE

LISTA DE ABREVIATURAS E SIGLAS

CCH	Control Channel
DSRC	Dedicated Short Range Communications
FCC	Federal Comission Comunication
GPS	Global Position System
IEEE	Institute of Electrical and Electronics Engineers
LLC	Logical Link Control
MAC	Media Access Control
NS-3	Network Simulator Version 3
SCH	Service Channel
SOS	Save Our Souls
SUMO	Simulation of Urban Mobility
TCP	Transmission Control Protocol
UDP	User Data Protocol
V2I	Vehicle To Infrastructure
V2V	Vehicle To Vehicle
VANETs	Vehicular Ad Hoc Networks
WAVE	Wireless Access In The Vehicular Environment
WME	Wave Management Entity
WSMP	Wave Short Message Protocol

SUMÁRIO

1 INTRODUÇÃO

As comunicações *ad hoc wireless* e as tecnologias veiculares tem atingido um rápido desenvolvimento (PANICHPAPIBOON; PATTARA-ATIKOM, 2012). Diferentes tecnologias tem sido desenvolvidas com a capacidade de fornecer serviços diferenciados aos veículos. Nesse cenário, *Vehicular Ad Hoc Networks* (VANETs) podem ser usadas como uma alternativa para disseminação de informações (PANICHPAPIBOON; PATTARA-ATIKOM, 2012).

Um dos principais desafios da disseminação de informações em VANETs é a tempestade de *broadcast* (*Broadcast Storm*) (WISITPONGPHAN et al., 2007). Quando uma tempestade de *broadcast* está acontecendo em um determinado cenário, existe um alto nível de contenção e colisão de pacotes como resultado de um número excessivo de encaminhamentos (WISITPONGPHAN et al., 2007).

Neste trabalho é proposto um protocolo para a modelagem de decisão sobre o encaminhamento de pacote, intitulado E-ProbT. Ele será um protocolo de múltiplos saltos, combinando características de protocolos probabilísticos e de *delay*.

Considera-se que os carros que deveriam encaminhar os pacotes podem assumir um comportamento egoísta e não reencaminhá-los. Assim, a Teoria dos Jogos será usada para modelar recompensas e penalidades para os vizinhos de acordo com as suas decisões sobre encaminhamentos.

Algoritmos probabilísticos realizam a decisão sobre encaminhamento com base em um valor de *threshold*. Ele será gerado dinamicamente, de forma a adaptar a dinâmica da rede. Ele será definido de acordo com os melhores valores de probabilidade de encaminhamento dos vizinhos. Dessa forma, busca-se garantir retransmissões necessárias e evitar retransmissões redundantes. Esse *threshold* será gerado com base em entradas antigas e atuais fazendo-se uso da estatística Média Móvel Exponencial Ponderada (*Exponentially Weighted Moving Average - EWMA*).

Serão analisadas diferentes características dos vizinhos para a tomada de decisão sobre encaminhamento de pacotes. Elas serão traduzidas em termos matemáticos através de equações probabilísticas. Os resultados das equações serão classificados e combinados. O resultado da combinação indicará a decisão sobre encaminhamento.

O E-ProbT também faz uso de janelas de tempo. Elas serão usadas de forma a verificar a realização do encaminhamento de pacotes por outros nós.

1.1 MOTIVAÇÃO

No ambiente científico existe um contínuo esforço no sentido de encontrar melhores soluções ou alternativas para problemas do mundo real. Desta forma, busca-se criar uma nova solução para a tomada de decisão de encaminhamento de pacotes.

1.2 OBJETIVOS

O objetivo geral e os específicos estão descritos abaixo.

1.2.1 Objetivo Geral

O Protocolo proposto, intitulado *E-ProbT*, visa a criação de um protocolo probabilístico e temporal baseado em Teoria dos Jogos, estatística Média Móvel Exponencial Ponderada, fórmulas probabilísticas e mecanismos de temporização para a modelagem de decisão sobre o encaminhamento de pacotes. A junção desses mecanismos será feita de forma a maximizar a taxa de entrega e minimizar a taxa de redundância (métricas explicadas no Capítulo 5).

1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

a) Desenvolver uma análise referente ao problema do encaminhamento de pacotes em VANETs no cenário de protocolos de *broadcast* de informação probabilísticos;
b) Realizar um estudo sobre Teoria dos Jogos e a estatística Média Móvel Exponencial Ponderada;
c) Levantar as soluções utilizadas atualmente para *broadcast* de informações no cenário de VANETs;
d) Implementar a modelagem de encaminhamento de pacotes com base nas características propostas;
e) Comparar o protocolo proposto com outros trabalhos existentes na literatura usando simulações.

1.3 CONTRIBUIÇÃO

Um dos trabalhos base da proposta é o protocolo *ProbT* (LIMA et al., 2015). Ele foi publicado na *International Conference on Information Networking* (ICOIN) em 2015, de qualis Capes B1. Na citada conferência, foi eleito o melhor artigo, recebendo o título de "*Best Paper*".

Diversos mecanismos para um melhor desempenho do protocolo foram produzidos, possibilitando a criação de um novo protocolo, intitulado *E-ProbT*, descrito neste trabalho. O *E-ProbT* é um trabalho aceito para publicação na *Symposium on Applied Computing* (SAC 2016), de qualis Capes A1.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a fundamentação teórica necessária para a elaboração do E-ProbT, incluindo VANETs, Teoria dos Jogos e a estatística Média Móvel Exponencial Ponderada.

2.1 VANETS

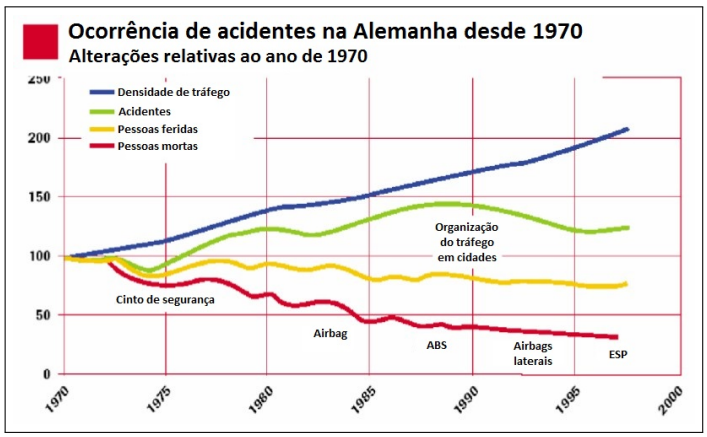
Os avanços recentes em tecnologia têm possibilitado um aumento no número de carros equipados com aparelhos GPS (*Global Position System*) e dispositivos Wi-Fi. Eles capacitam os nós para a realização de comunicações entre veículos (V2V - *Vehicle To Vehicle*), permitindo o surgimento das redes *ad hoc* veiculares (VANETs - *Vehicular Ad Hoc Networks*) (ZHANG, 2011). Em geral, as VANETs compõem uma sub-categoria das MANETs, nas quais os nós são veículos.

2.1.1 Motivação

Vários fatores podem ser apontados como razões para aplicações VANETs (KARGL, 2006):

- Segurança: esta é a principal razão para o surgimento das redes veiculares. Diversas inovações tecnológicas têm sido criadas para o aumento da segurança de motoristas e passageiros, de forma a reduzir o número de acidentes no trânsito. A Figura 1 mostra a diminuição do número de pessoas mortas e feridas na Alemanha, desde 1970, com o surgimento de inovações visando a segurança:

Figura 1 – Histórico de acidentes.



Fonte: Figura edidata de (KARGL, 2006)

Dentre diversas aplicações possíveis, as principais são:

  - Advertências sobre violações de sinais de trânsito;
  - Advertências sobre colisões;
  - Advertências sobre aproximação de veículos de emergência tais como ambulâncias, viaturas, dentre outros;
  - Rastreamentos de veículos roubados;
  - Serviços de SOS (*Save Our Souls*);
  - Advertências sobre velocidades em curvas;
  - Advertências sobre condições de pistas em geral.

- Aplicações de gerenciamento de tráfego:
  - Assistentes de interseções em curvas;
  - Controle inteligente de fluxo de tráfego;
  - Gerenciamento de conglomerados de carros.
- Aplicações de manutenção de hardware e/ou software do veículo:
  - Notificações de *Recall*;
  - Notificações de reparo *just-in-time*;
  - Diagnósticos *wireless*;
  - Atualização de software.
- Aplicações de conforto para motoristas e passageiros:
  - Aplicações para localização de vagas em estacionamento;
  - Orientação de rota;
  - Download de mapas;
  - Mensagens instantâneas.

2.1.2 Desafios

Podem ser elencados diversos desafios, com base em (ALVES et al., 2009), (YOUSEFI et al., 2006) e (KUROSE; ROSS, 2010), para as VANETs:

- A alta velocidade dos veículos: os veículos podem estar com altas velocidades. Essa é uma das características que diferenciam os nós das VANETs dos nós das MANETs;
- Dinamismo da topologia da rede: considerando os grandes valores de velocidade dos veículos, as topologias podem mudar rapidamente. Tal característica é ruim para as aplicações interessadas no conhecimento de dados como topologia da rede, densidade de

veículos, quantidade de vizinhos, dentre outros;

- Particionamento da rede: a falta de veículos em determinadas porções da rodovia pode fazer com que partes das rede fiquem desconectadas, produzindo uma partição na rede;
- Quebra de *links*: uma vez particionada a rede, *links* entre veículos de diferentes porções são quebrados, impossibilitando, *a priori*, a comunicação entre os veículos;
- *Links* de pequena duração: considerando que os carros nas rodovias podem ter diferentes velocidades, dois nós podem ficar dentro do raio de transmissão um do outro por um pequeno intervalo de tempo. Assim, as comunicações diretas entre dois veículos devem ser feitas em um tempo hábil, para garantir a entrega dos dados antes que os *links* sejam quebrados;
- Tamanho das redes: os protocolos projetados para VANETs devem ser escaláveis, entregando a mesma qualidade de serviço independente da quantidade de veículos na rede;
- Características naturais: diversos fatores naturais podem causar interferência na qualidade das transmissões, tais como a presença de prédios, árvores e outros veículos. As redes *wireless* em geral possuem esse desafio.

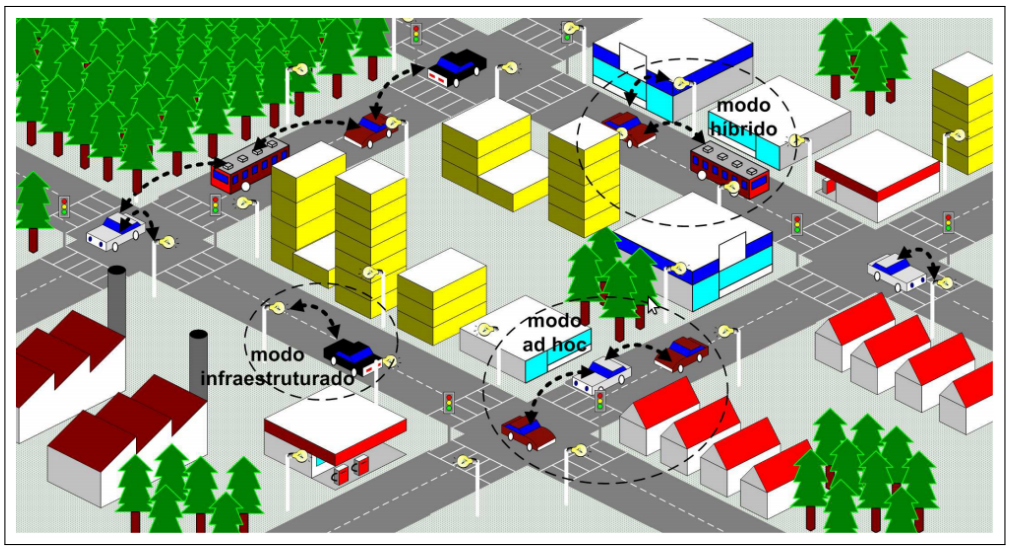
2.1.3 Arquiteturas

Por definição, as VANETs são formadas por nós móveis que se comunicam sem o auxílio de qualquer infra-estrutura existente. Entretanto, muitas pesquisas foram realizadas em VANETs também utilizando uma certa infra-estrutura. Tais pesquisas também eram classificadas como VANETs. Dessa forma, o termo VANETs passou a ser usado indistintamente. Por essa razão, existem três grandes categorias de VANETs (ALVES et al., 2009):

- *Ad Hoc* Puro (*Vehicular Ad Hoc Network*): existem comunicações apenas entre veículos (V2V). Assim, não é utilizada qualquer infra-estrutura;
- Infraestruturada: existem comunicações apenas entre veículo e nós de infra-estrutura (*Vehicle To Infrastructure* - V2I). Esses nós de infra-estrutura são colocados de forma estática ao longo das rodovias. Eles funcionam como pontos de acesso das redes IEEE 802.11;
- Híbrida: a junção das categorias infraestruturada e híbrida. Existe a possibilidade de comunicação entre veículos ou entre veículos e nós estáticos na rodovia.

Esses três tipos de arquiteturas são exibidos na Figura 2:

Figura 2 – Arquiteturas de VANETs.

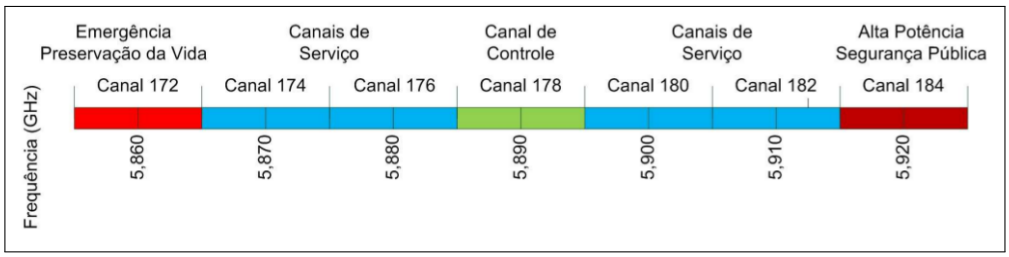


Fonte: (ALVES et al., 2009)

2.1.4 Padrões das VANETs

A primeira iniciativa para padronização das redes veiculares ocorreu nos Estados Unidos, em 1990 (ALVES et al., 2009). A FCC (*Federal Comission Comunication*) alocou 75 MHz do espectro de frequências, na faixa de 5,9 GHz (5,850 - 5,925). Essa faixa de frequências, exibida na Figura 3, ficou então reservada para aplicações DSRC (*Dedicated Short Range Communications*).

Figura 3 – Alocação de espectro para aplicações DSRC nos EUA.



Fonte: (ALVES et al., 2009)

O espectro foi dividido em 7 canais, cada um de 10 MHz. O canal central, 178, é um canal de controle. Os outros canais são de serviço. O envio e recebimento de dados nesses canais são baseados em intervalos de sincronização. Um intervalo de sincronização é composto de um intervalo no CCH *(Control Channel)* e um intervalo no SCH *(Service Channel)*. No intervalo CCH é observado o canal de controle. No intervalo SCH são utilizados os canais de serviço. Os canais extremos, 172 e 184, são voltados para aplicações anticolisão e aplicações públicas para

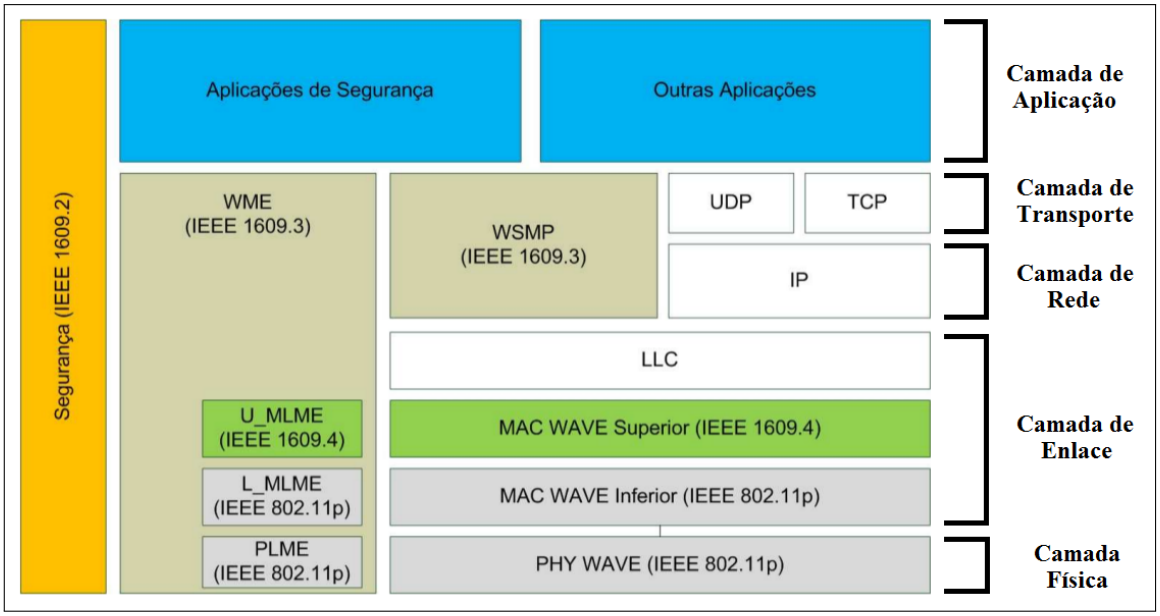
segurança e emergências em geral. As demais aplicações devem utilizar os canais de serviço, a saber: 174, 176, 180 e 182 (ALVES et al., 2009).

A faixa DSRC é classificada como livre e licenciada. Enquanto livre, não são cobradas taxas pela sua utilização. Enquanto licenciada, existem restrições das aplicações e tecnologias utilizadas.

Não apenas os Estados Unidos fizeram esforços para uso de tecnologias veiculares. Exemplo disso é o DSRC europeu, reservando a intervalo de frequência na faixa de 5.8 GHz (RITO, 2011). A partir do surgimento de diferentes tecnologias veiculares, surge a necessidade de definir um padrão de interoperabilidade. Assim, o IEEE (*Institute of Electrical and Electronics Engineers*) começou em 2004 uma padronização dentro do seu grupo de trabalho.

Existe um conjunto de padrões organizados em uma arquitetura chamada IEEE 802.11p WAVE (*Wireless Access In The Vehicular Environment*). Para a criação desse conjunto de padrões foi utilizado como base o padrão americano. A pilha de protocolos da arquitetura WAVE é descrita na Figura 4:

Figura 4 – Pilha de protocolos WAVE.



Fonte: (LIMA et al., 2014)

A pilha de protocolos WAVE é composta de um plano de dados e um plano de gerenciamento. O plano de dados é responsável pela troca de informações entre os dispositivos. O plano de gereciamento é responsável pela configuração e manutenção do sistema.

Existem entidades de gerenciamento criadas para as camadas de rede, enlace e física.

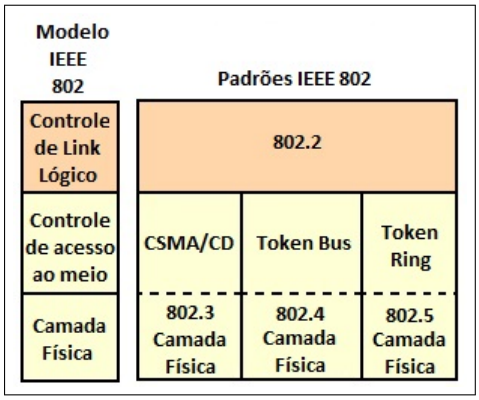
Para a camada física é definida a *Physical Layer Management Entity* (PLME). Para a camada de MAC (*Media Access Control*) é definida a *MAC Layer Management Entity* (MLME).

Esse conjunto de entidades de gerenciamento é agrupado em uma entidade chamada Entidade de Gerenciamento WAVE (WME - *Wave Management Entity*). Através da WME, as aplicações tem acesso à informações de outras camadas.

Na camada de transporte/rede podem ser usados os protocolos UDP/IP ou TCP/IP. O protocolo TCP (*Transmission Control Protocol*) deve passar por algumas modificações para ter um desempenho adequado para redes VANETs (BECHLER et al., 2005). Como alternativa para as camadas de transporte e rede com UDP/IP ou TCP/IP, pode ser usado o protocolo WSMP (*Wave Short Message Protocol*). Ele é projetado especialmente para o cenário de VANETs, tentanto tornar o envio de mensagens o mais eficente possível.

Assim como em demais redes 802.11, a camada LLC (*Logical Link Control*) é usada para ocultar a diferença entre os diversos tipos de redes 802.11, fornecendo um único formato e uma única interface com a camada de rede. Ela é projetada para trabalhar com diferentes protocolos MAC (CSMA/CD, CSMA/CA dentre outros). Uma ilustração das camadas LLC, enlace e a física é apresentada na Figura 5:

Figura 5 – Camada LLC.



Fonte: Figura edidata de (SHIPMAN, 2000)

É utilizado o mecanismo CSMA/CA na camada MAC do modelo WAVE. Os padrões IEEE P1609.4 e IEEE 802.11p fazem uma adaptação do padrão IEEE 802.11 para utilizar múltiplos canais da arquitetura WAVE. Foram evitadas grandes modificações na camada física para não ser necessária a utilização de uma nova tecnologia de radiotransmissão. Dentre as modificações feitas, pode ser citada a alteração da largura dos canais para 10 MHz (ALVES et al., 2009).

2.1.5 Disseminação de informações em VANETs

Quando deseja-se fazer o *broadcast* de algum pacote na rede, por definição, ele deve ser entregue a todos os nós da rede. Quando um nó fonte deseja disseminar um pacote, apenas um subconjunto dos nós da rede está dentro do seu raio de transmissão. Dessa forma, para que os nós fora do raio de transmissão recebam o pacote, os nós dentro do raio devem reencaminhá-lo.

Pesquisas mostram que o reencaminhamento feito por todos os nós da rede é ineficiente, causando os problemas associados à tempestade *broadcast*, tais como:

- Elevado número de colisões;
- Grandes períodos de contenção;
- Envio de dados desnecessários, pois muitos nós recebem diversas duplicatas de um mesmo pacote.

Dessa forma, deve existir uma política de seleção de nós encaminhadores. Por ser uma VANET, os cálculos sobre encaminhamentos devem ser feitos de forma distribuída, ou seja, cada nó tem a autonomia para "decidir" sobre a realização do encaminhamento.

Existem na literatura de VANETs diversas políticas de reencaminhamento. Cada uma possui vantagens e desvantagens. Cabe ao projetista de um protocolo a decisão sobre a política de *broadcast* a ser utilizada, de forma que sejam atendidas as necessidades da aplicação desejada. Em geral, deve-se criar políticas que minimizem a redundância de dados na rede.

2.1.6 Diferentes Regimes de Broadcast em VANETs

Considerando os diferentes cenários nos quais as redes veiculares podem ser aplicadas, existem diferentes densidades veiculares com as quais as redes devem lidar. O tráfego veicular é classificado, de acordo com a sua densidade, em três diferentes regimes (TONGUZ et al., 2007):

1. Regime de tráfego esparso: existe uma pequena quantidade de veículos na rede. Protocolos devem ser projetados especialmente para esse cenário de atuação. Nesse regime, é comum a ocorrência de partições na rede. Assim, não é possível entregar dados através dos encaminhamentos feitos por protocolos tradicionais. Uma estratégia utilizada para superar essa dificuldade é o *store-carry-and-forward* (TONGUZ et al., 2007). Em geral, utiliza-se um veículo que trafega de uma partição da rede para a outra, para realizar a entrega do pacote. Na porção da rede original, o veículo recebe o pacote e o armazena. O

veículo atravessa a região esparsa e, ao chegar na outra porção da rede, prossegue com o encaminhamento do pacote;

2. Regime de tráfego denso: neste regime existe uma grande quantidade de veículos na rede. Caso não seja feito um controle eficiente dos nós que farão a disseminação, um número excessivo de encaminhamentos será feito, causando os problemas associados à tempestade *broadcast* (TONGUZ et al., 2007);
3. Regime de tráfego regular: a densidade de carros na rodovia está entre alta e baixa. Nos regimes anteriores existia uma densidade veicular aproximadamente igual para todos os veículos na rede: ou alta ou baixa. No regime regular não existe uma uniformidade na densidade dos veículos. Assim, deve ser criado um protocolo aplicável à nós com diferentes densidades de vizinhos (TONGUZ et al., 2007).

O protocolo proposto neste trabalho, *E-ProbT*, é projetado para atuar nos regimes regular e denso. Para ser aplicável no regime esparso devem ser adicionados mecanismos de armazenamento, espera e envio.

2.2 TEORIA DOS JOGOS

Interação entre diferentes entidades é uma prática comum quando se estabelece algum tipo de comunicação. Exemplos disso são protocolos de roteamento, nos quais existem trocas de mensagens entre roteadores. Em muitas situações o comportamento de um agente afeta de forma positiva ou negativa o comportamento de outro. Essa característica é chamada de interdependência. Chama-se de cenários estratégicos as situações nas quais existe a presença de interdependência. Nesses casos, um agente deve escolher a melhor estratégia a ser tomada em face da previsão da ação de outros agentes envolvidos (ROBERTO, 2010). Por causa da grande importância das ações estratégicas em uma sociedade, diversos estudos foram realizados gerando uma teoria das ações estratégicas: a Teoria dos Jogos (FUDENBERG; TIROLE, 1991).

Um jogo é definido como uma estrutura logicamente consistente e matematicamente precisa que descreve formalmente um cenário estratégico (WATSON, 2007).

Um jogo possui os seguintes componentes:

- Jogadores: tomadores de decisões racionais;
- Estratégias: conjunto de ações que podem ser tomadas por um jogador;
- Resultados: conjunto de estratégias que os jogadores escolheram;
- *Payoff*: função de utilidade de uma determinada estratégia para cada jogador.

Um dos dilemas estudados pela Teoria dos Jogos é o Dilema do Voluntário (WEESIE, 1993). Nesse dilema o bem coletivo é proporcionado pela contribuição de um único voluntário. Um grupo de elementos (chamados jogadores) está interessado em determinado bem. Entretanto, para que se tenha esse bem, pelo menos um dos jogadores deve se voluntariar e fazer determinada ação. Ao se voluntariar e fazer determinada ação, esse jogador deve arcar com determinado custo. Caso nenhum nó se voluntarie, os nós não terão esse bem desejado. Dessa forma, os jogadores devem lidar com a seguinte dúvida: ou eles se voluntariam e tomam determinada ação (arcando com determinado custo) ou eles ficam quietos e esperam o voluntariamento por parte de outro jogador (correndo assim o risco de não obter o bem desejado se ninguém se voluntariar).

O cenário no qual um conjunto de nós deve tomar uma decisão independente sobre o encaminhamento de pacotes pode ser modelado pelo dilema dos voluntários. Nesse caso, a decisão a ser tomada é se um nó deve ou não encaminhar um pacote. O encaminhamento de pacotes é o bem coletivo desejado, pois deseja-se fazer o *broadcast* da informação. O custo é a perda associada à transmissão de um pacote.

2.3 MÉDIA MÓVEL EXPONENCIAL PONDERADA

A média móvel exponencial ponderada é uma estatística através da qual é calculada uma média de valores atribuindo-se diferentes pesos para entradas antigas e recentes (ČISAR; ČISAR, 2011). Essa média é definida de acordo com a Fórmula (2.1):

$$EWMA = (1-\alpha) * EWMA + \alpha * Sample \tag{2.1}$$

Onde:
EWMA = Valor da média estimado atualmente;
Sample = Valor da amostra medido;
α = Uma constante usada para atribuição de pesos. É definida no intervalo $(0,1]$ em $\mathbb{R}$;

A constante α indica o peso com o qual as entradas recentes entram no cálculo da média. Se definida com o valor 1, indica-se que a média é igual ao valor da amostra mais recente, desprezando-se as entradas antigas. Valores maiores que 0.5 atribuem um maior peso para as entradas recentes. Valores menores que 0.5 atribuem um menor peso para as entradas recentes (ČISAR; ČISAR, 2011).

2.3.1 Uso da EWMA para cálculo do intervalo de *timeout* do TCP

O TCP utiliza um mecanismo de controle de temporização/retransmissão para recuperar segmentos perdidos (JACOBSON, 1988) (KUROSE; ROSS, 2010). Muitos pontos devem ser definidos para um correto funcionamento desse algoritmo. Um dos pontos importantes é definido pela seguinte pergunta: quanto tempo deve durar esse intervalo para retransmissão?

Esse intervalo deve ser maior que tempo de viagem de ida e volta de um pacote (RTT), ou seja, o tempo definido entre o envio de um pacote e a recepção da confirmação do mesmo. Valores de *timeout* consideravelmente pequenos indicarão falsas perdas de pacotes, causando retransmissões desnecessárias. Valores de *timeout* consideravelmente grandes indicarão uma perda de pacote tardiamente. Assim, deve-se buscar uma forma de cálculo do *timeout* que seja adequada.

Essa forma deve levar em consideração que os RTTs podem sofrer variações como resultado de congestionamento dos roteadores e a variações de carga nos sistemas finais. São realizados diversos passos no TCP para determinação do valor de *timeout*.

O TCP mede, para um dos pacotes que estão sendo transmitidos por vez, o RTT. Após a chegada desse pacote, faz-se a medição para outro, de forma que esse procedimento é feito continuamente. Esse RTT que é medido é chamado de *SampleRTT*.

O TCP calcula um valor esperado para o próximo RTT com base nos valores de *SampleRTT* medidos. Esse valor esperado de RTT é chamado de *EstimatedRTT* e é calculado de acordo com a Fórmula (2.2):

$$EstimatedRTT \ = \ (1-\alpha) * EstimatedRTT \ + \ \alpha * SampleRTT \tag{2.2}$$

Onde:

EstimatedRTT = RTT estimado;

SampleRTT = Valor da amostra de RTT medido;

α = Uma constante usada para atribuição de pesos;

Essa fórmula é uma combinação ponderada. Em estatística, essa média é denominada média móvel exponencial ponderada. O valor recomendado para α é $\alpha = 1/8$, ou seja, $\alpha = 0,125$

(JACOBSON, 1988) (KUROSE; ROSS, 2010). Assim, teremos a Fórmula (2.3):

$$EstimatedRTT = 0,875 * EstimatedRTT + 0,125 * SampleRTT \quad (2.3)$$

Além de estimar o RTT, deve ser feita uma estimativa da sua variação. Essa estimativa é feita com base na Fórmula (2.4):

$$DevRTT = (1-\beta) * DevRTT + \beta * |SampleRTT - EstimatedRTT| \quad (2.4)$$

Onde:

DevRTT = Variação de RTT estimada;

SampleRTT = Valor da amostra de RTT medido;

EstimatedRTT = RTT estimado;

β = Uma constante usada para atribuição de pesos;

O valor recomendado para β é 0,25 (JACOBSON, 1988) (KUROSE; ROSS, 2010). Por fim, para cálculo do *timeout*, devem ser considerados em uma fórmula a estimação e a variação do RTT. O *timeout* é definido com base na Fórmula (2.5):

$$TimeoutInterval = EstimatedRTT + 4 * DevRTT \quad (2.5)$$

Onde:

TimeoutInterval = Valor de timeout definido;

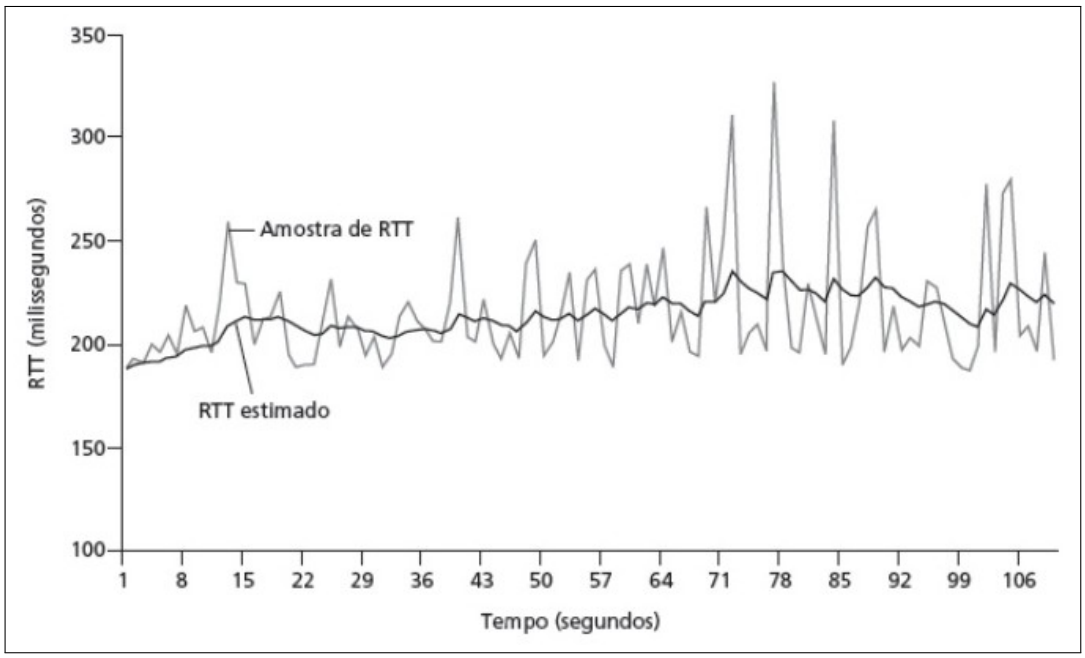
EstimatedRTT = RTT estimado;

DevRTT = Variação de RTT estimada;

A Figura 6 mostra os valores de *SampleRTT* e *EstimatedRTT* para o valor de $\alpha = 0,125$.

Os valores de *SampleRTT* e *EstimatedRTT* foram medidos para uma conexão TCP entre *gaia.cs.umass.edu* (em Amherst, Massachussets) e *fantasia.eurecom.fr* (no sul da França). A Figura 6 mostra uma atenuação dos valores de *SampleRTT* para o cálculo do *EstimatedRTT*.

Figura 6 – Estimação de RTTs.



Fonte: (KUROSE; ROSS, 2010)

2.4 USO DE PROBABILIDADE POR PARTE DE PROTOCOLOS DE BROADCAST PROBABILÍSTICOS

Em geral, os protocolos probabilísticos tomam a sua decisão de encaminhamento de pacote com base na geração de uma probabilidade. Esse valor pode ser baseado em uma ou mais características da rede. Ao receber um pacote, um nó deve tomar a decisão sobre encaminhamento. Assim, ele faz uso da sua fórmula probabilística para gerar a sua probabilidade de encaminhamento.

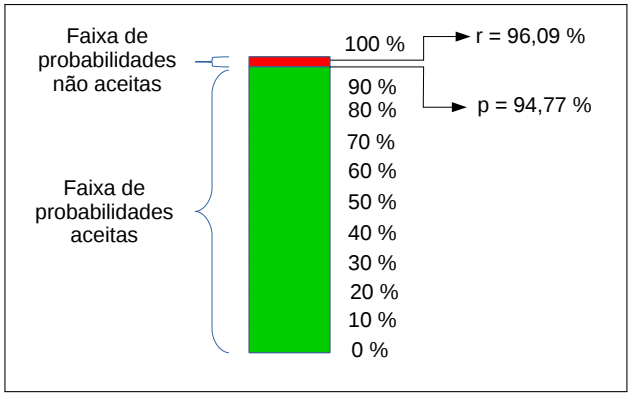
Em paralelo, deve ser gerado um número randômico, de acordo com a distribuição uniforme, que será comparado com tal probabilidade. Considere um exemplo no qual um nó recebeu um pacote e gerou uma probabilidade de encaminhamento p igual a 70 %. Na sequência, ele gera um número randômico r. Caso esse número randômico r esteja entre 0 e 70 %, ele faz o encaminhamento do pacote. Caso contrário, ele descarta.

Os nós com as maiores probabilidades de encaminhamento terão uma maior tendência a encaminhar o pacote e vice-versa. Apesar da probabilidade de encaminhamento quantificar determinada característica do nó receptor, tal mecanismo não garante que os nós com as maiores probabilidades encaminharão o pacote, nem que os nós com as menores probabilidades não encaminharão o dado. É possível a ocorrência de uma situação na qual um nó receptor com uma alta probabilidade de encaminhamento descarte o pacote e um outro nó com uma baixa

probabilidade de encaminhamento o encaminhe.

Consideremos dois casos. O caso 1 é exibido na Figura 7.

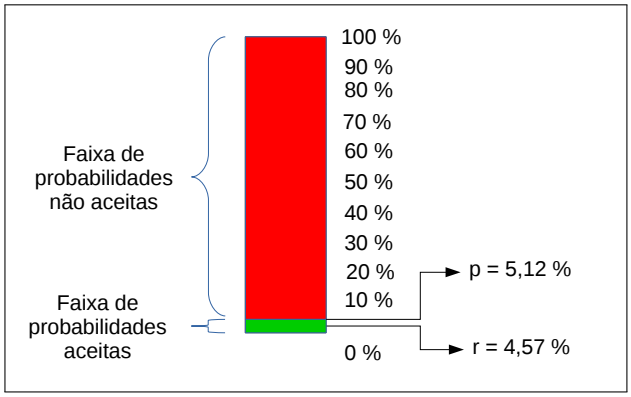
Figura 7 – Caso 1.



Fonte: Elaborado pelo autor

Um nó receptor gerou uma alta probabilidade de encaminhamento p. Entretanto, a distribuição uniforme gerou um número r fora da faixa que indica o encaminhamento. Assim, um nó que deveria fazer o encaminhamento do dado, apesar de ter uma alta probabilidade, faz um descarte. O caso 2 é exibido na Figura 8.

Figura 8 – Caso 2.



Fonte: Elaborado pelo autor

Um nó receptor gerou uma baixa probabilidade de encaminhamento p. Entretanto, a distribuição uniforme gerou um número r dentro da faixa que indica o encaminhamento. Assim, um nó que deveria fazer o descarte do dado, apesar de ter uma baixa probabilidade, faz o encaminhamento.

Para tratar tais casos, o *E-ProbT* faz uso de *thresholds* que se adequam à dinâmica da rede, como definidos no Capítulo 4 da proposta.

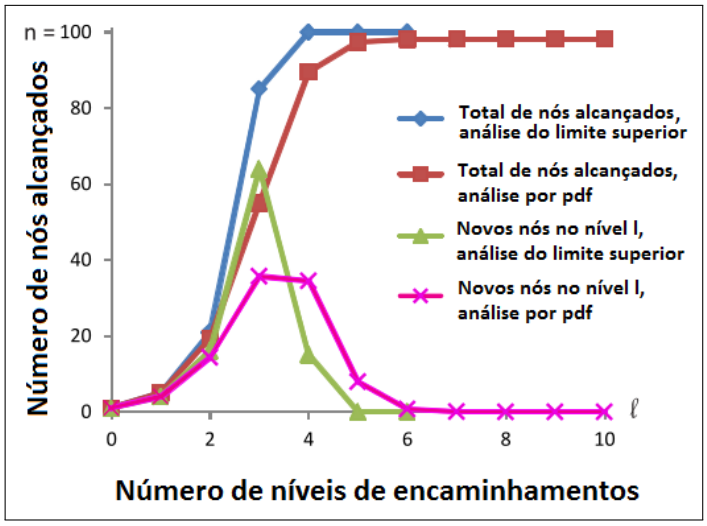
2.5 REGIÃO ALCANÇADA POR BROADCAST

Por definição, o *broadcast* é o ato de entregar um pacote produzido por um determinado nó à todos os outros nós da rede. A solução ótima para o *broadcast* de pacotes é NP-Completa (LIM; KIM, 2001). Assim, existem diversas abordagens com soluções sub-ótimas para o problema, tentando mitigar os efeitos da tempestade de *broadcast*.

Considerando que apenas uma porção da rede está dentro do raio de transmissão do nó emissor, são necessários encaminhamentos para que esse pacote seja entregue aos outros nós da rede. Idealmente, à medida que são feitos novos encaminhamentos, uma maior porção da rede tem acesso ao pacote produzido. De acordo com a decisão de encaminhamento, a porção da rede que vai tendo acesso ao pacote a cada encaminhamento pode variar.

Em (MOSER; MELLIAR-SMITH, 2013) faz-se uma estimação da quantidade de nós que receberam um determinado pacote de acordo com o número de saltos. Um dos resultados obtidos nesse estudo é exibido na Figura 9.

Figura 9 – Número de nós alcançados de acordo com o número de saltos.



Fonte: Figura editada de (MOSER; MELLIAR-SMITH, 2013)

Na Figura 9, é mostrado em azul o gráfico da quantidade de nós alcançados de acordo com o número de saltos. Para o protocolo em questão em (MOSER; MELLIAR-SMITH, 2013), o número de nós alcançados, considerando a Figura 9 e outras produzidas, se aproxima da terceira ou quarta potência do número de saltos.

3 TRABALHOS RELACIONADOS

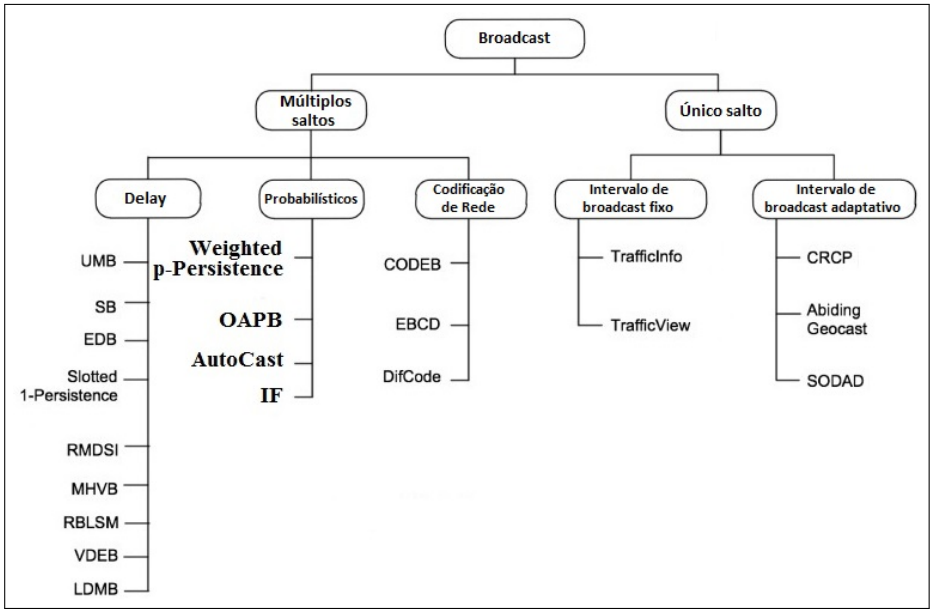
A forma mais simples de pensar em um *broadcast* de informações é definida pelo *Blind flooding* - em uma tradução livre, inundação cega. Quando um nó recebe um pacote de determinado *ID*, ele verifica se o mesmo já foi recebido anteriormente. Se esse pacote já foi recebido, então ele já foi processado anteriormente e não precisa mais ser encaminhado. Assim, realiza-se o descarte do mesmo. Caso esse pacote não tenha sido recebido anteriormente, realiza-se o encaminhamento do mesmo.

Pesquisas mostram que essa técnica de *flooding* é ineficiente porque não possui escalabilidade e produz a não desejada tempestade *broadcast*. Aplicando puramente essa técnica, a redundância dos dados aumenta, criando-se uma grande quantidade de pacotes duplicados na rede (KUMAR; DAVE, 2012). Além disso, ocorre um desperdício da largura de banda da rede, que poderia ser usada por outras aplicações.

Diversas pesquisas são realizadas visando soluções para o problema da tempestade broadcast. (LIM; KIM, 2001) mostra que a solução ótima para esse problema é NP-Completa. Como o uso de uma solução ótima é impraticável, existem diversos trabalhos na literatura visando a mitigação da tempestade *broadcast*, ou seja, amenização dos seus efeitos.

Em (PANICHPAPIBOON; PATTARA-ATIKOM, 2012) é feita uma classificação de protocolos de *broadcast* de informações para VANETs, exibida na Figura 10:

Figura 10 – Uma classificação dos protocolos de *broadcast* para VANETs.



Fonte: Figura edidata de (PANICHPAPIBOON; PATTARA-ATIKOM, 2012)

Nessa classificação, os protocolos são divididos em duas grandes categorias: único-salto (*single-hop*) e múltiplos saltos (*multi-hop*). A categoria de múltiplos saltos é dividida em 3 sub-categorias: protocolos baseados em *delay*, protocolos de codificação de rede e protocolos probabilísticos.

Protocolos baseados em *delay* fazem uso de intervalos de espera antes das retransmissões de pacotes. Ao final do intervalo, verifica-se se algum outro nó já fez o encaminhamento. Assim, caso algum outro nó já tenha feito o encaminhamento desse pacote, um novo *broadcast* pode ser evitado.

Nos protocolos baseados em codificação de rede também se busca a redução da quantidade de informações encaminhadas. Fazendo uso dessa técnica, dois ou mais pacotes podem ser combinados, de forma a criar um novo pacote. Esse novo pacote contém informações que representam os pacotes combinados. Dessa forma, ao invés de encaminhar cada um dos pacotes recebidos, faz-se apenas o encamihamento do pacote fruto da combinação.

Nos protocolos probabilísticos, em geral, uma ou mais características da rede são analisadas. Como resultado dessa análise, é criada uma fórmula probabilística que define o quão bom encaminhador um receptor é. Ao receber um pacote, o nó receptor calcula uma probabilidade de encaminhamento p de acordo com a fórmula definida. Para uso da probabilidade p, é gerado um valor randômico r entre 0 % e 100 %. O valor da probabilidade p é comparado com o valor de r. Se a probabilidade p é maior ou igual à r, prossegue-se com o encaminamento. Caso contrário, descarta-se o pacote.

Os protocolos colocados na categoria de probabilísticos, em (PANICHPAPIBOON; PATTARA-ATIKOM, 2012), são: *Weighted p-persistence*, *Optimized Adaptive Probabilistic Broadcast (OAPB)*, *AutoCast* e *Irresponsible Forwarding (IF)*.

Todos esses protocolos, em face da recepção de um pacote, verificam se o mesmo já foi recebido anteriormente. Se sim, ele é descartado. Caso contrário, gera-se a probabilidade de encaminhamento p. Na sequência, são descritas as fórmulas de cálculo dos mesmos.

3.1 WEIGHTED P-PERSISTENCE

O protocolo ***Weighted p-persistence*** (WISITPONGPHAN et al., 2007) calcula a probabilidade de encaminhamento de um receptor com base na distância do mesmo para a fonte

e no raio de transmissão. Ele faz uso da Fórmula (3.1):

$$p = \frac{Dij}{R} \tag{3.1}$$

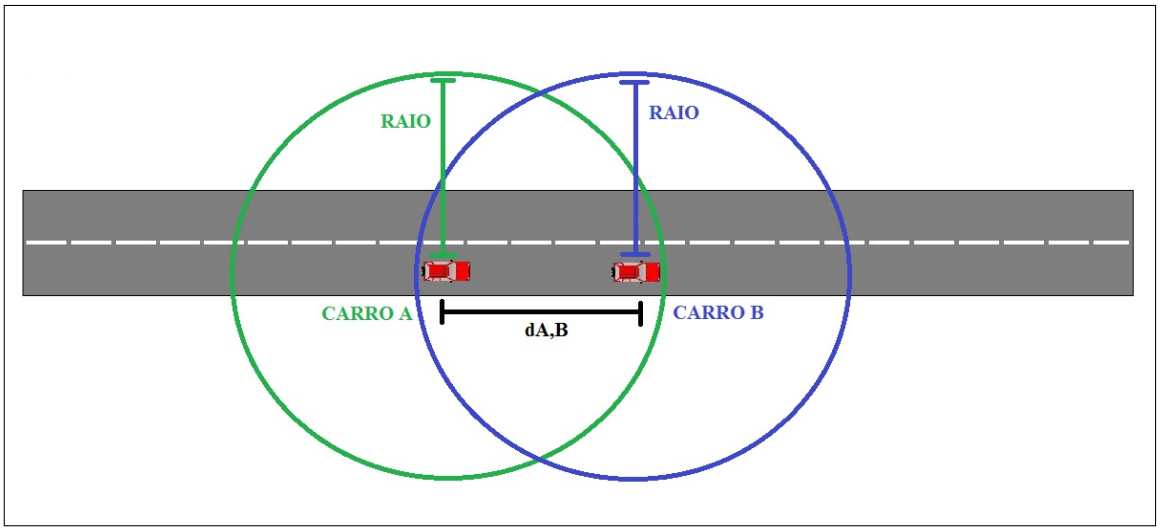
Onde:

Dij = Distância entre a fonte e a origem;

R = Raio de transmissão.

Usando a Fórmula (3.1), são atribuídas as maiores probabilidades de encaminhamento para os nós mais distantes do emissor e vice-versa. Obtem-se a distância *Dij* entre a fonte e a origem calculando a distância entre a posição do nó emissor (informada no pacote) e a posição do receptor (obtida através do GPS). Uma ilustração desse protocolo é exibida na Figura 11.

Figura 11 – Funcionamento do *Weighted p-persistence*.



Fonte: (LIMA et al., 2014)

Na Figura 11, tem-se dois carros A e B, um dentro do raio de transmissão do outro. Suponha que o B receba um pacote enviado pelo no A. O nó B verifica se este pacote já foi recebido anteriormente. Se sim, o pacote é descartado. Caso contrário, a mensagem é encaminhada com probabilidade $p = \frac{dA,B}{Raio}$. Considere um exemplo no qual a distância dA,B é 900 metros. Considere ainda que o raio de transmissão é de 1000 metros. Assim, a probabilidade *p* gerada será dada por (3.2):

$$p = \frac{Dij}{R} = \frac{900}{1000} = 0.9 \tag{3.2}$$

3.2 OPTIMIZED ADAPTIVE PROBABILISTIC BROADCAST (OAPB)

O protocolo ***Optimized Adaptive Probabilistic Broadcast (OAPB)*** (ALSHAER; HORLAIT, 2005) faz uso de zonas de vizinhos para o cálculo da probabilidade. As zonas de vizinhos usadas são:

SH_o = Vizinhos de um salto de um nó de *ID o*. Composta pelos nós localizados dentro do raio de transmissão do nó de *ID o*.

SH_o^2 = Vizinhos de dois saltos de um nó de *ID o*. Composta por todos os vizinhos de 1 salto dos vizinhos do nó de *ID o* que não pertencem à SH_o.

M_{o,c_r} = Vizinhos de dois saltos do nó de *ID o* alcançáveis apenas através do nó c_r.

São definidos então os seguintes elementos:

$$P_{r_o} = \left\{ \begin{array}{cc} \frac{\sum_{r=1}^{N(SH_o)} N(M_{o,c_r})}{N(SH_o)}, & \text{se } \sum_{r=1}^{N(SH_o)} N(M_{o,c_r}) <= N(SH_o) \\ 1, & \text{Caso contrário} \end{array} \right\} \tag{3.3}$$

$$P_{ro_{SH}} = \frac{N(SH_o)}{N(SH_o) + N(SH_o^2)} \tag{3.4}$$

$$P_{ro_{SH^2}} = \frac{N(SH_o^2)}{N(SH_o) + N(SH_o^2)} \tag{3.5}$$

Nas Fórmulas (3.3), (3.4) e (3.5) faz-se uso da notação *N(C)*. *C* denota um conjunto de nós e *N(C)* representa o número de elementos do conjunto *C*. A probabilidade de encaminhamento gerada por esse protocolo é dada por (3.6):

$$p = \frac{P_{r_o} + P_{ro_{SH}} + P_{ro_{SH^2}}}{3} \tag{3.6}$$

Onde:

P_{r_o} = Equação descrita por 3.3;

$P_{ro_{SH}}$ = Equação descrita por 3.4;

$P_{ro_{SH^2}}$ = Equação descrita por 3.5.

Esse protocolo também faz uso de intervalos de tempo de espera antes de repropagar o pacote. O intervalo de tempo de espera é dado por (3.7):

$$\Delta(t) = \Delta(t)max * (1-p) + \delta \tag{3.7}$$

Onde:

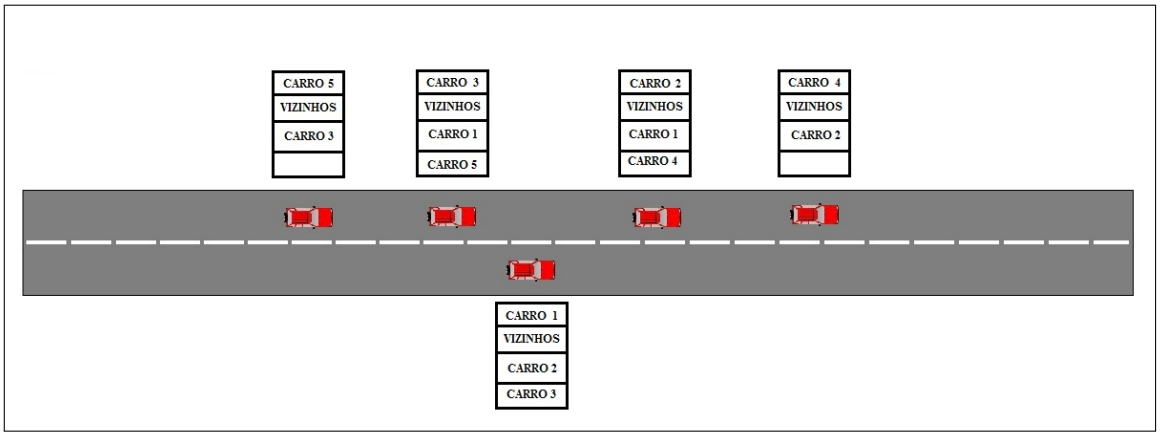
$\Delta(t)max$ = Máximo intervalo de *delay*;

p = Probabilidade de encaminhamento gerada;

δ = Variável randômica.

Considere um cenário nos qual o *OAPB* é aplicado, disponível em (LIMA et al., 2014). Ele é ilustrado na Figura 12.

Figura 12 – Funcionamento do *OAPB* - Cenário 1.



Fonte: (LIMA et al., 2014)

Os veículos e as suas tabelas de vizinhos são apresentadas. O veículo 1 recebe uma mensagem enviada pelo veículo 3. Caso seja uma mensagem não recebida anteriormente, ele deve gerar a probabilidade de encaminhamento (LIMA et al., 2014). Neste caso:

SH_1 = {Carro 2, Carro 3} $\Longrightarrow N(SH_1) = 2$

SH_1^2 = {Carro 4, Carro 5} $\Longrightarrow N(SH_1^2) = 2$

M_{1,c_2} = {Carro 4} $\Longrightarrow N(M_{1,c_2}) = 1$

M_{1,c_3} = {Carro 5} $\Longrightarrow N(M_{1,c_3}) = 1$

$\sum_{r=1}^{N(SH_1)} N(M_{1,c_r}) <= N(SH_1)$, pois $N(M_{1,c_2}) + N(M_{1,c_3}) = 1+1 = 2 <= N(SH_1) = 2$

Então, $P_{r_1} = \frac{\sum_{r=1}^{N(SH_1)} N(M_{1,c_r})}{N(SH_1)} = \frac{2}{2} = 1$

$P_{r1_{SH}} = \frac{N(SH_1)}{N(SH_1)+N(SH_1^2)} = \frac{2}{2+2} = \frac{2}{4} = 0.5$

$P_{r1_{SH^2}} = \frac{N(SH_1^2)}{N(SH_1)+N(SH_1^2)} = \frac{2}{2+2} = \frac{2}{4} = 0.5$

Assim, $p = \frac{P_{ro}+P_{ro_{SH}}+P_{ro_{SH^2}}}{3} = \frac{1+0.5+0.5}{3} = 0.666$

Assim, p = 0.666. Isso significa que a mensagem será encaminhada com probabilidade p igual à 0.666.

3.3 AUTOCAST

O protocolo ***AutoCast*** (WEGENER et al., 2007) calcula a probabilidade de encaminhamento com base no número de veículos da vizinhança alcançáveis através de 1 salto. Isso é feito para modelar as porções da rodovia com diferentes densidades de vizinhos. Ele faz uso da Fórmula (3.8):

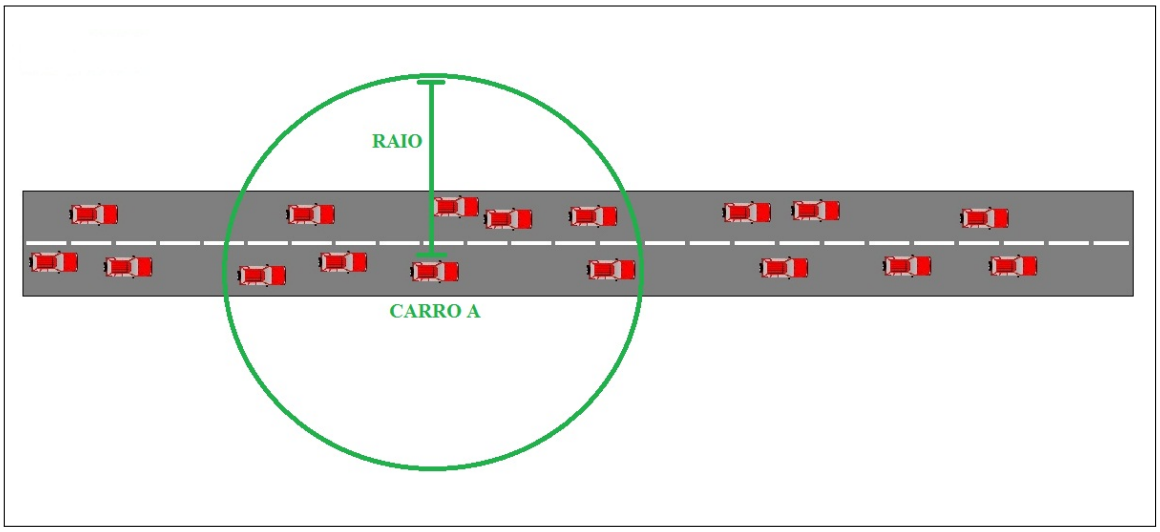
$$p = \frac{2}{Nh * 0.4} = \frac{5}{Nh} \tag{3.8}$$

Onde:

Nh = Número de veículos da vizinhança alcançáveis através de 1 salto.

Ele define maiores probabilidades para nós com pequenas quantidades de vizinhos. Em (PANICHPAPIBOON; PATTARA-ATIKOM, 2012) é feita uma crítica à esse protocolo, pelo fato de não ser indicado o valor de p quando Nh for menor que 5 (o que produziria valores de probabilidade maiores que 1). Um cenário de aplicação do Autocast, fornecido em (LIMA et al., 2014), é exibido na Figura 13:

Figura 13 – Funcionamento do *AutoCast*.



Fonte: (LIMA et al., 2014)

Neste cenário, o número de vizinhos do carro A é 7. A probabilidade de encaminhamento será dada por (3.9):

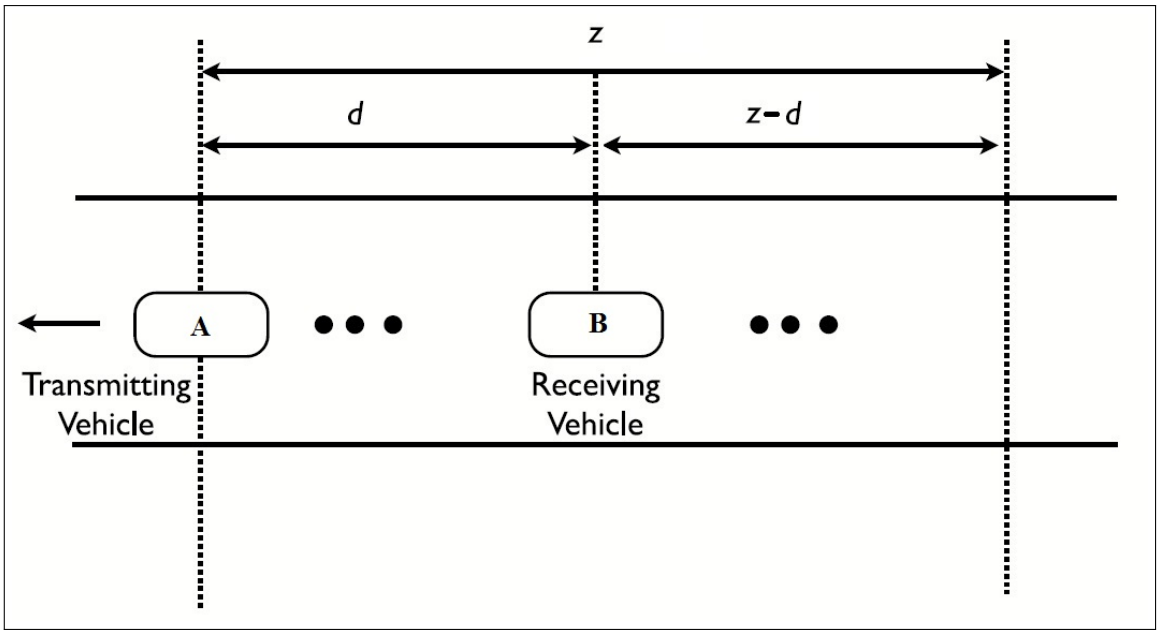
$$p = \frac{5}{Nh} = \frac{5}{7} = 0.714 \tag{3.9}$$

3.4 IRRESPONSIBLE FORWARDING (IF)

O ***Irresponsible Forwarding*** (PANICHPAPIBOON; FERRARI, 2008) leva em consideração a distância entre emissor e receptor bem como a distribuição estatística dos veículos para o cálculo da probabilidade de encaminhamento. Ele é executado da seguinte forma: uma vez recebido um pacote, o nó receptor avalia a probabilidade de haver um outro veículo mais distante para o emissor com uma melhor probabilidade de encaminhamento. Se houver uma alta probabilidade de existência de tal veículo (de acordo com a distância e a densidade veicular) o nó escolhe, "irresponsavelmente", não encaminhar o pacote. Caso contrário, ele faz o encaminhamento (PANICHPAPIBOON; FERRARI, 2008). Dessa escolha de não encaminhar o pacote deliberadamente, os autores atribuíram o nome do protocolo (*Irresponsible Forwarding*).

Considere um veículo B à uma distância d do transmissor A e um raio de transmissão igual a z, como ilustrado na Figura 14.

Figura 14 – Funcionamento do *Irresponsible Forwarding*.



Fonte: Figura edidata de (PANICHPAPIBOON; FERRARI, 2008)

De acordo com este protocolo, o nó B deve retransmitir o pacote apenas se a chance de encontrar outro nó na porção restante do raio $(z-d)$ for pequena. Assim, é usada a fórmula (3.10):

$$p = e^{-\frac{Ps*(z-d)}{c}} \tag{3.10}$$

Onde:

e = Número de Neper $\cong$ 2.718281;

Ps = Densidade veicular;

d = Distância entre a fonte e a origem;

z = Raio de transmissão;

c = Coeficiente de valor maior ou igual a 1 para regular a curva de probabilidade.

De acordo com a Fórmula (3.10), temos que a probabilidade de encaminhamento do pacote aumenta à medida que a distância entre o emissor e o receptor aumenta e vice-versa. A probabilidade também aumenta quando se tem menores valores de densidades veiculares. O coeficiente c é utilizado para modelar a curva de probabilidade, de forma que incrementos em c aumentam o valor da probabilidade.

3.5 STATISTICAL LOCATION-ASSISTED BROADCAST PROTOCOL (SLAB)

Protocolos de *broadcast* baseados em probabilidade devem gerar um valor probabilístico que modele a sua tendência de encaminhamento de pacotes. Se esse valor de probabilidade for maior que um valor de *threshold*, o protocolo decide por encaminhar um dado pacote. A chave para qualquer protocolo probabilístico é o valor de *threshold* (SLAVIK; MAHGOUB, 2011). Se esse valor é muito alto, a conectividade será degradada. Se esse valor é muito baixo, o protocolo não evitará que muitos nós façam o encaminhamento do pacote - o que pode levar ao problema de tempestade *broadcast*.

Este protocolo faz uso do método distância para a média. Este método usa a ideia base de que um nó deve encaminhar um pacote se uma pequena porção da sua área de transmissão está coberta por outros nós. Para tanto, ele calcula uma média da posição dos vizinhos através da

Fórmula (3.11).

$$(\overline{x},\overline{y}) \;=\; (\frac{1}{n}\sum_{i=1}^{n} x_i, \frac{1}{n}\sum_{i=1}^{n} y_i) \tag{3.11}$$

A variável distância para a média, chamada de M, é a distância normalizada do nó para a média de posição dos vizinhos. Se o nó está na posição (x,y), M é calculado através da Fórmula (3.12).

$$M \;=\; \frac{1}{r}\sqrt{(x-\overline{x})^2 \;+\; (y-\overline{y})^2} \tag{3.12}$$

O algoritmo para o método distância para a média é o seguinte:

1. Quando uma mensagem é recebida, ela é armazenada e configura-se um temporizador de *backoff* de duração $t = T_{MAX}(1-\frac{d}{r})$. Nesta fórmula, d é a distância para o emissor e r é o valor do raio;
2. Se uma mensagem é recebida durante o *backoff*, vai para o passo 1;
3. Quando o temporizador expira, calcula-se M e faz o encaminhamento se $M > M_c$. M_c é o valor de *threshold*.

O *threshold* é calculado através de um *perceptron* multi-camadas (Multi-Layer Perceptron - MLP). Um perceptron multi-camadas é um modelo de uma rede neural (PAL; MITRA, 1992). Ele consiste de múltiplas camadas de nós (representações de neurônios) que interagem usando conexões ponderadas. A primeira camada do perceptron é chamada de camada de entrada. Nessa camada são definidas uma ou mais entradas. O MLP usado é definido em função de 3 entradas:

1. Número de vizinhos;
2. Estatística Quadrática Q;
3. Fator de desvanecimento k de Rice (Rician Fading Factor k).

Esse trabalho mostrou que, trabalhando o valor de *threshold*, pode-se ter uma melhora significativa da performance do protocolo. Os protocolos tradicionais não indicam como fazem a geração do valor de threshold, levando a crer que eles são gerados de acordo com a distribuição uniforme.

3.6 DELAY-AWARE ROUTING BASED ON GAME-THEORY (DARGT)

O Protocolo DARGT é aplicado para roteamento em VANETs (SU; WANG, 2013). Ele faz uso de um jogo, baseado em recompensas e penalidades. Este jogo foi adaptado para broadcast e utilizado nesta proposta. Assim, ele se encontra entre os trabalhos relacionados. O seu funcionamento é dividido em duas fases:

1. Seleção do caminho;
2. Jogo de encaminhamento.

Na fase 1, deve ser selecionado o caminho através do qual ocorrerá o fluxo de dados da fonte para o destino. O protocolo AODV tradicional seleciona o melhor caminho de acordo com o menor número de saltos. Isso pode provocar grandes atrasos. Para evitar esses atrasos de transmissão, o DARGT calcula o custo do caminho considerando as seguintes características: carga do sistema, o número de saltos, um valor de confiança e o *delay* de cada caminho (SU; WANG, 2013).

O termo carga do sistema se refere à capacidade de fila de cada nó. Se um nó tem muitos pacotes esperando na fila, isto significa que ele tem uma alta carga de sistema. O valor de confiança de um nó é um valor de porcentagem e indica o nível do confiabilidade de um nó. Esse valor é calculado na fase 2.

Na fase 2, é aplicada a Teoria dos Jogos para o encaminhamento de pacotes pelo caminho selecionado. O jogo do encaminhamento pode ser expresso como:

1. Jogadores:
    a) Transmissor: nó que envia o pacote;
    b) Vizinhos: conjunto de nós que recebem o pacote e tomam uma decisão de encaminhamento.
2. Estratégia:
    a) Estratégias do transmissor:
        - Recompensar nós encaminhadores;
        - Penalizar os nós não encaminhadores.
    b) Estratégias dos vizinhos:
        - Encaminhar o pacote;
        - Descartar o pacote.
3. Utilidade:

A utilidade no jogo do encaminhamento é um múltiplo da confiança que o nó emissor

tem no nó encaminhador. A confiança é um valor real, pertencente ao intervalo [0,1]. Cada nó possui uma tabela de confiança em vizinhos. À medida que as decisões sobre encaminhamentos são realizadas, tais valores vão sendo alterados.

A medida que um nó vai encaminhando pacotes, a confiança para com esse nó, por parte de outros, irá aumentar. A medida que um nó vai descartando pacotes, a confiança para com esse nó, por parte de outros, irá diminuir. Com essa estratégia, espera-se diminuir o comportamento egoísta dos nós (esperar que algum outro nó faça o encaminhamento).

De acordo com o valor da confiança em um nó é calculado o valor da utilidade de pacote. Esse valor depende do intervalo de tempo que o encaminhador demora para reenviar o pacote.

- Entre 0 e 5 milisegundos: utilidade do pacote: $= + 2 * \alpha * T_f$;
- Acima de 5 e menor ou igual a 10 milisegundos: utilidade do pacote: $= + \alpha * T_f$;
- Acima de 10 e menor ou igual a 15 milisegundos: utilidade do pacote: $= 0$;
- Acima de 15 milisegundos: utilidade do pacote: $= - \alpha * T_f$.

O valor de utilidade selecionado será somado ao valor da confiança no nó encaminhador.

O valor de confiança é usado para 3 funções:

1. A confiança é um parâmetro de cálculo do custo de um caminho, usado na fase 1;
2. A confiança é usada para cálculo da recompensa de um pacote, na fase 2;
3. A confiança é usada no cálculo dos nós para saber se devem ou não encaminhar um pacote: ao receber um pacote, um nó verifica se é verdadeira a expressão matemática:

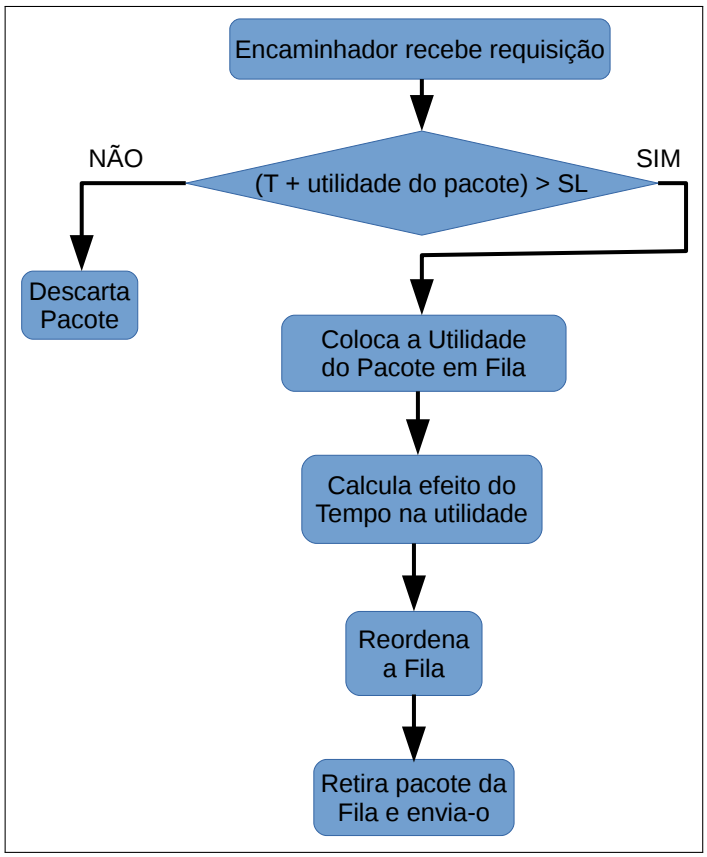
$$\text{Confiança} + \text{Utilidade do Pacote} > \text{Carga do Sistema}$$

Se verdadeira, coloca-se o pacote na fila de envios, verifica qual a posição de saída dos pacotes na fila que entrega o maior valor de retorno, reordena os pacotes na fila de acordo com a melhor soma de retorno e faz o envio dos pacotes. Caso contrário, descarta o pacote. Esse procedimento é exibido na Figura 15.

3.7 PROBT

O Protocolo *ProbT* (LIMA et al., 2015) é o principal trabalho usado como base para a elaboração do *E-ProbT*. Ele é um protocolo probabilístico e temporal para a mitigação do problema *broadcast* storm em VANETs. Por ser um protocolo probabilístico, deve gerar uma probabilidade para encaminhamento de pacotes. Tal probabilidade usa como base a quantidade

Figura 15 – Fluxograma.

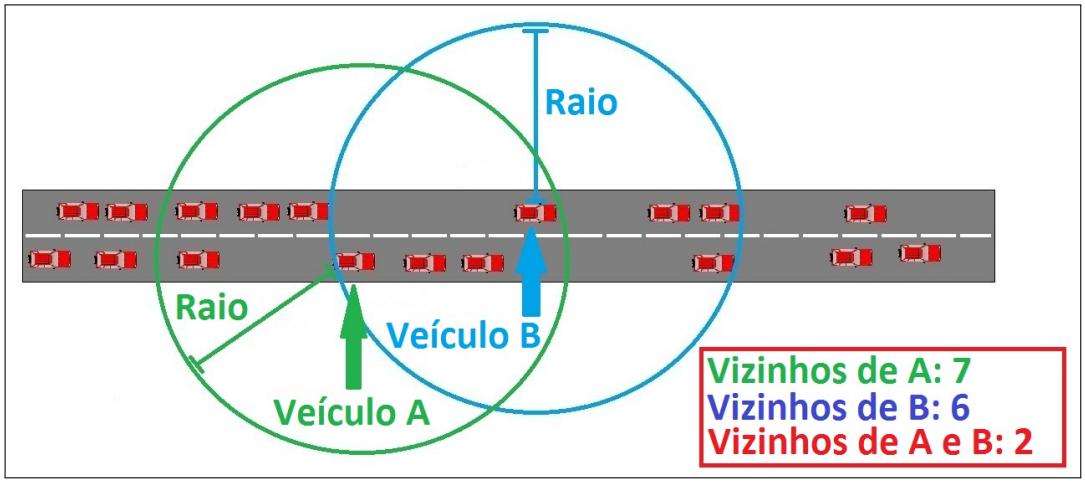


Fonte: Figura editada de (SU; WANG, 2013)

de vizinhos em comum bem como a distância entre emissor e receptor.

A quantidade de vizinhos em comum entre emissor e receptor é exibida na Figura 16.

Figura 16 – Quantidade de vizinhos em comum.



Fonte: Figura edidata de (LIMA et al., 2015)

A probabilidade associada à quantidade de vizinhos em comum é modelada através da Fórmula (3.13):

$$p1 = \frac{qNeigh - q}{qNeigh} \tag{3.13}$$

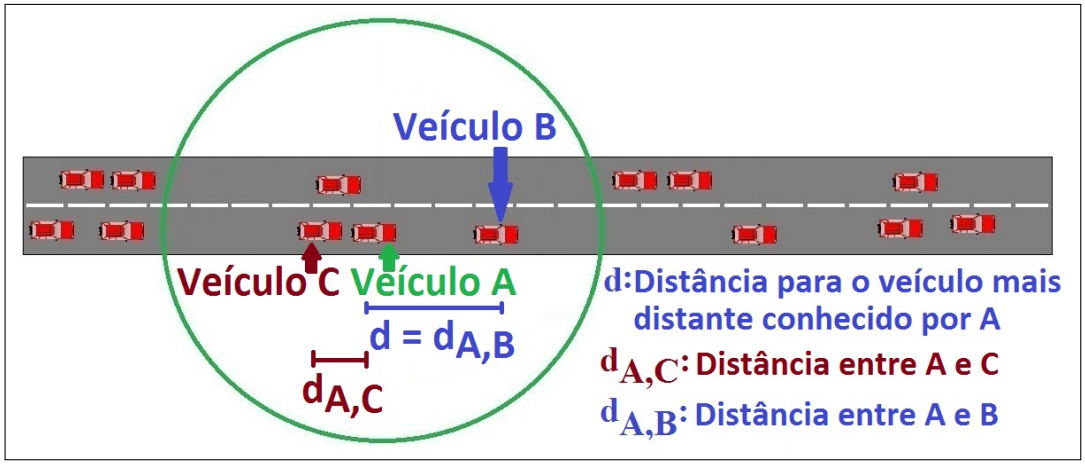
Onde:

qNeigh = Quantidade de vizinhos que o nó receptor possui;

q = Quantidade de vizinhos em comum entre a fonte e o receptor.

A distância entre emissor e receptor é exibida na Figura 17.

Figura 17 – Distância entre emissor e receptor.



Fonte: Figura editada de (LIMA et al., 2015)

A probabilidade associada à distância entre emissor e receptor é modelada através da Fórmula (3.14):

$$p2 = (1 - \frac{|dLastNeigh - dToSource|}{max(dLastNeigh, dToSource)}) \tag{3.14}$$

Onde:

dLastNeigh = Distância para o vizinho mais distante conhecido pela fonte;

dToSource = Distância do receptor para o nó fonte.

Por fim, define-se a probabilidade de encaminhamento do ProbT de acordo com a

Fórmula (3.15).

$$p = \alpha * p1 + (1 - \alpha) * v * p2 \tag{3.15}$$

Onde:

α = Fator de prioridade pertencente ao intervalo [0.0, 1.0] em $\mathbb{R}$.

$$v = \begin{cases} 1 & \text{, se o nó fonte está à uma distância maior ou igual à 0.9 * TransmissionRange} \\ 0 & \text{, caso contrário} \end{cases}$$

TransmissionRange = 1000 metros.

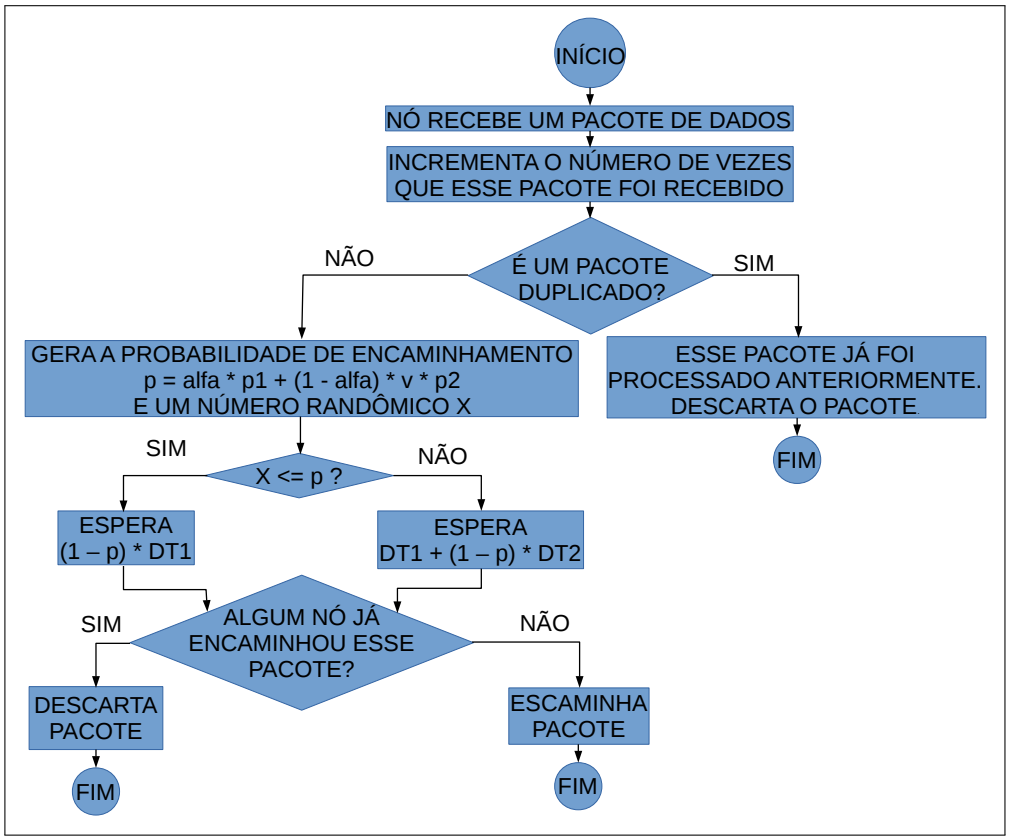
O fator de prioridade α é usado para atribuir pesos aos critérios analisados. A variável v é utilizada para aplicar uma setorização no raio de transmissão dos nós. Os nós que estiverem à uma distância menor que 0.9 * *TransmissionRange* terão como valor zero a probabilidade associada à distância entre a fonte e o receptor. Esse procedimento diminui a probabilidade de encaminhamentos realizados por nós próximos do emissor, como desejado.

A recepção e tratamento dos pacotes de dados são descritos pelo fluxograma exibido na Figura 18.

Ao receber um pacote, incrementa-se a quantidade de vezes que ele foi recebido. Então, verifica se o pacote é repetido. Se sim, ele é descartado. Caso contrário, gera uma probabilidade de encaminhamento p e um número randômico x. Se o valor de x for menor ou igual ao valor de p, o nó decide se voluntariar para o encaminhamento. Caso contrário, o nó decide esperar que algum outro nó faça o encaminhamento. Nos dois casos, são usados períodos de espera. Eles são usados para evitar transmissões ao mesmo tempo, bem como redundância de transmissões. Tais esperas são menores para os nós com maiores valores de probabilidade. Isso faz com que os melhores nós façam o encaminhamento. Nos dois casos, ao final do período de espera, verifica se algum nó fez o encaminhamento. Se sim, o pacote é descartado. Caso contrário, o nó faz o encaminhamento.

Os protocolos probabilísticos listados a seguir terão o seu desempenho comparado com o do *E-ProbT*: *Blind Flooding*, *Weighted p-persistence*, *AutoCast*, *Irresponsible Forwarding* e *ProbT*.

Figura 18 – Fluxograma de recepção de mensagens de dados.



Fonte: adaptado de (LIMA et al., 2014)

4 PROPOSTA

Este capítulo apresenta o protocolo proposto E-ProbT e a sua modelagem matemática. Ele faz uso de duas classificações:

1. Classificação do nó emissor quanto à confiança;
2. Classificação do nó receptor quanto à probabilidade de encaminhamento.

Para a classificação do nó emissor quanto à confiança, será aplicada a Teoria dos Jogos (WATSON, 2007). Para a classificação do nó receptor quanto à probabilidade de encaminhamento, será usada uma média móvel exponencial ponderada (ALVES et al., 2012), (KUROSE; ROSS, 2010). Por fim, a decisão sobre encaminhamento de pacotes será tomada de acordo com a combinação de:

1. Classificação do nó emissor quanto à confiança;
2. Classificação do nó receptor quanto à probabilidade de encaminhamento;
3. Fator de benevolência;
4. Fórmulas probabilísticas aplicadas no *ProbT* (LIMA et al., 2015);
5. Mecanismos de temporização aplicados no *ProbT* (LIMA et al., 2015).

4.1 CLASSIFICAÇÃO DO NÓ EMISSOR QUANTO À CONFIANÇA

A confiança é uma característica descritora de um vizinho de acordo com as suas decisões sobre reencaminhamento. A cada momento, os nós tomam decisões sobre encaminhamento de pacotes, podendo se decidir pelo encaminhamento ou pelo descarte. Um nó emissor de um pacote tem uma tabela de vizinhança. Ao emitir um pacote, ele verifica quais nós da sua tabela de vizinhança fizeram o encaminhamento. Essa verificação é feita analisando os pacotes recebidos após a emissão. Aqueles que encaminharam o pacote serão recompensados, ganhando um incremento no valor de confiança. Aqueles que não encaminharam serão penalizados, sofrendo um decréscimo no valor de confiança.

Considerando a dinâmica de uma VANET, nós podem tornar-se bons ou maus encaminhadores. Assim, os nós vizinhos terão a sua avaliação quanto à confiança alterada. Por exemplo: se um nó B era um bom encaminhador para o nó A, existia um valor alto de confiança associado a B na tabela de vizinhança de A. Entretanto, se B torna-se um mau encaminhador, ele não encaminha mais os pacotes de A. Assim, o valor de confiança de A em B começa a ser reduzido. Em determinado momento, após vários decréscimos realizados, o nó B deixa de ser

classificado como um bom encaminhador, passando a ser classificado como mau encaminhador.

Por outro lado, se um nó *B* era um mau encaminhador para o nó *A*, existia um valor baixo de confiança associado à *B* na tabela de vizinhança de *A*. Entretanto, se *B* torna-se um bom encaminhador, ele passa a encaminhar os pacotes de *A*. Assim, o valor de confiança de *A* em *B* começa a ser incrementado. Em determinado momento, após vários acréscimos realizados, o nó *B* deixa de ser classificado como um mau encaminhador, passando a ser classificado como bom encaminhador.

Deve-se deixar claro que um nó *B* pode decidir por não encaminhar um pacote de *A*, por não ser um bom encaminhador para *A* de acordo com as suas fórmulas probabilísticas. Nesse caso, o pacote deixa de ser encaminhado não por uma decisão visando prejudicar a comunicação do outro nó, e sim, por se considerar um mal encaminhador, de forma que o seu encaminhamento seria prejudicial para a rede. Assim, os mecanismos de recompensa e penalidade são aplicados no mesmo sentido. Decrementos no valor de confiança em algum vizinho são feitos porque naquele momento esse vizinho não era um bom encaminhador. Incrementos no valor de confiança em algum vizinho são feitos porque naquele momento esse vizinho era um bom encaminhador. Dessa forma, a aplicação do mecanismo de recompensa/penalidade é feita para uma boa descrição da vizinhança.

Os nós possuem uma tabela de vizinhança. Nessa tabela existe uma entrada associada à confiança para cada vizinho. A entrada na tabela associada à confiança é preenchida através da aplicação da Teoria dos Jogos (WATSON, 2007). Os valores de confiança nos vizinhos serão usados na decisão sobre o encaminhamento de pacotes. Será feita uma adaptação do jogo aplicado no trabalho relacionado *DARGT* (SU; WANG, 2013). No *DARGT*, o jogo foi aplicado para a roteamento. No *E-ProbT*, ele será adaptado para *broadcast*.

4.1.1 Definição formal do jogo

O jogo utilizado nesse trabalho tem a seguinte definição formal:

1. Jogadores:
    a) Transmissor: nó que envia o pacote;
    b) Vizinhos: conjunto de nós que recebem o pacote e tomam uma decisão de encaminhamento.
2. Estratégia:
    a) Estratégias do transmissor:

- Recompensar os nós encaminhadores;
- Penalisar os nós não encaminhadores.

b) Estratégias dos vizinhos:

- Encaminhar o pacote;
- Descartar o pacote.

A confiança é um valor entre 0 e 100 %. Se um nó B encaminha um pacote enviado por um nó A, então a confiança de A em B é incrementada de acordo com um valor de recompensa. Todos os vizinhos que não encaminharem uma mensagem recebida receberão uma penalidade no valor da confiança. As recompensas e penalidades são definidas na Tabela 1.

Tabela 1 – *Valores de Recompensa e Penalidade*

Tipo	Valor
Recompensa	REC
Penalidade	$\frac{PEN}{nSaltos^3}$

Para uma compreensão do jogo, devem ser analisados 2 cenários de aplicação:

- Primeiro salto;
- Dois ou mais saltos.

No primeiro salto, inicialmente, apenas o nó emissor tinha acesso ao dado. Após a transmissão, os vizinhos receberão o pacote em questão. Nesse momento, cada nó deve tomar a sua decisão sobre o encaminhamento. Os nós encaminhadores serão recompensados, tendo um incremento de REC no valor da confiança das entradas da tabela do nó emissor associadas aos mesmos. Os nós não encaminhadores serão penalisados, tendo um decremento de PEN no valor da confiança das entradas da tabela do nó emissor associadas aos mesmos.

Enquanto o jogo é aplicado, um nó emissor atualiza a confiança que ele possui nos vizinhos. Os maus encaminhadores, à medida que tomarem sucessivas decisões por descarte, terão baixos valores de confiança associados aos mesmos na tabela do emissor. Os bons encaminhadores, à medida que tomarem sucessivas decisões por encaminhamento, terão altos valores de confiança associados aos mesmos na tabela do emissor. Assim, tem-se uma tabela de vizinhança que permite analisar cada vizinho com base no seu padrão de encaminhamento.

A partir do segundo salto, o raio de transmissão de um nó encaminhador é composto por nós que já tiveram acesso ao dado bem como nós que ainda não o receberam. Por esta razão, em face da recepção de um pacote duplicado, os nós receptores o descartam para evitar retransmissões redundantes. Nesses casos, não seria justo aplicar um mecanismo de penalidade

com o mesmo rigor do primeiro salto, pois estariam sendo penalisados nós que tomaram uma boa decisão para a rede. Entretanto, devem ser penalisados aqueles que o receberam pela primeira vez e ainda assim não o encaminharam. Deve então ser aplicada uma penalização que não seja tão severa, bem como não seja tão branda. Considerando os resultados obtidos em (MOSER; MELLIAR-SMITH, 2013), faz-se uma suavização da penalidade aplicada aos nós não encaminhadores de acordo com o número de saltos.

Assim, se o pacote tem uma pequena quantidade de saltos, uma pequena porção da rede já o recebeu. Por outro lado, se o pacote tem uma grande quantidade de saltos, uma grande porção da rede já o recebeu. Assim, a punição aplicada neste trabalho será diminuída de acordo com o número de saltos. Baseado nos resultados de (MOSER; MELLIAR-SMITH, 2013) e em simulações realizadas do trabalho proposto, a punição aplicada é dividida pela terceira potência do número de saltos, conforme descrita na Tabela 1.

Considerando o valor de confiança em um vizinho armazenado na tabela de vizinhança, deve existir um mecanismo que os classifique. Esse mecanismo é definido na regra de classificação com base na confiança.

4.1.2 Regra de classificação com base na confiança

Existe uma regra de classificação de vizinhos com base no valor da confiança nos mesmos. As faixas de valores dessa classificação são exibidas na Tabela 2.

Tabela 2 – *Faixas de valores de Confiança*

Intervalo	Classificação
$0\ \% \leq$ valor $\leq k\ \%$	Egoísta
$k\ \% <$ valor $\leq 100\ \%$	Altruísta

Se o valor de confiança em um determinado vizinho estiver entre 0 e K %, ele é classificado como Egoísta. Se o valor de confiança for maior que K %, tal vizinho é classificado como Altruísta. Essa classificação é usada no processo de decisão de encaminhamento de pacotes. Os nós classificados como Altruístas serão recompensados por outros nós, tendo uma maior tendência para ter os seus pacotes encaminhados. Assim, altos valores de K produzem um maior rigor na decisão de encaminhamento. Baixos valores de K produzem uma maior complacência nos encaminhamentos. De acordo com o rigor desejado, deve-se escolher o valor de K.

4.2 CLASSIFICAÇÃO DO NÓ RECEPTOR QUANTO À PROBABILIDADE DE ENCAMINHAMENTO

Neste trabalho, um nó também será classificado de acordo com o valor da probabilidade de encaminhamento (3.15) do *ProbT*, descrita na seção (3.7) dos trabalhos relacionados (Capítulo 3). Valores de probabilidade pertencem ao intervalo $[0.0\ \%, 100.0\ \%]$ em $\mathbb{R}$. Esse intervalo será dividido em faixas. De acordo com a faixa de valores na qual a probabilidade estiver, será obtida a classificação do nó. As faixas de valores são exibidas na Tabela 3.

Tabela 3 – *Faixas de valores de Probabilidade*

Intervalo	Classificação
$0\ \% < \text{valor} \leq V1\ \%$	Ruim
$V1\ \% < \text{valor} \leq V2\ \%$	Regular
$V2\ \% < \text{valor} \leq 100\ \%$	Ótimo

V1 e *V2* são usados como *thresholds*. Eles são gerados de forma dinâmica pelo emissor e são informados aos vizinhos através do pacote a ser transmitido. Ao receber o pacote, o vizinho gera a sua probabilidade de encaminhamento e verifica em qual intervalo ela se encontra. Se a probabilidade for maior ou igual a 0 % e menor ou igual a um certo valor *V1* %, o vizinho será classificado como Ruim. Se a probabilidade for maior que *V1* % e menor ou igual a um certo valor *V2* %, o vizinho será classificado como Regular. Se a probabilidade for maior que *V2* % e menor ou igual a 100 %, o vizinho será classificado como Ótimo. Essa classificação será usada na decisão de encaminhamento.

No *E-probT*, os nós classificados como ótimos terão uma forte tendência para encaminhamento do pacote recebido. Assim, o *threshold V2* é um dos valores cruciais para um bom funcionamento do protocolo, uma vez que ele é o limiar das classificações como ótimo e regular.

Para satisfazer os objetivos do *broadcast*, apenas os melhores vizinhos devem encaminhar o pacote (na melhor das hipóteses apenas o melhor). Considerando a presença de nós dentro do raio de transmissão do emissor, alguns estarão em uma ótima posição para encaminhamento, outros não. O quão bom encaminhador um nó é será definido pela aplicação da Fórmula probabilística do *ProbT* (3.15), como descrito na Seção (3.7) dos trabalhos relacionados. Assim, os encaminhadores ruins terão baixos valores de probabilidades, os regulares terão valores medianos e os ótimos terão altos valores de probabilidade.

O *threshold V2* indicará quais são os melhores encaminhadores. Com esse objetivo,

antes de enviar um pacote, o nó emissor fará uma estimativa dos valores de probabilidade de encaminhamento dos vizinhos. De posse dessa estimativa, ele cria um valor de *threshold V2* com o qual apenas os melhores vizinhos tenham uma tendência à encaminhar o pacote.

Como o nó emissor está trabalhando com estimativas, não se pode garantir que os nós receptores terão tais valores de probabilidade. Assim, o nó emissor calculará uma média dos melhores valores de probabilidade estimados. Fazendo uma média dos melhores valores evita-se que nós classificados como ruins ou regulares tenham fortes tendências para encaminhamento. Além disso, se adaptará de forma dinâmica às variações da rede veicular.

A quantidade de amostras de melhores valores de probabilidade estimadas usadas para o cálculo da média é um importante fator. Se usadas muitas amostras, a média tende a cair, possibilitando a realização de encaminhamentos por parte de nós não desejados com menores valores de probabilidade. Se usadas poucas amostras, a média de estimativas pode ficar superior aos maiores valores de probabilidade reais da rede, uma vez que são usadas estimativas. Assim, são escolhidos os três maiores valores de estimativas de probabilidade de encaminhamento de vizinhos.

Visando uma melhor caracterização do valor de *threshold V2*, ele passará por um processo de média móvel exponencial ponderada. Assim, também serão levados em consideração valores antigos e recentes da média de estimativas de melhores probabilidades. Através desse processo, tenta-se eliminar possíveis disparidades de valores bem como possíveis imperfeições na estimativa realizada. Assim, aplica-se uma suavização da curva de valores calculados, semelhante ao processo que acontece para cálculo do timeout do TCP (KUROSE; ROSS, 2010), como descrito na Seção (2.3.1).

Os vizinhos com as melhores probabilidades de encaminhamento são aqueles mais distantes. Isso se deve à Fórmula probabilística do *ProbT*. Assim, para ganho de performance e evitar cálculos desnecessários são estimadas apenas as probabilidades dos vizinhos mais distantes. Tais vizinhos mais distantes são obtidos através da aplicação do mecanismo de predição de posição, fornecido pelo *ProbT*. Tal mecanismo usa como base os dados dos vizinhos que foram salvos na tabela de vizinhança (posição, velocidade, aceleração e tempo), obtidos através de mensagens *HELLO* que são enviadas por cada nó.

De posse dos três vizinhos mais distantes, faz-se a estimativa das probabilidades de encaminhamento de cada um. Na sequência, realiza-se a média desses três valores. Essa média será denominada de $SampleThreshold$. À partir de $SampleThreshold$, de forma semelhante ao

TCP, calcula-se o valor de *threshold* estimado, chamado de $EstimatedThreshold$. Esse cálculo será de acordo com a Fórmula (4.1):

$$EstimatedThreshold = (1-\beta) * EstimatedThreshold + \beta * SampleThreshold \quad (4.1)$$

Onde:
EstimatedThreshold = *Threshold* estimado;
SampleThreshold = Valor da amostra da média de estimativas de probabilidades;
β = Uma constante usada para atribuição de pesos.

O valor da constante β utilizada é o mesmo valor recomendado para o TCP. Assim, $\beta = 1/8 = 0,125$ (JACOBSON, 1988) (KUROSE; ROSS, 2010). Além de estimar o *threshold*, deve ser feita uma estimativa da sua variação. Essa estimação é feita com base na Fórmula (4.2):

$$DevThreshold = (1-\gamma) * DevThreshold + \gamma * |SampleThreshold - EstimatedThreshold| \quad (4.2)$$

Onde:
DevThreshold = Variação de *threshold* estimada;
SampleThreshold = Valor da amostra da média de estimativas de probabilidades;
EstimatedThreshold = *Threshold* estimado;
γ = Uma constante usada para atribuição de pesos.

O valor da constante γ utilizada é o mesmo valor recomendado para o TCP. Assim, $\gamma = 1/8 = 0,125$ (JACOBSON, 1988) (KUROSE; ROSS, 2010). Por fim, para cálculo do *threshold V2*, devem ser considerados em uma fórmula a estimação e a variação do *threshold*. Esse cálculo é feito como base na Fórmula (4.3):

$$Threshold\ V2 = EstimatedThreshold + DevThreshold \quad (4.3)$$

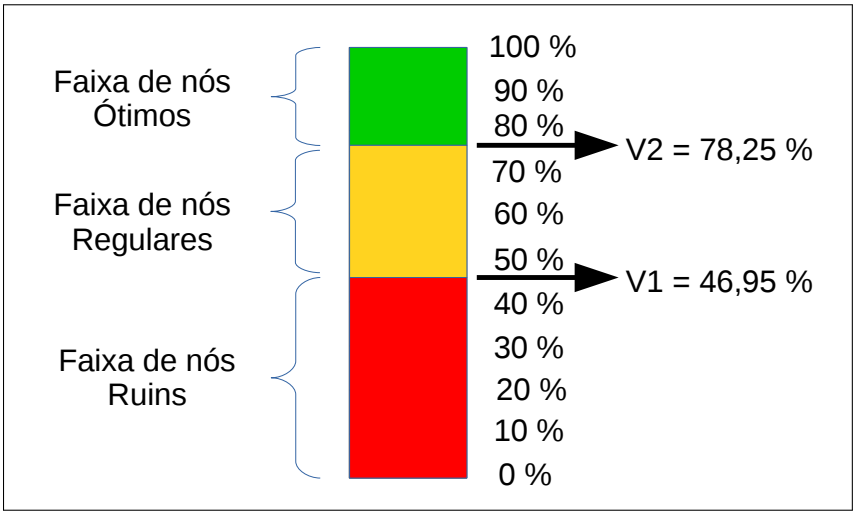
Onde:
Threshold V2 = Valor de *threshold* definido;
EstimatedThreshold = *Threshold* estimado;

DevThreshold = Variação de *threshold* estimada.

O valor do *threshold V1* é calculado como uma porcentagem do valor de *V2*. Se desejada uma grande faixa de nós regulares, deve-se definir *V1* como uma pequena porcentagem de *V2*. Se desejada uma pequena faixa de nós regulares, deve-se definir *V1* como uma grande porcentagem de *V2*.

Consideremos um exemplo no qual são calculados os valores de *V1* e *V2*: determinado nó decide fazer o *broadcast* de um pacote de dados. Ele estima as maiores probabilidades de encaminhamento dos vizinhos e faz uma média dos mesmos. Essa média é passada para a média móvel exponencial ponderada, obtendo um novo valor de *threshold V2*. Se após a aplicação da média móvel exponencial ponderada o valor de *V2* for *78,25 %* e a porcentagem em questão for *60 %*, por exemplo, o valor de *V1* será *60 %* de *78,25 %*, ou seja, *46,95 %*. Neste exemplo dado, teríamos o cenário descrito pela Figura 19.

Figura 19 – Exemplo de valores de *V1* e *V2*.



Fonte: Elaborado pelo autor

A porcentagem do valor de *V2* que é definido como o valor de *V1* é definida no Capítulo 5.

4.3 FATOR DE BENEVOLÊNCIA

O fator de benevolência indica o quão disposto a ajudar no encaminhamento de pacotes o receptor é. Ele é definido como um valor constante entre 0 e 100 %. Quanto maior o valor desse fator, maior a benevolência com a qual ele encaminhará pacotes recebidos dos

vizinhos. A sua utilização é feita na decisão sobre encaminhamento.

4.4 DECISÃO SOBRE ENCAMINHAMENTO

A decisão sobre o encaminhamento será tomada através da combinação das seguintes funcionalidades:

1. Classificação do nó emissor quanto à confiança, como definido pela Tabela 2;
2. Classificação do nó receptor quanto à probabilidade de encaminhamento, como definido pela Tabela 3;
3. Fator de benevolência;
4. Fórmulas probabilísticas aplicadas no *ProbT* (LIMA et al., 2015);
5. Mecanismos de temporização aplicados no *ProbT* (LIMA et al., 2015).

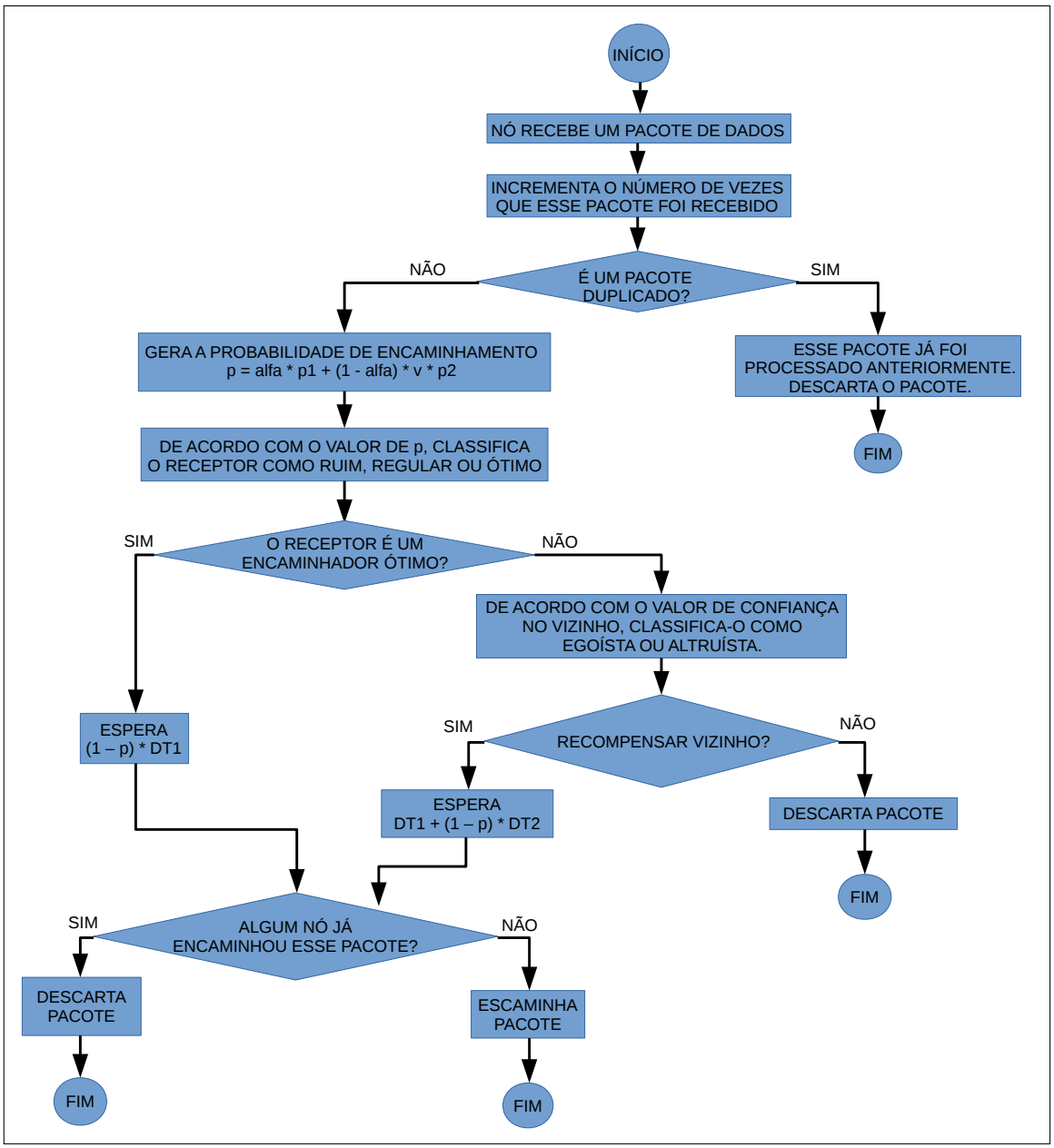
Essa combinação será feita de acordo com a Figura 20.

Este procedimento é ativado a partir da recepção de um pacote de tráfego. Quando tal fato ocorre, o nó verifica se já recebeu esse pacote anteriormente. Se já tiver recebido, ocorre o descarte. Se não, ele gera a probabilidade p de encaminhamento definida na Fórmula (3.15). De acordo com o valor de p, classifica-se o nó emissor como Ruim, Regular ou Ótimo, como indicado na Tabela 3.

Se o receptor for um ótimo encaminhador, ele aplicará os mecanismos de diferenciação no tempo na primeira janela para encaminhamento do pacote. Os nós devem esperar um intervalo de tempo e então verificar se algum outro nó já encaminhou esse pacote recebido. Se nenhum outro nó tiver encaminhado, o nó deve realizar o encaminhamento. Nós com maiores valores de probabilidade esperarão menores valores de tempo e vice-versa. Os nós com menores valores de probabilidade, ao sair do intervalo de espera, podem verificar que algum outro nó já enviou esse pacote. Assim, não será mais necessário o seu encaminhamento. Através desse mecanismo, faz-se com que os nós com melhores probabilidades enviem o pacote. Além disso, dificulta o ocasionamento de colisões.

Se o receptor não for um ótimo encaminhador, deve aguardar para que um outro nó faça o encaminhamento. Assim, ele deve esperar toda a primeira janela de tempo por encaminhamentos. Se esse pacote não foi encaminhado na primeira janela, deve-se realizar o reencaminhamento desse pacote. Entretanto, vários nós podem perceber a falta de encaminhamento e prosseguir com o *broadcast* (o que deve ser evitado). Aplica-se então um novo mecanismo de diferenciação no tempo, desta vez na segunda janela. Nessa fase será aplicado

Figura 20 – Diagrama de recepção e encaminhamento de pacotes de dados.



Fonte: elaborado pelo autor

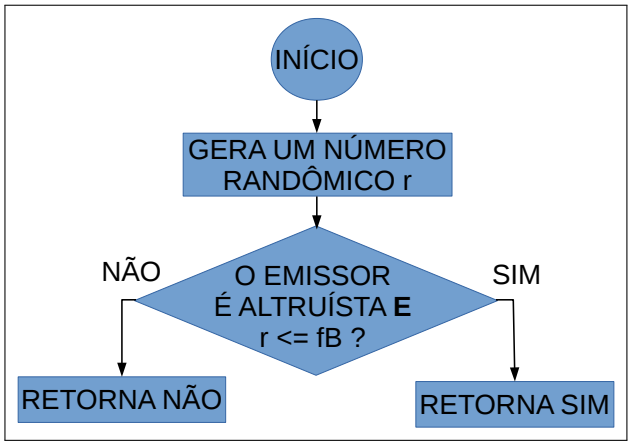
um mecanismo de recompensa/penalidade do emissor. Ele será aplicado como uma política de recompensa para os nós altruístas e penalização para os nós egoístas. Tal mecanismo tem as seguintes propriedades:

- Recompensa os vizinhos que encaminharam os pacotes anteriormente enviados por este nó;
- Pensaliza os vizinhos que rejeitaram o encaminhamento;
- Limitar a quantidade de encaminhamentos por nós não adequados.

O ponto central para recompensa/penalidade de vizinhos é a classificação do emissor quanto a confiança, como descrito no Fluxograma da Figura 20. Com base nessa classificação, são propostas algumas variantes desse modelo que também levam em conta outras características:

1. Confiança no vizinho: prosseguirá com o mecanismo de diferenciação no tempo apenas se o emissor for altruísta (de acordo com a Tabela 2). Essa variante é chamada de *E-ProbT 0*;
2. Confiança no vizinho **e** probabilidade: prosseguirá com o mecanismo de diferenciação no tempo apenas se o emissor for altruísta (de acordo com a Tabela 2) **e** se o receptor for regular (de acordo com a Tabela 3). Essa variante é chamada de *E-ProbT 1*;
3. Confiança no vizinho **ou** probabilidade: prosseguirá com o mecanismo de diferenciação no tempo apenas se o emissor for altruísta (de acordo com a Tabela 2) **ou** se o receptor for regular (de acordo com a Tabela 3). Essa variante é chamada de *E-ProbT 2*;
4. Confiança no vizinho **e** um fator de benevolência: gera-se um número randômico, de acordo com a distribuição uniforme, entre 0 e 1. Caso esse número seja menor ou igual ao fator de benevolência (também entre 0 e 1) **e** o emissor seja altruísta (de acordo com a Tabela 2), prossegue-se com o mecanismo de diferenciação no tempo. Caso contrário, o pacote é descartado. Essa variante é chamada de *E-ProbT 3*. Ela é descrita na Figura 21.

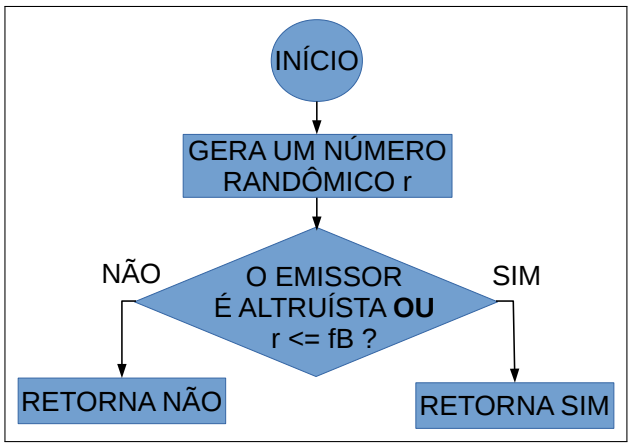
Figura 21 – Mecanismo de recompensa na segunda janela de tempo.



Fonte: elaborado pelo autor

5. Confiança no vizinho **ou** um fator de benevolência: gera-se um número randômico, de acordo com a distribuição uniforme, entre 0 e 1. Caso esse número seja menor ou igual ao fator de benevolência (também entre 0 e 1) **ou** o emissor seja altruísta (de acordo com a Tabela 2), prossegue-se com o mecanismo de diferenciação no tempo. Caso contrário, o pacote é descartado. Essa variante é chamada de *E-ProbT 4*. Ela é descrita na Figura 22.

Figura 22 – Mecanismo de recompensa na segunda janela de tempo.



Fonte: elaborado pelo autor

Os intervalos de espera na primeira e segunda janela de tempo são definidos de acordo com as Fórmulas (4.4) e (4.5), respectivamente:

$$Tempo\ de\ espera\ 1 = (1 - Prob) * DT1 \tag{4.4}$$

Onde:

Prob = Probabilidade gerada;

DT1 = Intervalo de tempo 1.

$$Tempo\ de\ espera\ 2 = DT1 + (1 - Prob) * DT2 \tag{4.5}$$

Onde:

Prob = Probabilidade gerada;

DT1 = Intervalo de tempo 1;

DT2 = Intervalo de tempo 2.

4.5 PACOTES UTILIZADOS NO PROTOCOLO

Nesta Seção são definidos os pacotes utilizados no protocolo.

4.5.1 Categorias de pacotes

No protocolo *E-ProbT* são consideradas duas categorias de pacotes:

- Pacotes *Hello*: estes pacotes são usados pelos nós para informarem aos seus vizinhos as suas características de mobilidade (posição, velocidade e aceleração). Estes pacotes não são encaminhados pelos nós receptores. Uma vez recebido, as suas informações são salvas em uma entrada na tabela de vizinhança. Se não houver uma entrada na tabela associada à esse vizinho originador da mensagem, uma entrada é então criada. Se já houver uma entrada associada a esse vizinho, os dados são atualizados;
- Pacotes de Tráfego (ou dados): estes pacotes são usados para simular uma aplicação de *broadcast* de informações. Assim, os nós receptores devem usar uma fórmula probabilística para decidir sobre o encaminhamento destes pacotes.

4.5.2 Formato do pacote de tráfego

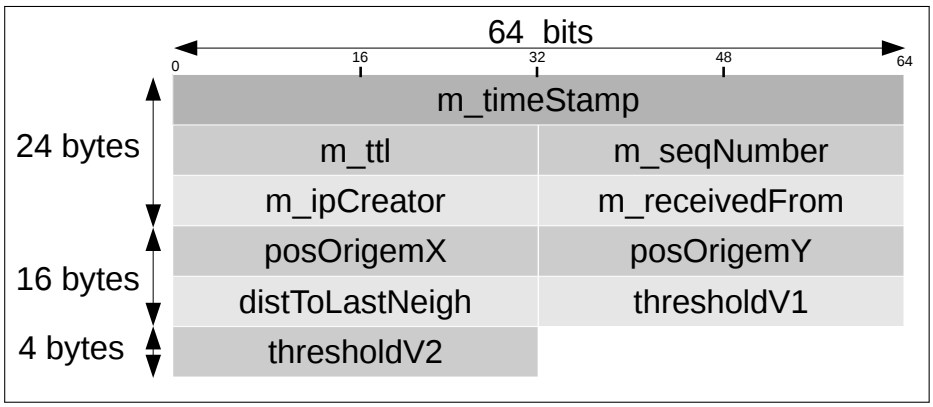
O pacote de tráfego é criado de acordo com o Código Fonte 1. Ele é descrito de acordo com a Linguagem de programação C++.

Código-fonte 1 – Pacote de tráfego

```
1  class TrafficHeader: public Header
2  {
3   private:
4        double m_timeStamp;
5        int m_ttl;
6        int m_seqNumber;
7        Ipv4Address m_ipCreator;
8        Ipv4Address m_receivedFrom;
9        float posOrigemX;
10       float posOrigemY;
11       float distToLastNeigh;
12       float thresholdV1;
13       float thresholdV2;
14 }
```

O Código Fonte 1 produz o pacote exibido na Figura 23.

Figura 23 – Formato do pacote de tráfego.

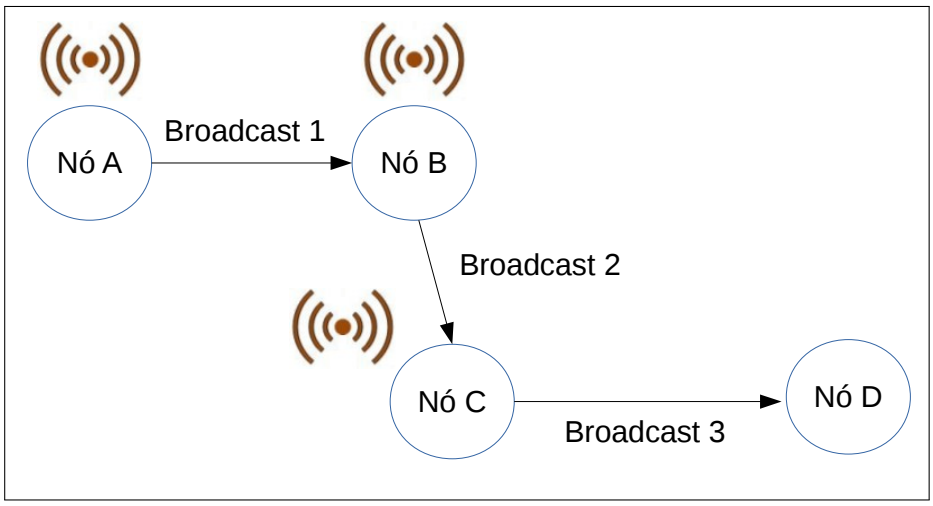


Fonte: Elaborado pelo autor

O pacote de tráfego usado, descrito na Figura 23, contém campos que são necessários para o funcionamento dos protocolos simulados. A seguir, descreve-se a semântica de cada campo desse pacote:

1. **double m_timeStamp**: é um campo que descreve o instante de tempo no qual o pacote foi criado pelo seu originador. Esse campo não é alterado à medida que são feitos encaminhamentos. Uma das suas possíveis utilizações é para saber se ele foi recentemente produzido. Dependendo do protocolo, o conceito de recente pode ser alterado. Alguns protocolos podem desejar descartar um pacote caso ele não seja recente. Este campo, no simulador de rede NS3 *(Network Simulator version 3)*, possui 8 bytes;
2. **int m_ttl**: é um campo que descreve a quantidade de saltos do pacote. Ele é utilizado para evitar que pacotes circulem indefinidamente na rede. Tambem é utilizado no Jogo do Encaminhamento. Ele é definido inicialmente como zero. A cada encaminhamento, o seu valor é incrementado. Este campo, no simulador de rede NS-3, possui 4 bytes;
3. **int m_seqNumber**: é um campo usado para indicar o número de sequência de um pacote. Considerando que um nó gera diversos pacotes, eles devem ser numerados para que os receptores saibam qual o *id* do mesmo. O seu valor não é alterado a medida que são feitos encaminhamentos. Este campo, no simulador de rede NS-3, possui 4 bytes;
4. **Ipv4Address m_ipCreator**: esse campo é usado para indicar o endereço ip do nó criador desse pacote. Quando um nó recebe um pacote, ele pode vir diretamente do nó originador ou de algum encaminhador. Assim, existe a necessidade de armazenar no pacote qual nó o originou. Está sendo usada a versão 4 do protocolo IP. Assim, ele possui 4 bytes;
5. **Ipv4Address m_receivedFrom**: ao encaminhar um pacote, indica-se de qual nó ele foi recebido. Considere a Figura 24.

Figura 24 – Exemplo de encaminhamento de pacote de tráfego.



Fonte: Elaborado pelo autor

Na Figura 24, temos uma sequência de *broadcasts* realizados até que o nó D receba o pacote. Como o nó A origina o pacote, no *broadcast 1*, ele não foi recebido de nenhum outro nó. Assim, o campo m_receivedFrom não precisa ser preenchido. Os nós receptores indentificam tal situação através do TTL com valor 1. Temos então a seguinte configuração:

- Ipv4Address m_ipCreator: IP de A;
- Ipv4Address m_receivedFrom: preenchido com zeros.

O nó B recebe o pacote fruto do *broadcast 1*. Supondo que ele encaminha o pacote, temos a seguinte configuração:

- Ipv4Address m_ipCreator: IP de A;
- Ipv4Address m_receivedFrom: IP de A.

O nó C recebe o pacote fruto do *broadcast 2*. Supondo que ele encaminha o pacote, temos a seguinte configuração:

- Ipv4Address m_ipCreator: IP de A;
- Ipv4Address m_receivedFrom: IP de B.

O nó D recebe o pacote fruto do *broadcast 3*. Assim, ele sabe que foi produzido pelo nó A e foi encaminhado como solicitação de B. O campo m_receivedFrom possui 4 bytes.

Esse campo é necessário para que cada nó emissor recompense os nós encaminhadores. Considere o exemplo anterior: o nó B, ao receber o pacote encaminhado por C, verifica quem foi o encaminhador anterior. Neste caso, ao verificar que o nó C encaminhou o seu pacote, recompensa o vizinho C.

6. **float posOrigemX**: indica o valor da coordenada x da posição do emissor. A cada

encaminhamento do pacote, o seu valor é alterado com a coordenada x do encaminhador. Ele é utilizado para o cálculo da distância entre o emissor e o receptor. Este campo, no simulador de rede NS-3, possui 4 bytes;

7. **float posOrigemY**: indica o valor da coordenada y da posição do emissor. A cada encaminhamento do pacote, o seu valor é alterado com a coordenada y do encaminhador. Ele é utilizado para o cálculo da distância entre o emissor e o receptor. Este campo, no simulador de rede NS-3, possui 4 bytes;
8. **float distToLastNeigh**: indica a distância para o vizinho mais distante do emissor. Essa característica é usada pelo *E-ProbT* no cálculo da probabilidade de encaminhamento. Este campo, no simulador de rede NS-3, possui 4 bytes;
9. **float thresholdV1**: indica o valor do *threshold V1*, definido na Seção 4.2. Este campo, no simulador de rede NS-3, possui 4 bytes;
10. **float thresholdV2**: indica o valor do *threshold V2*, definido na Seção 4.2. Este campo, no simulador de rede NS-3, possui 4 bytes.

O tamanho do pacote de tráfego, dado pela soma dos tamanhos dos campos indicados, é de 44 bytes.

4.5.3 Formato do pacote Hello

O formato do pacote Hello é exibido no Código Fonte 2. Ele é descrito de acordo com a Linguagem de programação C++.

Código-fonte 2 – Formato do pacote Hello

```
1 class GeneralHeader: public Header
2 {
3  private:
4       uint8_t m_type;
5       Vector2d m_pos;
6       Vector2d m_vel;
7       Vector2d m_accel;
8       double m_timeStamp;
9 }
```

No Código Fonte 2, temos uma parte da classe que declara o formato do pacote *Hello*. Ele contém campos que são necessários para o funcionamento do protocolo de *Hello*, usado para que os nós montem sua tabela de vizinhança. A seguir, descreve-se a semântica de cada campo desse pacote:

1. **uint8_t m_type**: usado para indicar o tipo de pacote *Hello*. Neste trabalho, é definida uma única versão. Este campo, no simulador de rede NS-3, possui 1 byte;
2. **Vector2d m_pos**: indica a posição do nó emissor no formato (posição ordenada x, posição ordenada y). Este campo, no simulador de rede NS-3, possui 16 bytes;
3. **Vector2d m_vel**: indica a velocidade do nó emissor no formato (velocidade ordenada x, velocidade ordenada y). Este campo, no simulador de rede NS-3, possui 16 bytes;
4. **Vector2d m_accel**: indica a aceleração do nó emissor no formato (aceleração ordenada x, aceleração ordenada y). Este campo, no simulador de rede NS-3, possui 16 bytes;
5. **double m_timeStamp**: indica o instante de tempo que o pacote foi criado. Este campo, no simulador de rede NS-3, possui 8 bytes.

O tamanho do pacote *Hello*, dado pela soma dos tamanhos dos campos indicados, é 57 bytes.

5 RESULTADOS

Para a realização desse trabalho, foram utilizados dois simuladores: o Simulador de Mobilidade Urbana (SUMO) e o Simulador de Rede versão 3 (NS-3 - *Network Simulator Version 3*). O Sumo foi utilizado para a geração do arquivo descritor da mobilidade dos veículos. O NS-3 foi utilizado para fazer a simulação dos protocolos. O arquivo descritor da mobilidade dos nós, criado pelo SUMO, foi passado como entrada para o NS-3.

5.0.4 *SUMO*

Através do SUMO (*Simulation of Urban Mobility*), foi criado o arquivo descritor da mobilidade dos veículos. O cenário criado é uma autoestrada com duas vias. Os veículos pertencentes à simulação se movimentam por essa autoestrada com velocidades diferentes, mesma direção e mesmo sentido. Nessa simulação, cada veículo é colocado em uma das duas vias e pode realizar ultrapassagens. Em (WISITPONGPHAN et al., 2007) encontra-se uma classificação para níveis de tráfego veiculares para uma via:

- Light (leve): 10 carros/km/via;
- Moderate (moderado): 25 carros/km/via;
- Heavy (denso): 50 carros/km/via;
- Jam (congestionado): 100 carros/km/via.

Cada veículo é atribuído à uma das duas vias de forma aleatória. Como são usadas duas vias, os valores usados foram 20, 30, 60, 90 e 120 carros/km para a obtenção de valores entre 10, 15, 30, 45 e 60 carros/km/via correspondentes a leve, moderado e denso (WISITPONGPHAN et al., 2007). Além disso, uma vez determinada a quantidade de carros na simulação e a densidade, eles são posicionados na autoestrada seguindo um espaçamento inter veicular distribuído exponencialmente (PANICHPAPIBOON; FERRARI, 2008). Todos os veículos se movimentam da esquerda para a direita.

5.0.5 *NS3*

A simulação do NS3 é feita da seguinte maneira: existem os pacotes *Hello* e os pacotes de tráfego. Os pacotes *Hello* não são encaminhados. Os pacotes de tráfego vão ser encaminhados de acordo com as fórmulas probabilísticas do *Blind Flooding*, *Wheighted p-Persistence*, *AutoCast*, *Irresponsible Forwarding*, *ProbT* e ***E-ProbT***. As mensagens de tráfego

são geradas segundo uma periodicidade de 1 segundo.

São propostas 5 variantes de E-ProbT (numeradas de zero a quatro de acordo com o modelo de recompensa). Assim, além da simulação dos 5 trabalhos relacionados, realiza-se a simulação das 5 variantes do E-ProbT. Inicialmente, faz-se a simulação das 5 variantes do E-ProbT. Dentre essas variantes, uma é escolhida e comparada com os trabalhos relacionados.

Dessa forma, existe um protocolo de envio de mensagens *Hello* e 10 protocolos de envio de mensagens de tráfego. Cada protocolo de mensagens de tráfego deve ser executado de forma paralela ao protocolo de *Hello*. Os 10 protocolos de tráfego são testados nos mesmos cenários, em diferentes momentos, para a comparação de resultados ser realizada de forma justa.

Foi realizado um intervalo de *warm-up* (em uma tradução livre: aquecimento). Considerando a aplicação da Teoria dos Jogos no jogo do encaminhamento, a confiança de um nó em seus vizinhos é inicializada com um valor *default*. Esse período é utilizado para uma melhor caracterização da confiança, antes do início da medição de métricas utilizadas.

Dentre todos os nós da rede, escolhe-se um para ser o gerador de pacotes. Assim, os demais nós da rede são encaminhadores. Esse procedimento foi adotado por ser o utilizado nos trabalhos (ROBERTO et al., 2011), (PAULA et al., 2014) e (LIMA et al., 2015).

As simulações são realizadas de acordo com as configurações exibidas na Tabela 4:

5.0.5.1 Escolha dos parâmetros

Em geral, os parâmetros da Tabela 4 foram escolhidos por configurações semelhantes utilizadas nos trabalhos (WISITPONGPHAN et al., 2007), (ROBERTO et al., 2011), (PAULA et al., 2014) e (LIMA et al., 2015). Cada um dos parâmetros escolhidos é explicado a seguir:

1. **Número de iterações**: 33. Esta quantidade foi escolhida por ser a mesma usada em trabalhos similares, tais como (ROBERTO et al., 2011), (PAULA et al., 2014) e (LIMA et al., 2015);
2. **Número de veículos**: 300. Usada em trabalhos similares, tais como: (ROBERTO et al., 2011), (PAULA et al., 2014) e (LIMA et al., 2015);
3. **Densidades veiculares**: 20, 30, 60, 90 e 120 veículos/km. Esses são os valores proporcionais aos indicados em (WISITPONGPHAN et al., 2007) para obtenção de níveis de tráfego leve, moderado e denso;
4. **Velocidade inicial**: 36, 54, 72, 90 km/h. Esses valores foram escolhidos para ter amostras de carros lentos e rápidos. Esses valores foram escolhidos de forma empírica;

Tabela 4 – *Configuração dos cenários*

Parâmetro	Valor
Número de iterações	33
Número de veículos	300
Densidades veiculares	20, 30, 60, 90 e 120 veículos/km
Velocidade inicial	36, 54, 72, 90 km/h
Tamanho do pacote de tráfego	48 bytes
Taxa de envio de pacotes de tráfego	1 pacote/seg
Modelo de propagação	Nakagami
Transporte	UDP (*User Data Protocol*)
MAC/PHY	802.11p
Raio de transmissão	1 km
Tempo de simulação	300 Segundos
α	0.3
β	0,125
γ	0,25
Recompensa (REC)	0,2
Fator da Penalidade (PEN)	0,01
Threshold V1	60 % de *V2*
Fator de benevolência (fB)	70 %
k	60 %
DT1	0.05 fs (femtosegundos: 10^{-15} s)
DT2	0.05 fs (femtosegundos: 10^{-15} s)
Intervalo de confiança	95 %

5. **Tamanho do pacote de tráfego**: 48 bytes. Esse valor é uma decorrência dos tamanhos dos campos usados pelo protocolo;
6. **Taxa de envio de pacotes de tráfego**: 1 pacote/seg. O interesse da simulação é medir características como taxa de entrega. Elas podem ser obtidas com qualquer taxa de pacotes de tráfego. Maiores taxas de envio resultam em maiores quantidades de colisões, o que influenciaria as métricas analisadas;
7. **Modelo de propagação**: Nakagami. Esse é o modelo recomendado pela comunidade científica por apresentar uma modelagem mais próxima dos ambientes reais (POONIA; SINGH, 2012), (SINGH, 2012);
8. **Transporte**: UDP. Este protocolo foi o escolhido, em detrimento do TCP, por não utilizar temporizadores e retransmissões;
9. **MAC/PHY**: 802.11p. Este protocolo foi o escolhido por ser o que modela as VANETs;
10. **Raio de transmissão**: 1 km. Esta quantidade foi escolhida por ser a mesma usada em trabalhos similares, tais como (PAULA et al., 2014) e (LIMA et al., 2015);
11. **Tempo de simulação**: 300 Segundos. Usada em trabalhos similares, tais como (LIMA et

al., 2015);

12. α: 0.3. Este é o parâmetro de que define os pesos da probabilidade de encaminhamento do *E-ProbT*, definida pela Fórmula (3.15). Foi mantido o mesmo valor utilizado pelo *ProbT*;
13. β: 0,125. Este é o valor recomendado pelos implementadores do TCP (KUROSE; ROSS, 2010);
14. γ: 0,25. Este é o valor recomendado pelos implementadores do TCP (KUROSE; ROSS, 2010);
15. **Recompensa (REC)**: 0,2. Uma vez que um nó encaminha um pacote, existe uma forte tendência a que ele se torne um ótimo encaminhador. Assim, a confiança nesse vizinho deve aumentar rapidamente, para que a modelagem do jogo represente o ambiente real. Por essa razão, foi atribuído um valor de recompensa que permite crescer rapidamente a confiança no vizinho. Esse valor foi escolhido de forma empírica;
16. **Fator da Penalidade (PEN)**: 0,01. Considerando um raio de transmissão com diversos veículos, podem existir muitos nós com uma maior tendência para encamihar pacotes do que determinados nós. Isso faz com que tais nós encaminhem os pacotes primeiro, provocando sucessivos descartes por parte de nós com menores probabiliadades. Se configurado um alto valor de penalidade, a confiança nos mesmos tenderá para zero, o que provocará uma demora no crescimento da confiança quando tais nós se tornarem bons encaminhadores. Tal demora não deve existir, pois determinados nós podem se tornar bons encaminhadores por um curto período de tempo, considerando a dinâmica das VANETs. Assim, configura-se um baixo valor para penalidade. Esse valor foi escolhido de forma empírica;
17. ***Threshold V1***: 60 % de *V2*. Este valor foi escolhido para reservar uma pequena faixa de probabilidade para nós regulares. Ele foi escolhido de forma empírica;
18. **Fator de benevolência (fB)**: 70 %. Este valor foi escolhido para que os nós tenham uma maior benevolência para encaminhar pacotes na segunda janela de tempo. Ele foi escolhido de forma empírica;
19. **k**: 60 %. Este é o limiar da clasificação de nós como Egoístas ou Altruístas. Foi configurado um elevado valor para que o protocolo seja exigente com relação a confiança nos vizinhos. Ele foi escolhido de forma empírica;
20. **DT1**: 0.05 fs (femtosegundos: 10^{-15} *s*). O simulador de rede NS3 permite o trabalho com intervalos de tempo na ordem de femtosegundos. Foi configurado um pequeno valor para

que não sejam introduzidos atrasos consideráveis nas comunicações. Ele foi escolhido de forma empírica;

21. **DT2**: 0.05 fs (femtosegundos: 10^{-15} *s*). O simulador de rede NS3 permite o trabalho com intervalos de tempo na ordem de femtosegundos. Foi configurado um pequeno valor para que não sejam introduzidos atrasos consideráveis nas comunicações. Ele foi escolhido de forma empírica;
22. **Intervalo de confiança**: 95 %. Usado em trabalhos similares, tais como: (PAULA et al., 2014) e (LIMA et al., 2015).

5.0.6 Métricas

As métricas usadas para comparar os protocolos foram escolhidas devido a suas utilizações em trabalhos similares sobre estratégias probabilísticas em VANETs, como descrito em (PANICHPAPIBOON; PATTARA-ATIKOM, 2012), (PAULA et al., 2014) e (LIMA et al., 2015). São usadas 3 métricas:

1. Taxa de entrega de pacotes normalizada;
2. Número de saltos;
3. Taxa de redundância.

5.0.6.1 Taxa de entrega de pacotes normalizada

É uma taxa pertencente ao intervalo [0, 1] em $\mathbb{R}$. Ela é a relação entre a quantidade de veículos receptores dos pacotes transmitidos e o número de veículos da rede. Busca-se entregar um pacote a todos os nós da rede. Assim, apresentam uma melhor performance os protocolos que apresentarem as maiores taxas de entrega. Esta métrica é calculada de acordo com a Fórmula (5.1):

$$\textit{Taxa de entrega de pacotes} = \frac{\textit{Total de veículos receptores dos pacotes}}{\textit{Total de veículos na rede}} \tag{5.1}$$

5.0.6.2 Número de saltos

Essa métrica é uma média da máxima quantidade de saltos que cada pacote fez. Busca-se entregar o pacote até as posições mais distantes da rede. Assim, apresentam uma melhor performance os protocolos com maiores quantidades de saltos. Esta métrica é calculada

de acordo com a Fórmula (5.2):

$$\textit{Número de saltos} = \frac{\textit{Soma das máximas quantidades de saltos}}{\textit{Quantidade de pacotes gerados}} \tag{5.2}$$

5.0.6.3 Taxa de redundância

Essa métrica indica a quantidade média de duplicações de cada pacote recebida pelos nós. Apresentam uma melhor performance os protocolos com menores taxas de duplicações. Esta métrica é calculada de acordo com a Fórmula (5.3):

$$\textit{Taxa de redundância} = \frac{\textit{Soma da quantidade de duplicações}}{\textit{Total de pacotes recebidos de diferentes IDs}} \tag{5.3}$$

Na Fórmula (5.3) também é levado em consideração o nó emissor, uma vez que ele teve acesso ao dado no momento da geração e pode, eventualmente, receber novas cópias deste pacote.

Observação: existem diversas métricas que podem ser usadas para comparar protocolos de broadcast. Um estudo sobre diversas opções de métricas é realizado em (PANICH-PAPIBOON; PATTARA-ATIKOM, 2012). Em geral, as métricas indicadas mostram diferentes formas de analisar a eficiência da entrega de pacotes, o quão distante um pacote é levado e a redundância de pacotes produzida. Essas características foram estudadas através das métricas selecionadas para este trabalho. Outras métricas podem ser analisadas, tais como tempo médio para entrega de pacotes ou quantidade de processamento, entretanto, não fazem parte do escopo deste trabalho.

5.0.7 *Resultados obtidos entre as 5 variantes do E-ProbT*

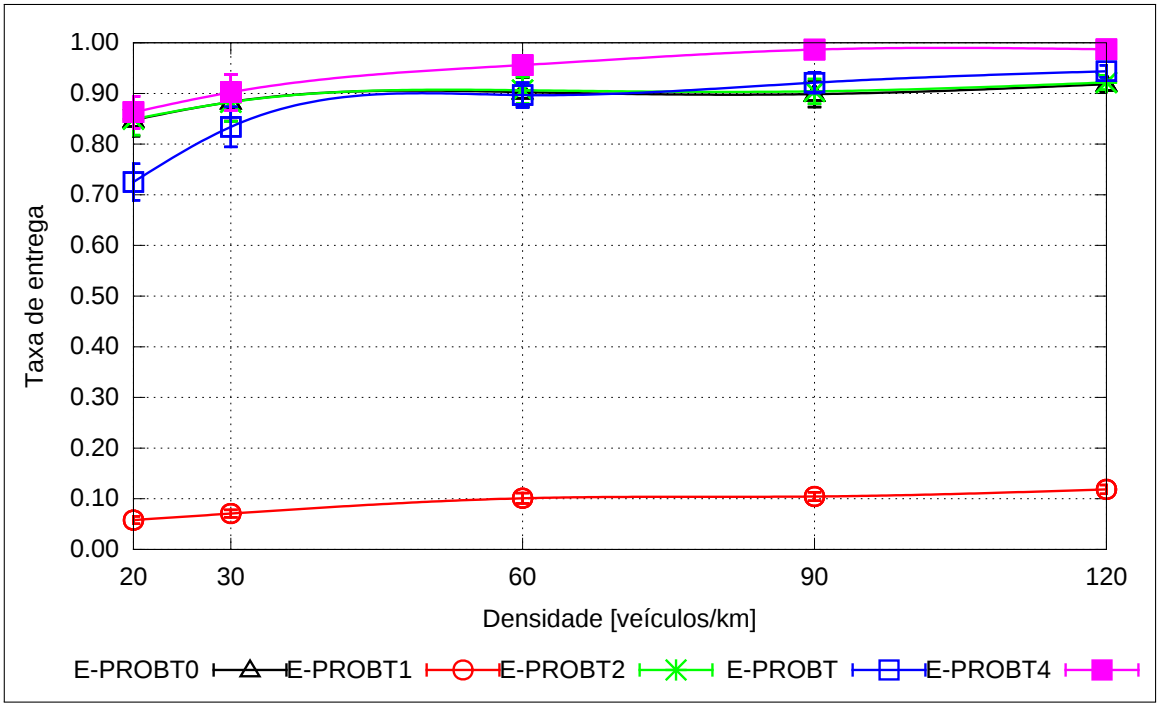
Inicialmente, exibe-se os resultados da simulação das 5 variantes do E-ProbT.

5.0.7.1 Taxa de entrega de pacotes normalizada

A Figura 25 apresenta os resultados da métrica taxa de entrega de pacotes normalizada.

A Figura 25 mostra que as diferentes variantes do E-ProbT obtem altas taxas de entrega, da ordem de 90 %. Com as maiores taxas de entrega, na ordem de 95 %, apresenta-se o E-ProbT 4. O E-ProbT 4 se baseia tanto na confiança no vizinho como em um fator de benevolência.

Figura 25 – Taxa de entrega de pacotes normalizada.



Fonte: elaborado pelo autor

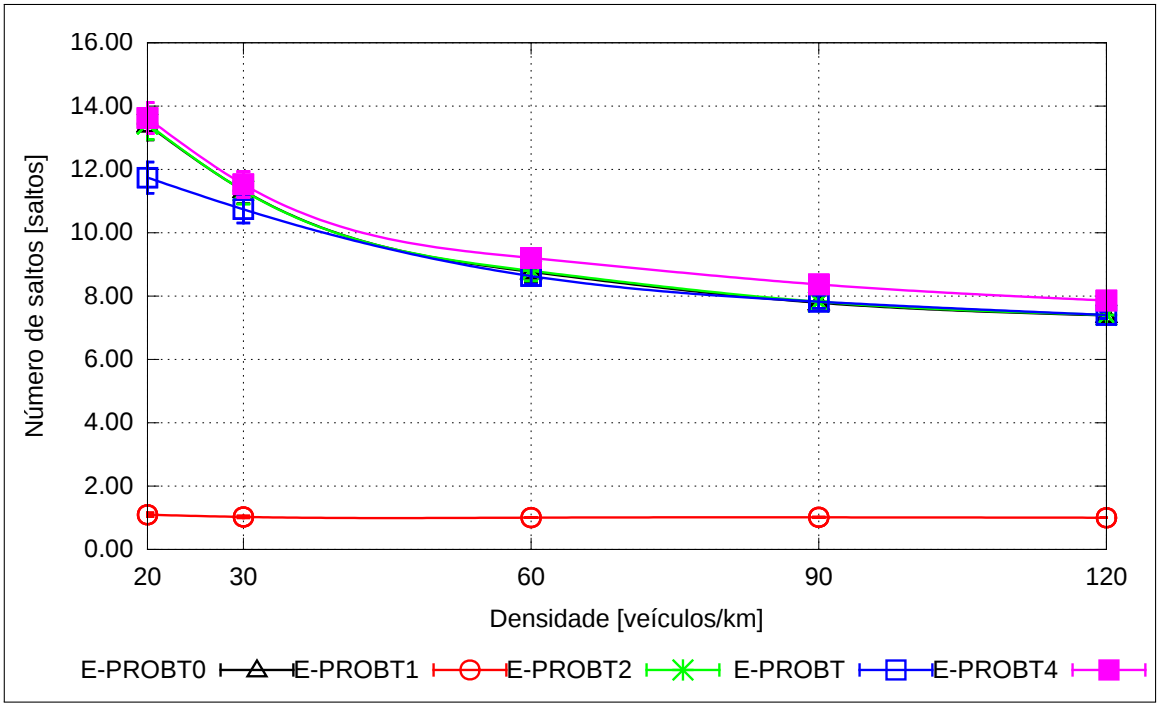
Esse modelo permite o encaminhamento caso a confiança ou o fator de benevolência sejam adequados. Isso permite uma maior quantidade de encaminhamentos, com relação as outras variantes, pois são analisadas 2 características. Essa modelagem se mostrou eficaz, produzindo altas taxa de entrega, sendo então um bom mecanismo de seleção de encaminhador na segunda janela de tempo. As outras variantes do E-ProbT, com exceção da versão 1, também se mostraram eficientes. Cada uma apresenta um rigor diferente para encaminhamento na segunda janela de tempo, produzindo variações nas taxas de entrega. Em geral, os critérios de seleção de encaminhadores, na segunda janela de tempo, foram adequados para selecionar os melhores encaminhadores. O E-ProbT versão 1 se mostra extremamente exigente com relação a seleção de encaminhadores. Dessa forma, poucos nós tem a qualificação necessária, na segunda janela de tempo, para o encaminhamento. Essa métrica mostra que de acordo com o rigor usado na decisão do encaminhamento são produzidas diferentes taxas de entrega.

5.0.7.2 Número de saltos

A Figura 26 apresenta os resultados da métrica número de saltos.

A Figura 26 mostra que as diferentes variantes do E-ProbT entregam os pacotes em longas distâncias. Com as maiores quantidades de saltos apresenta-se o E-ProbT 4. Os

Figura 26 – Número de saltos.



Fonte: elaborado pelo autor

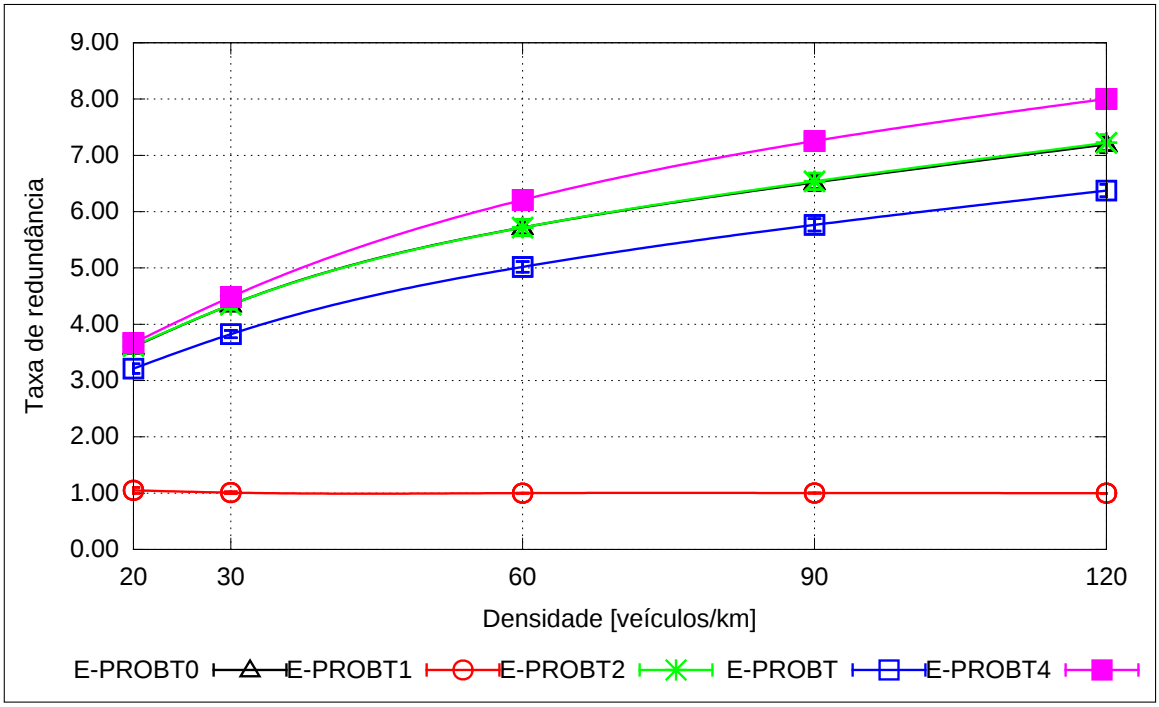
mecanismos de seleção aplicados nas diferentes variantes do E-ProbT, com exceção da versão 1, se mostram eficientes. O rigor aplicado no processo de decisão da segunda janela de tempo produz as diferenciações nas variantes do E-ProbT. Em geral, os pacotes são entregues em longas distâncias e de forma adequada, produzindo as altas taxa de entrega exibidas na Figura 25. A variante 1 possui muitas exigências, o que habilita poucos nós para a realização do encaminhamento. Isso se reflete na métrica número de saltos, fazendo com que o pacote não chegue em posições distantes do emissor.

5.0.7.3 Taxa de redundância

A Figura 27 apresenta os resultados da métrica taxa de redundância.

As maiores taxas de redundância ocorrem na variante E-ProbT 4. Como as demais variantes do E-ProbT produzem menores quantidades de encaminhamentos, elas tiveram uma menor taxa de redundância. Como a variante 1 apresenta baixas taxas de entrega e número de saltos, ela é a que apresenta a menor taxa de redundancia. Assim sendo, ela não pode ser comparada de forma justa com as demais variantes. Na modelagem aplicada, a taxa de redundância é um *trade-off* com relação as métricas taxa de entrega e número de saltos. Em geral, protocolos com maiores taxas de entrega e número de saltos tendem a possuir uma maior

Figura 27 – Taxa de redundância.



Fonte: elaborado pelo autor

taxa de redundância.

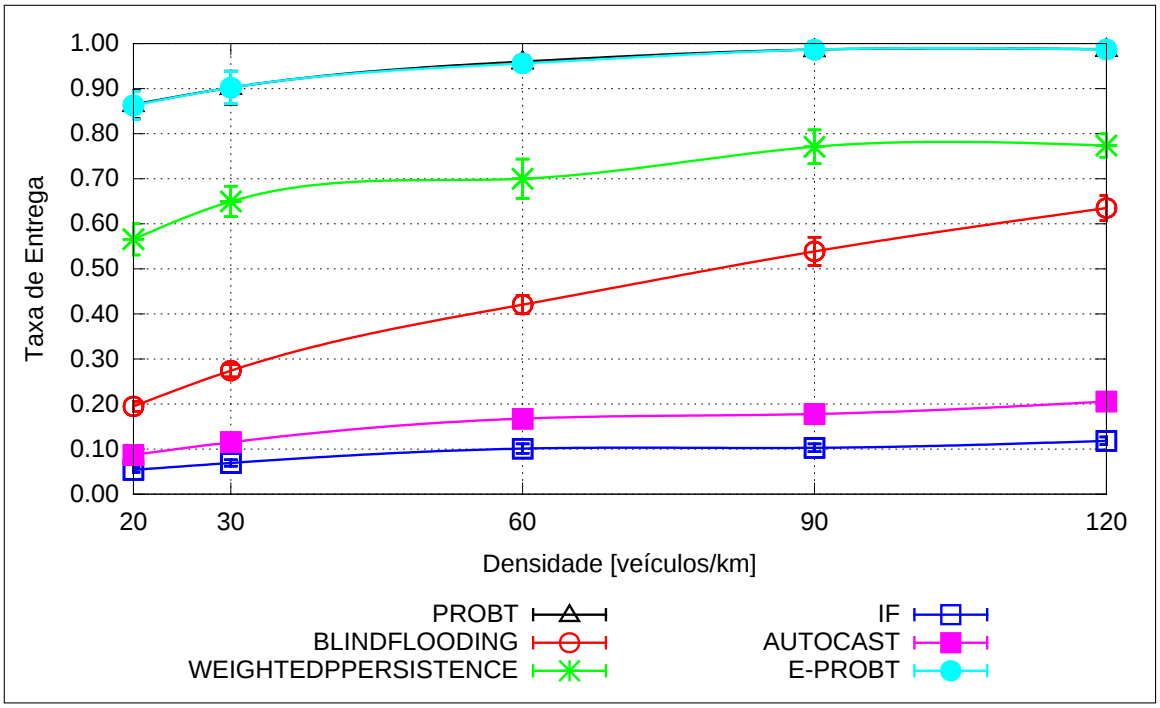
Com base nos resultados ilustrados nas Figuras 25, 26 e 27, a variante E-ProbT 4 ganhou destaque, tendo altas taxas de entrega e número de saltos. As outras variantes podem ser usadas se desejado um maior controle da taxa de redundância. Em geral, elas apresentam uma redução na taxa de redundância, mantendo altas a taxa de entrega e número de saltos. Para comparação com os demais trabalhos relacionados será usada a variante 4 do E-ProbT. A partir deste ponto, refere-se à E-ProbT 4 como E-ProbT.

5.0.8 *Resultados obtidos comparando o E-ProbT com trabalhos relacionados*

5.0.8.1 Taxa de entrega de pacotes normalizada

A Figura 28 apresenta os resultados da métrica taxa de entrega de pacotes normalizada.

Figura 28 – Taxa de entrega de pacotes normalizada.



Fonte: elaborado pelo autor

Os Protocolos *Blind Flooding*, *AutoCast* e *Irresponsible Forwarding* apresentaram taxas de entrega consideravelmente baixas. Tais protocolos não possuem quaisquer mecanismos sobre verificação de encaminhamento por parte de outros nós. Assim, sempre que uma probabilidade não modela corretamente a tendência por encaminhar dados, a performance tende a cair. O Protocolo *Wheighted p-Persistence* apresentou uma taxa de entrega regular. Usando a sua

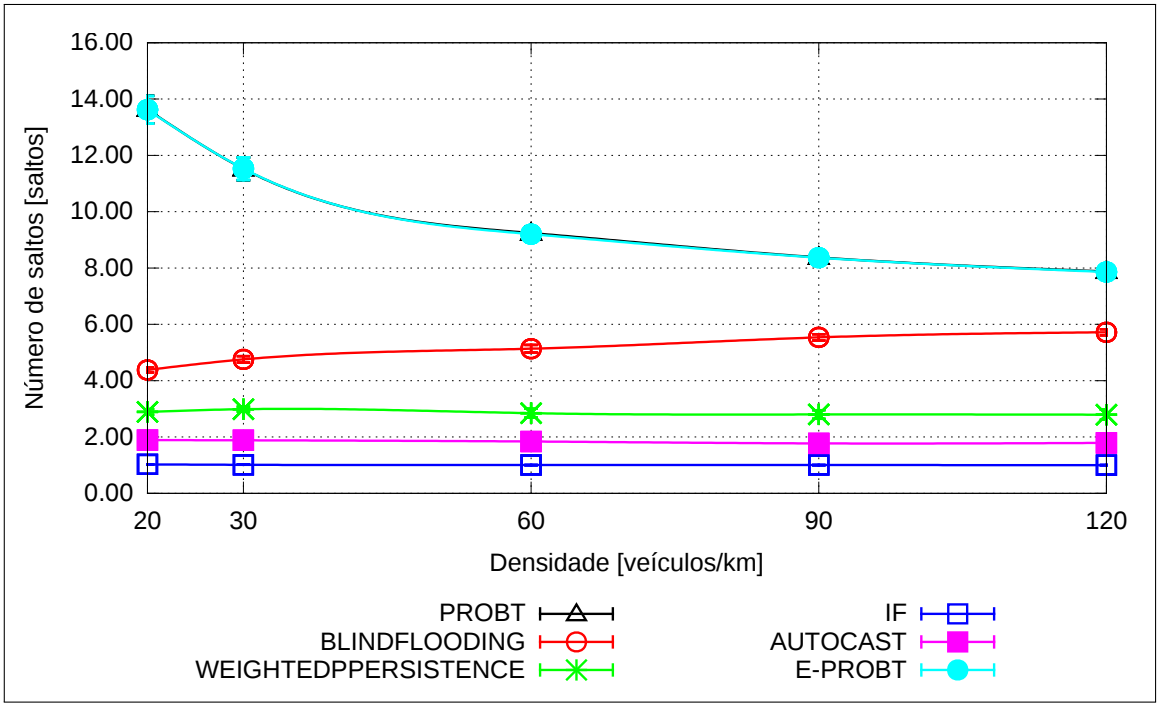
fórmula que modela a tendência por encaminhar dados com base na distância para o emissor, obtem-se uma taxa de entrega mediana. Entretanto, muitos nós deixam de receber os pacotes. Pode ser indicada como razão para a não recepção dos pacotes, o fato desse protocolo considerar o raio de transmissão preenchido por nós, o que nem sempre é verdade. Uma outra razão é a falta de verificação do encaminhamento de dados por outros nós. Dentre os protocolos com taxas de entregas próximas do 100 % estão o *ProbT* e ***E-ProbT***. Suas taxas de entrega, considerando esse conjunto de parâmetros usados, podem ser consideradas iguais. O Protocolo *ProbT* apresenta uma alta taxa de entrega devido à eficiente junção de uma fórmula probabilística e mecanismos de temporização. O Protocolo *E-ProbT* obtem a mesma taxa de entrega usando um mecanismo de seleção de encaminhadores refinado. Esse refinamento pode ser observado em outra métrica a ser analisada: taxa de redundância. Esse refinamento se dá pela análise da tendência de encaminhamento da vizinhança, definido pelo jogo do encaminhamento. Evita-se ainda que nós com baixas probabilidades encaminhem pacotes quando existem nós com probabilidades superiores, o que é possível no *ProbT*. Isso foi obtido através dos *thresholds* que se adaptam à dinâmica da rede. Alterando o conjunto de parâmetros, o *E-ProbT* obtem uma taxa de entrega ainda maior, inclusive para densidades menores. Foi mantido esse conjunto de parâmetros para manter baixa a métrica taxa de redundância.

5.0.8.2 Número de saltos

A Figura 29 apresenta os resultados da métrica número de saltos.

Os Protocolos *Wheighted p-Persistence*, *AutoCast* e *Irresponsible Forwarding* apresentaram quantidades de saltos consideravelmente baixas. Tais protocolos não verificam o encaminhamento de dados por outros nós, o que deixa as porções da rede mais distantes do emissor sem receber muitos pacotes. O Protocolo *Blind Flooding* obtém uma quantidade de saltos mediana. Muitos nós distantes do emissor continuam ser receber os pacotes, como resultado das colisões que ocorrem no processo de encaminhamento. Os Protocolos com as maiores quantidades de saltos são o *ProbT* e ***E-ProbT***. Eles conseguem entregar os dados em porções mais distantes da rede, inclusive dentro de 14 saltos para a menor densidade. Isso se deve a política de fazer com que os nós mais distantes do emissor encaminhem os pacotes, possibilitando uma maior propagação e uma menor quantidade de colisões. Destaca-se ainda a verificação de encaminhamento por parte de outros nós, o que permite a entrega de dados à porções da rede que, a priori, não receberiam tais pacotes. Alterando o conjunto de parâmetros, o *E-ProbT* obtém

Figura 29 – Número de saltos.



Fonte: elaborado pelo autor

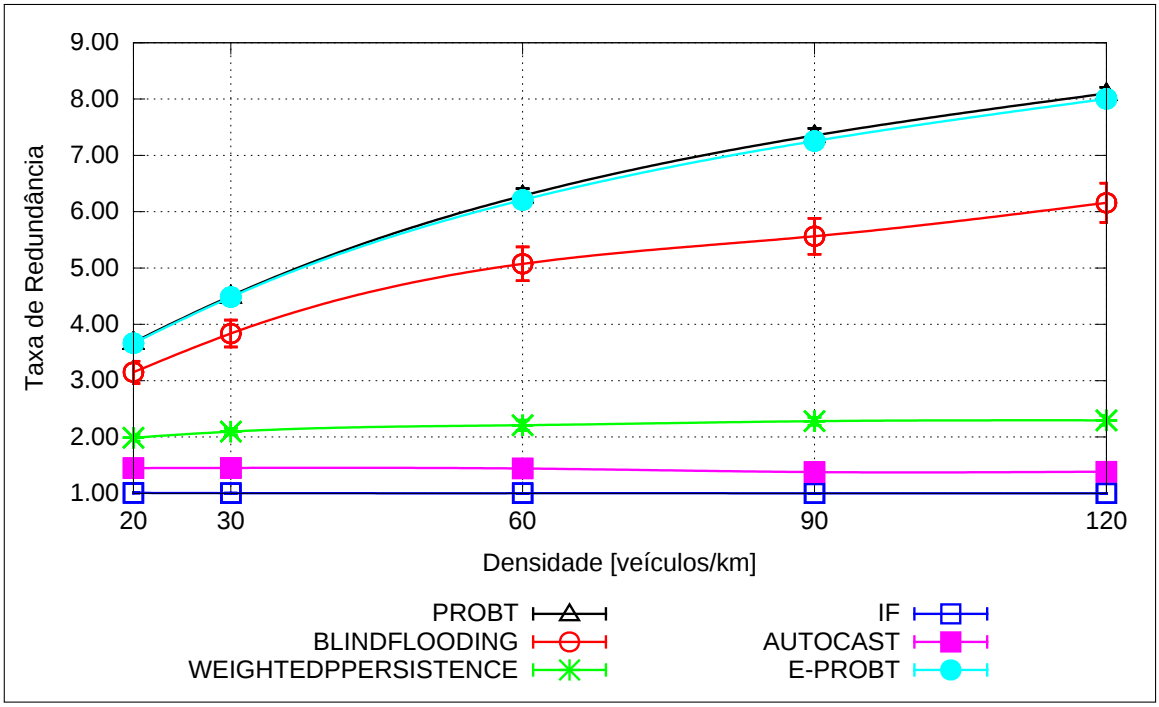
uma quantidade de saltos ainda maior. Foi mantido esse conjunto de parâmetros para manter baixa a métrica taxa de redundância.

5.0.8.3 Taxa de redundância

A Figura 30 apresenta os resultados da métrica taxa de redundância.

Os Protocolos *AutoCast* e *Irresponsible Forwarding* apresentaram taxas de entrega e número de saltos consideravelmente baixas. Assim, não podem ter a taxa de redundância comparada com a dos demais protocolos. Os Protocolos *Wheighted p-Persistence* e *Blind Flooding* apresentaram valores medianos de taxa de entrega e número de saltos. Entretanto, uma grande porção da rede continua sem receber pacotes. Assim, suas taxas de redundância se mantem baixas. Vale destacar que o protocolo *Blind Flooding* é o que mais produz pacotes. Entretanto, as colisões existentes fazem com que muitos pacotes não sejam contabilizados pela métrica. Dentre os protocolos com as maiores taxa de redundância estão o *ProbT* e ***E-ProbT***. Isso se deve às altas taxas de entrega dos mesmos. Simulações mostram que diminuições na tendência de encaminhamento dos mesmos diminuem consideravelmente a taxa de redundância, como esperado. O Protocolo *E-ProbT* mostrou uma leve diminuição, quando comparado com o *ProbT*, na taxa de redundância. Isso pode ser observado como um ponto positivo para o *E-ProbT*,

Figura 30 – Taxa de redundância.



Fonte: elaborado pelo autor

mostrando que a seleção de vizinhos encaminhadores ocorre de forma a reduzir a redundância. Entretanto, tal diminuição ficou dentro do intervalo de confiança, o que pode ser considerado um novo empate. Outros conjuntos de parâmetros diminuem a taxa de redundância apresentada pelo *E-ProbT*. Essa configuração foi mantida para manter alta a taxa de entrega.

Após a realização de diversas simulações e análises das suas métricas, o *E-ProbT* mostrou ter um mecanismo de seleção de encaminhadores refinado. Cada uma das métricas pode ter seus valores alterados, de acordo com a vontade do utilizador do protocolo. Para isso, basta a escolha de um novo conjunto de parâmetros que satisfaçam as suas necessidades.

Cabe ao utilizador do protocolo uma escolha de parâmetros de configuração que mantenha as métricas analisadas dentro de níveis aceitáveis. Maiores taxas de entrega podem ser obtidas, resultando em maiores taxa de redundância. Menores taxa de redundância podem ser obtidas, resultando em menores taxa de entrega.

Os resultados exibidos neste trabalho e outros obtidos em laboratório mostram que o *E-ProbT* utiliza mecanismos que permitem um melhor controle das métricas analisadas, se comparado com o *ProbT*. Assim, tem-se uma nova forma de disseminação de informações de forma eficiente.

6 CONCLUSÕES E TRABALHOS FUTUROS

Foi proposto um protocolo de *broadcast* de mensagens em VANETs, intitulado *E-ProbT*. A decisão sobre encaminhamento de pacotes é baseada em:

1. Classificação do nó emissor quanto à confiança;
2. Classificação do nó receptor quanto à probabilidade de encaminhamento;
3. Fator de benevolência;
4. Fórmulas probabilísticas aplicadas no *ProbT* (LIMA et al., 2015);
5. Mecanismos de temporização aplicados no *ProbT* (LIMA et al., 2015).

A classificação do nó emissor quanto à confiança é aplicada com base na Teoria dos Jogos. A classificação do nó receptor quanto à probabilidade de encaminhamento é baseada em uma média móvel exponencial ponderada.

O *E-ProbT* foi comparado com os protocolos *Blind Flooding*, *Wheighted p-Persistence*, *AutoCast*, *Irresponsible Forwarding* e *ProbT*. Os resultados caracterizaram o *E-ProbT* como um novo protocolo efetivo para o *broadcast* de informações.

As métricas analisadas mostraram uma alta performance do protocolo do ponto de vista de taxa de entrega e quantidade de saltos. Os parâmetros utilizados devem ser escolhidos de forma a manter a taxa de duplicação dentro de uma faixa aceitável. Assim, a modelagem utilizada pelo E-ProbT é recomendada para implementadores que desejam tais características nos seus protocolos.

Como trabalhos futuros, podem ser indicados:

1. Extensão do protocolo para o cenário *Manhattan*: este protocolo foi simulado no cenário de uma auto-estrada. Um outro cenário no qual as VANETs pode ser aplicado é o de uma cidade, particularmente as suas ruas. Usualmente, a literatura se refere a tal cenário como *Manhattan*, considerando o formato das suas ruas em malha e com quarteirões em formato retangular. É possível ainda utilizar cenários de quaisquer outras cidades nas simulações. Para isso, podem ser usadas duas estratégias:
    - Modelagem manual das ruas: o formato da cidade pode ser modelado, ponto a ponto, usando simuladores microscópicos, tais como o SUMO. Em simuladores microscópicos podem ser definidos diferentes detalhes de um cenário tais como pontos de cruzamento, número de vias, semáforos, fluxos de veículos, momentos de partida de fluxo e tipos de veículos;
    - Uso de ferramenta de modelagem de cidades: existem simuladores de tráfego que

podem receber arquivos descritores de topologia de uma cidade. Tais arquivos podem descrever características tais como ruas, avenidas, prédios, casas, elevações e depressões. Uma das formas de obter um arquivo de topologia é através do OpenStreetMaps (ARELLANO; MAHGOUB, 2013). Ele permite navegar pelas regiões do globo e escolher a porção a ser salva em algum formato de banco de dados que seja aceito pelo simulador de tráfego. Um dos simuladores de tráfego capazes de receber tais arquivos como entrada é o VEINS (ARELLANO; MAHGOUB, 2013).

A simulação deste protocolo em uma cidade é importante, uma vez que diversas aplicações para VANETs funcionam em cidades e realizam *broadcast* de informações (KARGL, 2006). O E-ProbT foi modelado de forma a ser aplicado em diferentes cenários. A sua aplicação no cenário *Manhattan* é perfeitamente viável, requerendo apenas um estudo para analisar a necessidade de possíveis adequações (provavelmente não são necessárias).

2. Uso de intervalos de espera na camada de enlace: os intervalos de espera utilizados no processo de decisão de encaminhamento, DT1 e DT2, são utilizados na camada de aplicação. Considerando que a camada de enlace também realiza intervalos de espera, é possível analisar um trabalho em conjunto dessas duas camadas, o que é conhecido na literatura como *cross-layer*.

   O *cross-layer* pode reduzir possíveis esperas desnecessárias, o que é altamente indicado para *broadcast* de informações. Assim, deve ser realizado um estudo para analisar a viabilidade do *cross-layer* entre a camada de aplicação e a camada de enlace, de forma a minimizar os intervalos de espera.

3. Estudo de uma melhor configuração de parâmetros para o E-ProbT: a simulação apresentada neste trabalho foi baseada no conjunto de parâmetros exibido na Tabela 4. Parte dos parâmetros utilizados foi escolhida por sua utilização em outros trabalhos da literatura de VANETS ((WISITPONGPHAN et al., 2007), (ROBERTO et al., 2011), (PAULA et al., 2014) e (LIMA et al., 2015)).

   Uma outra parte dos parâmetros foi escolhida de forma a dar prioridade a determinada característica, entretanto, definida de forma empírica uma vez que a sua utilização foi proposta por este trabalho. Assim, cabe um estudo para analisar uma melhor configuração de parâmetros para o E-ProbT.

4. Utilização de outras métricas para comparação de protocolos: os protocolos comparados neste trabalho foram analisados considerando 3 métricas:

- Taxa de entrega de pacotes normalizada;
- Número de saltos;
- Taxa de redundância.

Tais métricas foram escolhidas na definição da proposta, por representarem as características alvo do trabalho. Entretanto, existem outras métricas que podem ser usadas para comparação de protocolos. Um estudo sobre métricas de protocolos para *broadcast* de informações em VANETs é apresentado em (PANICHPAPIBOON; PATTARA-ATIKOM, 2012). De acordo com a necessidade, determinado implementador pode desejar determinada característica no seu protocolo de *broadcast*. Assim, cabe a ele a inserção de uma métrica que reflita a sua necessidade. Dentre tais métricas podem ser citadas o intervalo médio para entrega de pacotes e a quantidade média de processamento.

REFERÊNCIAS

ALSHAER, H.; HORLAIT, E. An optimized adaptive broadcast scheme for inter-vehicle communication. In: IEEE. **Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st**. [S.l.], 2005. v. 5, p. 2840–2844.

ALVES, C. da C.; HENNING, E.; KONRATH, A. C.; WALTER, O. M. F. C.; SAMOHYL, R. W. A estatística média móvel exponencialmente ponderada para o controle preditivo, monitoramento e ajuste de processos. **Congresso Latino-Iberoamericano de Investigación Operativa (CLAIO). Simpósio Brasileiro de Pesquisa Operacional (SBPO)**, 2012.

ALVES, R. d. S.; CAMPBELL, I. d. V.; COUTO, R. d. S.; CAMPISTA, M. E. M.; MORAES, I. M.; RUBINSTEIN, M. G.; COSTA, L. H. M.; DUARTE, O. C. M.; ABDALLA, M. Redes veiculares: Princípios, aplicações e desafios. **Minicursos do Simpósio Brasileiro de Redes de Computadores, SBRC**, 2009.

ARELLANO, W.; MAHGOUB, I. TrafficModeler extensions: A case for rapid VANET simulation using, OMNET++, SUMO, and VEINS. In: **10th International Conference on High Capacity Optical Networks and Enabling Technologies (HONET-CNS 2013)**. Magosa, Cyprus: IEEE, 2013. p. 109–115.

BECHLER, M.; JAAP, S.; WOLF, L. An optimized tcp for internet access of vehicular ad hoc networks. In: **NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems**. [S.l.]: Springer, 2005. p. 869–880.

ČISAR, P.; ČISAR, S. M. Optimization methods of ewma statistics. **Acta Polytechnica Hungarica**, v. 8, n. 5, p. 73–87, 2011.

FUDENBERG, D.; TIROLE, J. **Game Theory**. [S.l.]: The MIT Press, 1991.

JACOBSON, V. Congestion avoidance and control. In: ACM. **ACM SIGCOMM computer communication review**. [S.l.], 1988. v. 18, n. 4, p. 314–329.

KARGL, F. Vehicular communications and vanets. In: **Talks 23rd Chaos Communication Congress**. [S.l.: s.n.], 2006.

KUMAR, R.; DAVE, M. A review of various vanet data dissemination protocols. **International Journal of u-and e-Service, Science and Technology**, v. 5, n. 3, p. 27–44, 2012.

KUROSE, J. F.; ROSS, K. W. **Redes de Computadores e Internet: Uma abordagem top-down, 5ª Edição**. [S.l.]: Pearson Education Inc., 2010. ISBN 978-85-88639-97-3.

LIM, H.; KIM, C. Flooding in wireless ad hoc networks. **Computer Communications**, Elsevier, v. 24, n. 3, p. 353–363, 2001.

LIMA, D.; PAULA, M. R. P.; ROBERTO, F. M.; CARDOSO, A. R.; Celestino Júnior, J. ProbT: a temporal probabilistic protocol to mitigate the broadcast storm problem in VANETs. In: **International Conference on Information Networking 2015 (ICOIN 2015)**. Siem Reap, Cambodia: [s.n.], 2015.

LIMA, D. S.; PAULA, M. R. P.; ROBERTO, F. M.; CARDOSO, A. R.; Celestino Júnior, J. **ProbT: um protocolo probabilístico e temporal para mitigação do problema broadcast storm em VANETs**. 2014. Trabalho de conclusão de curso de graduação. Universidade Estadual do Ceará. Ceará.

MOSER, L. E.; MELLIAR-SMITH, P. Probabilistic analysis of message forwarding. In: IEEE. **Computer Communications and Networks (ICCCN), 2013 22nd International Conference on**. [S.l.], 2013. p. 1–8.

PAL, S. K.; MITRA, S. Multilayer perceptron, fuzzy sets, and classification. **Neural Networks, IEEE Transactions on**, IEEE, v. 3, n. 5, p. 683–697, 1992.

PANICHPAPIBOON, S.; FERRARI, G. Irresponsible forwarding. In: IEEE. **ITS Telecommunications, 2008. ITST 2008. 8th International Conference on**. [S.l.], 2008. p. 311–316.

PANICHPAPIBOON, S.; PATTARA-ATIKOM, W. A review of information dissemination protocols for vehicular ad hoc networks. **Communications Surveys & Tutorials, IEEE**, IEEE, v. 14, n. 3, p. 784–798, 2012.

PAULA, M. R. P.; LIMA, D. S.; ROBERTO, F. M.; CARDOSO, A. R.; JR, J. C. A technique to mitigate the broadcast storm problem in vanets. **International Conference on Networks (ICN 2014)**, p. 253, 2014.

POONIA, R. C.; SINGH, V. Performance evaluation of radio propagation model for vehicular ad hoc networks using vanetmobisim and ns-2. **International Journal of Distributed and Parallel Systems (IJDPS)**, v. 3, n. 4, p. 145–155, 2012.

RITO, P. F. V. **Receptor SDR para comunicações DSRC**. Dissertação (Mestrado) — Universidade de Aveiro, 2011.

ROBERTO, F. M. **Um mecanismo de difusão para redes VANETs baseado em Cinemática e Teoria dos Jogos**. Dissertação (Mestrado) — Universidade Estadual do Ceará, 2010.

ROBERTO, F. M.; CELESTINO, J.; SCHULZRINNE, H. Using a symmetric game based in volunteer's dilemma to improve vanets multihop broadcast communication. In: IEEE. **Personal Indoor and Mobile Radio Communications (PIMRC), 2011 IEEE 22nd International Symposium on**. [S.l.], 2011. p. 777–782.

SHIPMAN, S. **Camada LLC**. 2000. Disponível em: <http://www.houseofhum.com/stephen/library/network_primer/ch02.html>.

SINGH, P. K. Influences of tworayground and nakagami propagation model for the performance of adhoc routing protocol in vanet. **International Journal of Computer Applications**, International Journal of Computer Applications, 244 5 th Avenue,# 1526, New York, NY 10001, USA India, v. 45, n. 22, 2012.

SLAVIK, M.; MAHGOUB, I. Applying machine learning to the design of multi-hop broadcast protocols for vanet. In: IEEE. **Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International**. [S.l.], 2011. p. 1742–1747.

SU, J.-H.; WANG, T.-P. Delay-aware routing based on game theory in vehicular wireless networks. **International Conference on Advanced Information Technologies**, 2013.

TONGUZ, O.; WISITPONGPHAN, N.; BAI, F.; MUDALIGE, P.; SADEKAR, V. Broadcasting in vanet. In: IEEE. **2007 mobile networking for vehicular environments**. [S.l.], 2007. p. 7–12.

WATSON, J. **Strategy: An Introduction to Game Theory**. [S.l.]: W. W. Norton & Company; 2 edition, 2007.

WEESIE, J. Asymmetry and timing in the volunteerś dilemma. **Journal of Conflict Resolution**, v. 37, n. 3, p. 569–590, 1993.

WEGENER, A.; HELLBRUCK, H.; FISCHER, S.; SCHMIDT, C.; FEKETE, S. Autocast: An adaptive data dissemination protocol for traffic information systems. In: IEEE. **Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th**. [S.l.], 2007. p. 1947–1951.

WISITPONGPHAN, N.; TONGUZ, O. K.; PARIKH, J.; MUDALIGE, P.; BAI, F.; SADEKAR, V. Broadcast storm mitigation techniques in vehicular ad hoc networks. **Wireless Communications, IEEE**, IEEE, v. 14, n. 6, p. 84–94, 2007.

YOUSEFI, S.; MOUSAVI, M. S.; FATHY, M. Vehicular ad hoc networks (vanets): challenges and perspectives. In: IEEE. **ITS Telecommunications Proceedings, 2006 6th International Conference on**. [S.l.], 2006. p. 761–766.

ZHANG, J. A survey on trust management for vanets. In: IEEE. **Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on**. [S.l.], 2011. p. 105–112.