

Uma Avaliação Experimental de Desempenho do Roteamento Multicaminhos em Redes Veiculares

Ricardo Macedo, Robson Melo, Leonardo Melniski,
Aldri Santos, Michele Nogueira

Núcleo de Redes Sem Fio e Redes Avançadas (NR2)
Universidade Federal do Paraná (UFPR)
Caixa Postal 19081 - Curitiba – PR – Brasil

{rmacedo, rgmelo, lem09, aldri, michele}@inf.ufpr.br

Abstract. *Vehicular ad-hoc networks (VANETs) are composed of mobile devices wireless communication embedded in automobiles to help drivers on the highways. Due to frequent changes in network topology, one of the main challenges of VANETs is to establish routes between nodes. This paper presents an experimental performance evaluation of single path and multipath routing protocols. The chosen metrics for this study are the rate of packet loss and jitter. Results show that the multipath protocol presents a packet loss rate four times lower than the single path approach in static scenarios, and over three times smaller in mobile settings. However, in all scenarios a higher jitter is detected for the multipath approach.*

Resumo. *As redes veiculares ad-hoc (VANETs) são compostas por dispositivos móveis de comunicação sem fio embarcados em automóveis para auxiliar os condutores nas rodovias. Devido às frequentes alterações da topologia da rede, um dos principais desafios das VANETs consiste no estabelecimento de rotas entre os nós. Este trabalho apresenta uma avaliação de desempenho experimental comparando protocolos de roteamento com base em múltiplos caminhos e melhor caminho em cenários com baixa mobilidade. As métricas escolhidas neste estudo são a taxa de perda dos pacotes e jitter. A média dos resultados obtidos na análise demonstram que o protocolo multicaminho apresenta uma taxa de perda de pacotes cerca de quatro vezes menor que a abordagem de melhor caminho no cenário estático, e aproximadamente três vezes menor no cenário móvel. Contudo, em todos os cenários, um jitter mais elevado foi detectado na abordagem multicaminho.*

1. Introdução

As redes veiculares ou *Vehicular Ad-hoc NETWORK* (VANETs) desempenham um papel importante no suporte aos sistemas inteligentes de transporte (ITS). Essas redes são formadas por veículos equipados com dispositivos de comunicação sem fio e pontos de acesso fixos, distribuídos às margens de ruas ou estradas. As conexões entre os nós da VANET podem ser do tipo veículos a veículo, veículo a ponto de acesso ou híbrido [Faezipour et al. 2012]. Com a infraestrutura provida por essas redes, inúmeros serviços e aplicações poderão ser beneficiados, como é o caso dos ITS, que operam em um ambiente formado por usuários no trânsito.

Através da infraestrutura das VANETs será possível prover conectividade entre veículos em diferentes rodovias por meio da Internet, e assim disseminar informação sobre as condições de uma estrada em outras rodovias, permitindo aos condutores repensar sobre a escolha de seu trajeto devido a trecho congestionado [Park et al. 2010]. Esse cenário pode trazer outros benefícios, tais como, o gerenciamento de rodovias e de transporte coletivo, o controle de tráfego, o gerenciamento de serviços de emergência, o rastreamento de frotas de veículos de carga, de transporte coletivo, assim como o provimento de informações atuais e dinâmicas que apoiem os usuários em suas decisões.

Contudo, é um desafio transmitir de maneira robusta e eficaz informações em VANETs. A alta mobilidade e as variações de velocidade dos veículos ocasionam frequentes desconexões, fragmentações da rede e atrasos na entrega de pacotes [Sofra et al. 2011]. Dessa forma, corre-se o risco de enviar pacotes de dados através de rotas inválidas, ocasionando retransmissões, atrasos e perdas de pacotes [Bisht et al. 2012].

Os protocolos de roteamento são responsáveis por estabelecer os caminhos que devem ser utilizados para que duas entidades se comuniquem. Existem duas principais abordagens de protocolos de roteamento aplicáveis em VANETs, os protocolos de caminho único e os de multicaminhos. Os protocolos de caminho único escolhem o melhor caminho com base em diferentes métricas, obtidas por parâmetros dos caminhos existentes entre um nó origem e um destino. Enquanto que os protocolos multicaminhos escolhem n entre x caminhos existentes a partir de um emissor e um receptor. Devido às frequentes mudanças na topologia das VANETs, percebe-se que os protocolos multicaminhos apresentam estratégias de roteamento mais eficazes para este tipo de rede [Spaho et al. 2012].

Entre os protocolos de roteamento de caminho único, o *Optimized Link State Routing protocol* (OLSR) se apresenta como uma solução para VANETs [Spaho et al. 2012]. O OLSR mantém uma tabela de roteamento em cada nó da rede e utiliza mensagens de controle *TC* e *HELLO* para encontrar a topologia da rede e estabelecer as rotas. Uma característica do OLSR é a eleição de nós intermediários, chamados de *Multi Point Relays* (MPR). Utilizando os MPRs é possível selecionar a melhor rota através de múltiplos saltos. Todavia, notam-se iniciativas para otimizar o OLSR para a alta dinamicidade de topologias como é o caso das VANETs [Toutouh et al. 2012]. Neste sentido, o emprego de multicaminhos no OLSR como o protocolo *MultiPath Optimized Link State Routing* (MP-OLSR) vem sendo bastante discutido na literatura para prover uma comunicação mais robusta, confiável, que diminua o congestionamento da rede e reduza o atraso na entrega das mensagens. Porém, a maioria das iniciativas ainda são embrionárias, utilizadas apenas em simulações [Yi et al. 2011, Radu et al. 2012].

Este trabalho apresenta uma avaliação experimental do desempenho de uma VANET, utilizando os protocolos de roteamento OLSR e MP-OLSR. A principal contribuição consiste em comparar o desempenho desses protocolos em um ambiente real. Devido ao alto impacto na transferência de dados, as métricas escolhidas para o estudo foram a taxa de perda de pacotes e *jitter*. O estudo foi conduzido em dois cenários, com baixa mobilidade e sem mobilidade. Os resultados obtidos demonstram que o MP-OLSR apresentou uma taxa de perda de pacotes cerca de quatro vezes menor que o OLSR no cenário estático, e aproximadamente três vezes menor no cenário móvel. No entanto, o OLSR apresentou um *jitter* menor nos dois cenários em relação ao MP-OLSR.

Este artigo está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 descreve o protocolo OLSR, e a Seção 4 explica o funcionamento do MP-OLSR. A Seção 5 detalha os experimentos. A Seção 6 conclui o artigo.

2. Trabalhos Relacionados

As principais características das redes VANETs são as constantes fragmentações da rede e frequentes desconexões que causam atrasos na comunicação [Sofra et al. 2011]. Devido a essas características, o roteamento de pacotes neste tipo de rede torna-se um desafio. Contudo, inúmeros trabalhos na literatura têm discutido a aplicabilidade e desempenho de protocolos de roteamentos já existentes nas VANETs. Dentre as abordagens de maiores evidências estão os protocolos de roteamento de caminho único e os de multicaminhos.

O trabalho de [Toutouh et al. 2012] apresenta um estudo comparativo do desempenho entre dois protocolos de caminho único em VANETs. Essa pesquisa analisou o OLSR e *Destination-Sequenced Distance-Vector* (DSDV) utilizando as métricas de taxa de entrega de pacotes e a vazão. Nesse trabalho, foram realizadas simulações com o simulador de redes NS, versão 3 e o modelo de mobilidade CAVENET (*Cellular Automaton based VEHicular NETWORK*). Os resultados desta pesquisa indicaram que o OLSR apresentou um desempenho superior ao DSDV, tanto na taxa de entrega de pacotes quanto na vazão, pois a duração do enlace e a estabilidade do caminho do OLSR tornou-se superior devido ao mecanismo *Multi Point Relays* (MPR) que reduz o atraso no roteamento.

Um outro estudo sobre o desempenho de protocolos de caminho único é o de [Bisht et al. 2012]. Nesse trabalho os autores argumentam a necessidade de adaptação do protocolo OLSR devido à constante mudança de topologia da rede, o que influencia diretamente no encaminhamento dos pacotes. Ao considerar a adaptação e a autoconfiguração como um problema de otimização, os autores alegam que uma possível solução consiste em estratégias de otimizações com base em metaheurísticas. A prova de conceito destas estratégias foi realizada através de simulações no simulador de redes NS, versão 2. Os resultados obtidos consistiram em configurações automáticas e customizadas do OLSR para diferentes cenários de redes VANETs.

Em [Yi et al. 2011] os autores propõem um protocolo multicaminho denominado MP-OLSR. O MP-OLSR objetiva incrementar a resiliência contra falhas nos nós e enlaces através de múltiplos caminhos disjuntos. Para que esse protocolo empregasse multicaminhos em topologias de redes esparsas e densas, foram realizadas modificações no algoritmo de *Dijkstra*. Uma característica do MP-OLSR é que as rotas são calculadas sob demanda, onde os caminhos são obtidos a partir do nó de origem. O estudo foi validado através de simulações com cenários móveis e experimentações apenas em ambientes estáticos. Os resultados revelaram que o MP-OLSR, quando aplicado em redes móveis, satisfaz os requisitos de aplicações críticas.

Em [Radu et al. 2012], apresentou-se um estudo com a aplicação de um protocolo multicaminho em VANETs. No trabalho em questão foi proposto um sistema de monitoramento para coletar os níveis de poluição sonora através de sensores que usaram a infraestrutura VANET para mapear as regiões mais afetadas e encaminhar essas informações para uma base de dados na Internet. Para melhorar a confiabilidade da transmissão, evitar o congestionamento do tráfego de dados e reduzir o atraso na entrega dos pacotes, foi empregado o protocolo MP-OLSR. A validação dos resultados deste trabalho deu-

se através de simulação utilizando o simulador Qualnet. Os resultados mostraram que a utilização de múltiplos caminhos esparsamente separados provê melhor desempenho em termos de taxa de distribuição de pacotes e atraso em VANETs.

Embora os trabalhos mencionados discutam o emprego do protocolo MP-OLSR em redes VANETs, percebe-se que a grande maioria realiza apenas avaliações que utilizam técnicas de simulação. Os estudos realizados não consideram a implantação deste protocolo em cenários reais de VANETs, para uma análise mais detalhada de seu desempenho. Métricas como perdas de pacotes e vazão são avaliadas através de simulações, que podem ter resultados diferentes de cenários reais com condições adversas de propagação de sinal, interferência e entrega de pacotes.

3. Optimized Link State Routing

O *Optimized Link State Routing* (OLSR) é um protocolo de roteamento proativo, que estabelece as rotas para o encaminhamento de pacotes de dados, antes de enviá-los [Johnson and Hancke 2009]. O OLSR reduz a sobrecarga de inundação de informações sobre o estado dos enlaces, exigindo que menos nós encaminhem as informações adiante. Um *broadcast* do nó X é encaminhado somente se ele for um *multi point relays* (MPR). Os MPRs de um nó são seus vizinhos a um salto de distância que podem alcançar qualquer nó a dois saltos. Assim, o nó de origem deve alcançar qualquer nó a dois enlaces de distância dos MPRs, mesmo que para isso utilizem outros MPRs [Johnson and Hancke 2009].

Cada nó transmite *beacons* periodicamente para sua lista de vizinhos. Deste modo, todos os nós conhecem seus vizinhos de dois saltos, a fim de escolher o MPR. A Figura 1 ilustra como o protocolo de roteamento OLSR difunde o encaminhamento das mensagens do nó C através da rede, via os MPRs selecionados.

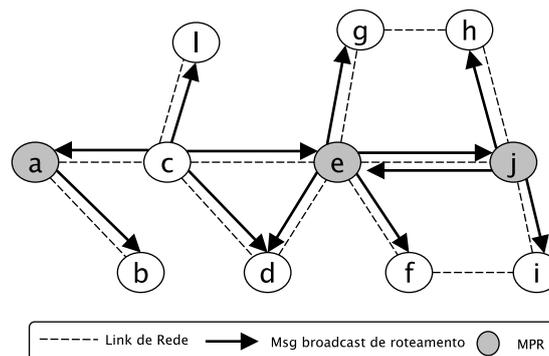


Figura 1. Escolha do melhor caminhos com o OLSR

Cálculo da rota e escolha dos enlaces com OLSR

O OLSR utiliza duas diferentes métricas de roteamento para a escolha dos enlaces:

1. *Métrica de roteamento de histerese*¹ - Calcula a qualidade do caminho entre os nós, a fim estabilizar a rede na presença de múltiplas rotas. O histórico do caminho é calculado usando um processo iterativo. Se q_n é a qualidade do caminho depois

¹Tendência de um sistema de conservar suas propriedades na ausência de um estímulo que a gerou.

de n pacotes e h é a escala constante de histerese entre 0 e 1, então, a qualidade do caminho para cada pacote recebido com sucesso é definido como:

$$q_n = (1 - h)q_{n-1} + h \quad (1)$$

Assim, q_0 começará como 0 e o valor de q_n vai sempre estar entre 0 e 1. Para cada pacote não recebido a qualidade do enlace será:

$$q_n = (1 - h)q_{n-1} \quad (2)$$

Quando um caminho exceder um valor limite máximo, q_{max} , o mesmo é considerado estável e quando a qualidade do caminho cair abaixo do limite mínimo, q_{min} , entende-se como uma falha no caminho.

2. *Contagem de transmissão esperada (ETX)* - Consiste em uma métrica de roteamento incorporada ao OLS existente no padrão oficial do protocolo. Ela calcula o número de retransmissões necessárias para um pacote chegar ao destino. A qualidade do enlace, LQ , é a fração de pacotes que foram recebidos com sucesso dos seus vizinhos dentro de um intervalo de tempo. A qualidade do enlace do vizinho, NLQ , é a fração de pacotes recebidos com sucesso por um vizinho dentro de um dado intervalo de tempo. Desta forma, o ETX é calculado por:

$$ETX = \frac{1}{LQ + NLQ} \quad (3)$$

Em caminhos de múltiplos saltos os valores de ETX de cada salto são somados para o cálculo do ETX do caminho completo.

4. MultiPath Optimized Link State Routing

O *MultiPath Optimized Link State Routing* (MP-OLSR) é um protocolo híbrido (entre o modelo de roteamento proativo e reativo) e de múltiplos caminhos. Foi desenvolvido com base no OLSR que agrega as vantagens do mecanismo de MPR para inundar a rede com informações de controle do tráfego. No MP-OLSR, o comportamento proativo do OLSR é alterado para uma detecção de rotas sob demanda. O protocolo é composto por duas fases: detecção da topologia e cálculo das rotas.

1. **Detecção da Topologia.** Para a detecção da topologia da rede, o MP-OLSR aproveita os métodos do OLSR. As mensagens de controle *HELLO* e *TC* são trocadas constantemente entre os nós, para manter a topologia atualizada. Entretanto, os nós MP-OLSR armazenam tabelas de roteamento. Assim o protocolo calcula múltiplas rotas quando pacotes de dados precisam ser enviados.
2. **Cálculo da rota.** Nesta fase os nós calculam múltiplos caminhos sob demanda, com base em informações obtidas na fase de detecção de topologia.

Para o cálculo da rota mais precisamente, o protocolo implementa o algoritmo de múltiplos caminhos de Dijkstra, que retorna as rotas mais curtas. Os caminhos obtidos podem ser agrupados em duas categorias:

- Disjuntos: Os caminhos do tipo nós disjuntos não compartilham nós, exceto para nós de origem ou destino. Os caminhos de enlaces disjuntos podem compartilhar alguns nós, mas todos os enlaces são diferentes.
- Inter-relacionados: Os caminhos podem compartilhar diversos nós e enlaces.

Cálculo da rota com algoritmo *Multipath Dijkstra*

Para a explicação do algoritmo, assume-se uma rede *ad hoc* representada por um grafo $G = (V, E, c)$, onde V é o conjunto de vértices, E é o conjunto de arestas e $c : V \rightarrow \mathcal{R}^{*+}$ é uma função de custo estritamente positiva. G é um grafo simples, ou seja, não direcionado, livre de laços e não aceita múltiplas arestas para dois vértices distintos.

Para cada nó de origem da rede, o protocolo guardará uma variável booleana $flag_i$ para identificar a validade das rotas para o nó correspondente. Inicialmente a $flag_i$ é *falsa* e significa que não existe uma rota para aquele nó. Quando existir uma requisição de rota para um certo nó i , o nó origem primeiramente irá verificar a $flag_i$.

- Se a $flag_i$ for *falsa*, o nó irá executar o Algoritmo 1 para obter os múltiplos caminhos para o nó i , salvar estas informações na tabela de roteamento de múltiplos caminhos e atualizar a $flag_i$ para *verdadeira*.
- Caso a $flag_i$ seja *verdadeira*, o nó encontrará uma rota válida para o nó i na tabela de roteamento de múltiplos caminhos.

Ao receber novas mensagens de *TC* ou *HELLO* e resultar em uma mudança da topologia rede, todas as *flags* retornaram para o valor *false*. A obtenção de N caminhos de uma origem o para um destino d é detalhada no Algoritmo 1.

Algoritmo 1 *Multipath Dijkstra*

```

1: MultipathDijkstra( $o, d, G, N$ )
2:  $C_1 \leftarrow C$ 
3:  $G_1 \leftarrow G$ 
4: para ( $i \leftarrow 1$  to  $N$ ) faça
5:    $ArvoreOrigem_i \leftarrow Dijkstra(G_i, o)$ 
6:    $P_i \leftarrow CapturaCaminho(ArvoreOrigem_i, d)$ 
7:   para todo arcos  $e$  em  $E$  faça
8:     se  $e$  está em  $P_i$  OU  $Reverse(e)$  está em  $P_i$  então
9:        $C_{i+1}(e) \leftarrow f_p(C_i(e))$ 
10:    senão se o vértice  $Head(e)$  está em  $P_i$  então
11:       $C_{i+1}(e) \leftarrow f_e(C_i(e))$ 
12:    senão
13:       $C_{i+1}(e) \leftarrow C_i(e)$ 
14:    fim se
15:  fim para
16:   $G_{i+1} \leftarrow (R, E, C_{i+1})$ 
17: fim para
18: retorna ( $P_1, P_2, \dots, P_N$ )

```

O algoritmo é aplicado no grafo $G = (V, E, c)$, dois vértices $(o, d) \in E$ e um número estritamente positivo N . Isto provê N -tuplas (P_1, P_2, \dots, P_N) de caminhos (o, d) extraídos do grafo G . $Dijkstra(G_i, o)$ é o algoritmo padrão *Dijkstra* que extrai uma árvore com caminhos mais curtos de um vértice n em um grafo G ; $CapturaCaminho(ArvoreOrigem, n)$ é uma função que extrai o caminho mais curto para n , de uma árvore de origem $ArvoreOrigem$; $Reverse(e)$ retorna a aresta oposta de e ; $Head(e)$ provê o primeiro vértice da aresta e .

As funções incrementais $f_p : \mathcal{R}^{*+} \leftarrow \mathcal{R}^{*+}$ e $f_e : \mathcal{R}^{*+} \leftarrow \mathcal{R}^{*+}$ são usadas a cada passo para obter caminhos disjuntos entre o e d . Os valores destas funções são definidas

pelo usuário. A função f_p é usada para incrementar o custo das arestas que pertencem ao caminho P_i . Isto influenciará os caminhos futuros a usar arestas diferentes. O f_e é usado para incrementar os custos das arestas que levam para os vértices do caminho anterior P_i . Considerando que id é uma função identidade, as seguintes possibilidades são prováveis:

- se $id = f_e < f_p$, caminhos tendem para arestas disjuntas;
- se $id < f_e = f_p$, caminhos tendem para vértices disjuntos;
- se $id < f_e < f_p$, caminhos também tendem para vértice disjuntos, mas quando não for possível tendem para aresta disjunta.

A Figura 2 ilustra a execução do Algoritmo 1 em uma rede onde o nó o precisa encontrar múltiplos caminhos até o nó d , considerando que as funções $f_p(c)$ e $f_e(c)$ possuem valores definidos pelo usuário como $3c$ e c respectivamente.

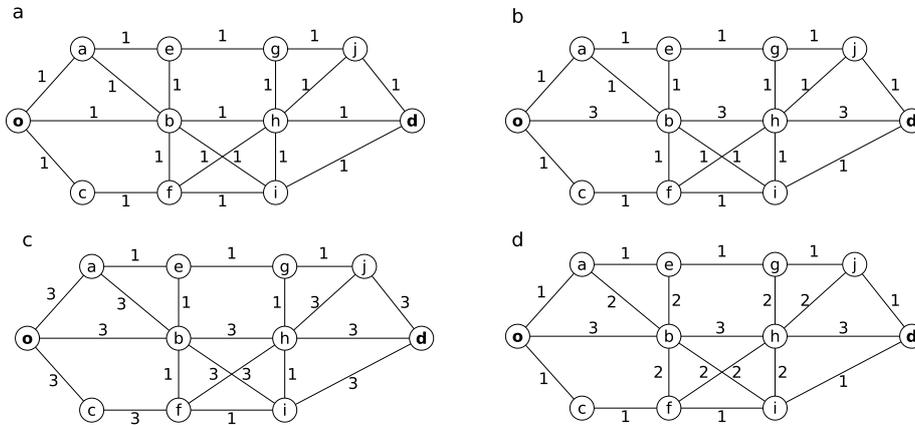


Figura 2. Construção de múltiplos caminhos com o protocolo MP-OLSR

Na Figura 2(a), o nó o precisa descobrir múltiplos caminhos até o nó d . Considerando o custo inicial c de cada aresta com sendo 1. Assim, a função $f_p(c) = 3c$ e $f_e(c) = c$. Deste modo, apenas nós no caminho encontrado tem seu custo aumentado. Na primeira execução do algoritmo o caminho será $o \rightarrow b \rightarrow h \rightarrow d$ e as arestas (o, b) , (b, h) e (h, d) aumentarão de um para três seu custo usando a função f_p (Figura 2(b)).

O caminho $o \rightarrow c \rightarrow f \rightarrow i \rightarrow d$ é de vértices disjuntos. Contudo, o caminho $o \rightarrow a \rightarrow b \rightarrow i \rightarrow d$ tem a mesma chance de ser encontrado e é um caminho de arestas disjuntas. O próximo caminho que será encontrado é $o \rightarrow c \rightarrow f \rightarrow h \rightarrow j \rightarrow d$, que também é um caminho de arestas disjuntas, veja na Figura 2(c) o resultado.

Para obter caminhos de vértices disjuntos, altera-se $f_e(c) = 2c$ e mantém-se $f_p(c) = 3c$. Assim, o primeiro caminho encontrado será $o \rightarrow b \rightarrow h \rightarrow d$ e o grafo resultante com os pesos de arestas atualizados pode ser visto na Figura 2(d).

Os próximos caminhos encontrados serão $o \rightarrow c \rightarrow f \rightarrow i \rightarrow d$ e depois $o \rightarrow a \rightarrow e \rightarrow g \rightarrow j \rightarrow d$. Esses caminhos são de vértices disjuntos. O próximo caminho encontrado será $o \rightarrow a \rightarrow b \rightarrow i \rightarrow d$, que é um exemplo de caminho inter-relacionado.

5. Análise de Desempenho

Esta seção apresenta a análise de desempenho experimental dos protocolos OLSR e MP-OLSR em uma VANET. De forma que a Seção 5.1 apresenta as métricas escolhidas. A

Seção 5.2 descreve os parâmetros do sistema e da carga. A Seção 5.3 detalha como os experimentos foram executados. Na Seção 5.4 são apresentados os resultados da análise.

5.1. Métricas Utilizadas

As métricas utilizadas para mensurar o comportamento dos protocolos foram o *jitter* e a Taxa de Perda de Pacotes (TPP). A TPP representa a taxa de pacotes que foram enviados por um nó emissor, mas que por algum motivo não foram recebidos pelo nó receptor. Em uma VANET, os motivos que podem ocasionar uma elevada TPP podem ser a presença de obstáculos entre os nós, fragmentações na topologia da rede e a alta mobilidade dos nós. A definição de TPP é dada por:

$$TPP = \frac{\text{Número de pacotes perdidos}}{\text{Número total de pacotes transmitidos}} \quad (4)$$

Jitter consiste na variação estatística do atraso na entrega de pacotes em uma rede. Considerando que T_j representa o atraso presenciado pelo j -ésimo pacote de uma fila. A diferença do tempo de transmissão entre dois pacotes consecutivos pode ser escrita como:

$$J_j = T_{j+1} - T_j \quad (5)$$

Onde o valor de J_j pode ser positivo ou negativo. Como o *jitter* não pode ser representado através de um valor negativo, então este é obtido através do valor absoluto da variável randômica E

$$J = E[|T_{j+1} - T_j|] \quad (6)$$

Estas métricas foram escolhidas devido a sua forte relação na transmissão de informações do estado atual de rodovias. Por exemplo, na transmissão de um pequeno vídeo que mostra a causa de um congestionamento de uma estrada. Uma alta taxa de perda de pacotes e/ou um elevado *jitter* podem tornar o conteúdo transmitido não interpretável no nó receptor. Conhecendo a definição das métricas selecionadas, na próxima seção será abordado o ambiente utilizado através da descrição dos parâmetros do sistema e da carga onde estas métricas foram empregadas.

5.2. Parâmetros do Sistema e da Carga

O sistema utilizado para análise dos protocolos consistiu em uma VANET composta por cinco nós. As características deste ambiente podem ser representadas através de vários parâmetros. Para melhor apresentá-los, foi realizada a seguinte classificação: parâmetros da carga, parâmetros específicos dos dispositivos, parâmetros específicos dos protocolos e parâmetros genéricos do sistema (Tabela 1(a)).

De acordo com a Tabela 1(a), percebe-se que os parâmetros genéricos do sistema representam as características comuns do cenário utilizado. A frequência de propagação das transmissões utilizada na comunicação entre os nós foi a 2.412 GHz definida pelo padrão 802.11 b. A versão do OLSR empregada foi a 0.56r2, a mesma utilizada em [Yi et al. 2011]. No entanto, observa-se que o *kernel* adotado neste trabalho é mais

(a) Parâmetros Genéricos do Sistema		(b) Parâmetros Específicos dos Dispositivos		
Parâmetro	Valor	Dispositivo	Parâmetro	Valor
Rádio Frequência	2.412 GHz	Computador de Bordo	CPU	VIA 1300 MHz
Padrão	IEEE 802.11 b		Memória	1024 MB
Kernel	3.2.0-29		Chipset	VIA, VT6102
Sistema Operacional	Ubuntu 12.04	Notebook Positivo	CPU	Intel Core i5
Versão do OLSR	Olsrd 0.5.6r2		Memória	4 GB
			Chipset	Atheros, AR8151 v2.0
		Notebook Lenovo	CPU	Intel Core i5
			Memória	4 GB
			Chipset	Realtek, 8111/8168B

Tabela 1. Parâmetros Genérico do Sistema e Específicos dos Dispositivos

recente do que o empregado pelos autores do MP-OLSR. Para tornar possível esta adaptação foram realizadas modificações no código do MP-OLSR, para ajustar a forma que a estrutura de *socket buffer* no kernel do Linux era manipulada.

Os parâmetros específicos dos dispositivos descrevem as configurações peculiares dos cinco equipamentos utilizados no cenário de testes. Apesar dos cinco computadores estarem configurados com a mesma versão de sistema operacional, estes consistiam de equipamentos diferentes, com *drivers* heterogêneos. Foram utilizados três *Notebooks Lenovo*, um *Notebook Positivo* e um *Computador de Bordo*. As configuração destes dispositivos são representadas na Tabela 1(b).

Ao analisar a Tabela 1(b), observa-se ainda que a capacidade de processamento do computador de bordo é muito inferior as dos notebooks, no entanto considerou-se importante incluir este dispositivo no cenário de testes devido à sua aplicabilidade específica. Também foi considerado relevante relatar os *chipsets* dos dispositivos, visando auxiliar possíveis reproduções desses experimentos. As configurações projetadas para os testes com o OLSR e MP-OLSR foram classificadas como parâmetros específicos dos protocolos. Estes parâmetros estão associados com a forma de funcionamento dos protocolos e estão fortemente relacionados com os resultados obtidos. A Tabela 2(a) apresenta as configurações empregadas.

(a) Parâmetros Específicos dos Protocolos		(b) Parâmetros da Carga	
Parâmetro	Valor	Parâmetro	Valor
Intervalo de TC	5 s	Protocolo de transporte	UDP
Intervalo de HELLO	2 s	Porta	104
Intervalo de Atualização	2 s	Tamanho do arquivo	1.1 MB
Número de caminhos MP-OLSR	3	Tamanho do <i>buffer</i> de leitura/escrita	700 B
MP-OLSR f_e	$f_e(c) = 1c$		
MP-OLSR f_p	$f_p(c) = 1c$		

Tabela 2. Parâmetros Específicos do Protocolo e da Carga

Na Tabela 2(a), as três primeiras linhas apresentam a configuração do OLSR. A primeira linha diz respeito ao intervalo de TC, a segunda ao intervalo de HELLO e a

terceira ao intervalo das mensagens de atualização. As mensagens de atualização são requisições de mensagens de HELLO parciais, este parâmetro é empregado quando existe alguma limitação do tamanho dos pacotes de HELLO na rede e um nó recebeu apenas uma parte da mensagem de HELLO de um nó vizinho. Os valores empregados foram extraídos da RFC 3626 do OLSR [Clausen and Jacquet 2003], que indica a utilização de cinco segundos de intervalo para as mensagens de TC, dois segundos de intervalo para mensagens de HELLO e dois segundos para mensagens de atualização.

As configurações do MP-OLSR agregam as configurações do OLSR e as três últimas linhas da Tabela 2(a). Devido ao MP-OLSR empregar o sensoriamento da topologia do OLSR, utilizaram-se as mesmas configurações do OLSR. Além disso, o número de caminhos foi configurado para três e os valores de f_e e f_p para um. Este número de caminhos foi escolhido devido ao cenário de testes utilizado. Enquanto que f_e e f_p foram configurados para que os pesos utilizados no algoritmo *Dijkstra* fossem iguais na descoberta de multicaminhos.

Os parâmetros da carga detalham as características do volume de trabalho aplicado na rede. Na Tabela 2(b), o tipo de carga empregada em nos experimentos consistiu em um tráfego UDP, pela porta 104. O tamanho dos pacotes foi ajustado de 1500 Bytes para 700 Bytes para contornar erros de escrita em *sockets*. O tráfego gerado foi projetado para representar a transferência de um arquivo de vídeo sobre as condições de uma rodovia. Deste modo, os dados transferidos representam um arquivo de vídeo curto. A seguir é descrito como os experimentos foram executados e os fatores variados.

5.3. Execução dos Experimentos

Os experimentos foram realizados no Centro Politécnico da Universidade Federal do Paraná, com três automóveis provendo a mobilidade dos nós. Cada automóvel continha um dispositivo com conexão sem fio. A Figura 3 apresenta alguns registros desta experiência.

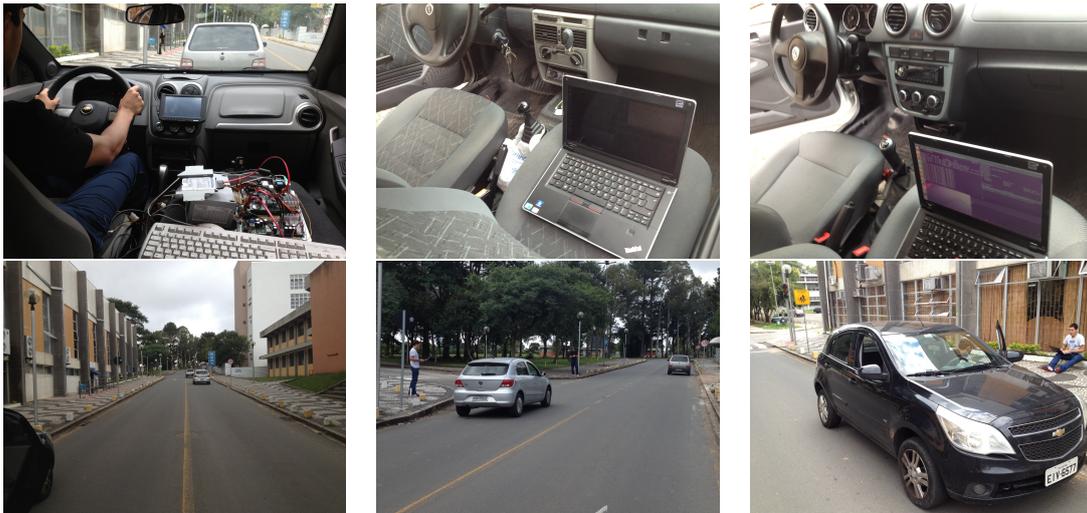


Figura 3. Fotos do Ambiente de Testes

Os cenários foram projetados com cinco computadores configurados com endereço IP estático, onde três computadores representam nós intermediários. Desse modo, foi possível analisar os protocolos para três caminhos diferentes. Devido ao raio de cobertura

dos dispositivos serem limitados, a utilização de altas velocidades e distâncias longas entre os nós móveis tornou-se um empecilho. Dessa forma, a velocidade dos veículos era de aproximadamente cinco Km/h, o que representa uma situação de congestionamento, onde muitas vezes os veículos não se movimentam. Portanto, este estudo aborda um cenário com baixa mobilidade e outro sem mobilidade.

Nos dois cenários, a métrica para sensoriamento da topologia da rede nos protocolos foi a ETX. No cenário estático, ilustrado na Figura 4, o nó emissor foi posicionado à aproximadamente 70 metros do nó receptor. Enquanto que os nós intermediários ficaram a cerca de 30 metros entre os nós comunicantes, alinhados em paralelo para que o custo da utilização de qualquer um dos caminhos fossem semelhantes.

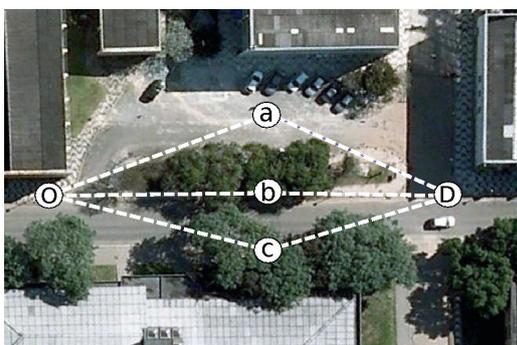


Figura 4. Cenário Estático

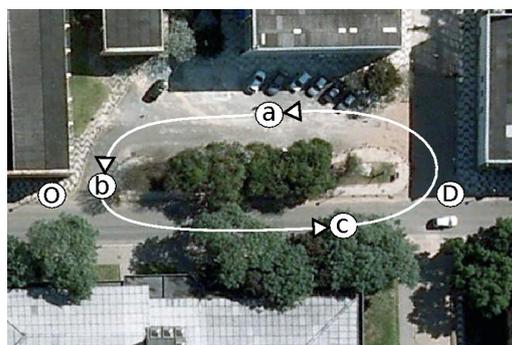


Figura 5. Cenário Móvel

Como visto na Figura 4, o ambiente de experimentos continha árvores que ofereciam resistência para a comunicação entre os nós. A ferramenta *iperf* [Iperf 2013] foi utilizada para geração de tráfego do nó *O* para o nó *D*. Primeiramente transferiu-se dados utilizando o protocolo MP-OLSR e em seguida o OLSR.

O cenário com baixa mobilidade, ilustrado na Figura 5, foi preparado para que os veículos estabelecessem uma velocidade média de 5 km/h. Neste caso, os automóveis circulavam de forma constante, de modo a alterar a cada instante os custos das rotas por cada nó. Para isto, os veículos ficaram afastados aproximadamente 30 metros de distância uns dos outros enquanto trafegavam em apenas uma direção. Neste cenário as árvores também consistiam em um obstáculo para a comunicação. Assim como no primeiro cenário, foram testados os dois protocolos com a ferramenta *iperf*.

5.4. Resultados

Esta seção apresenta os resultados da análise de desempenho, utilizando como métricas a TPP e *jitter*. Nas Figuras 6 e 7 são apresentados os valores correspondentes aos resultados das métricas, nos cenários sem mobilidade e com baixa mobilidade.

Na Figura 6 percebe-se que no cenário sem mobilidade, o *jitter* foi elevado para ambos os protocolos. No OLSR, a causa deste acontecimento foram as frequentes alterações de MPRs, pois os caminhos intermediários possuíam custos semelhantes e mínimas interferências no ambiente ocasionavam a escolha de um novo MPR. No cenário com baixa mobilidade, observa-se que o *jitter* foi menor. Este fato ocorreu em virtude da baixa mobilidade dos nós, que facilitou a escolha do melhor caminho. Em relação ao protocolo MP-OLSR obteve-se um *jitter* elevado devido a utilização dos três caminhos

simultaneamente, visto que o atraso de cada caminho foi considerado. Isto pode ser observado no cenário com baixa mobilidade, onde a média do *jitter* do MP-OLSR é próxima do valor obtido no cenário estático.

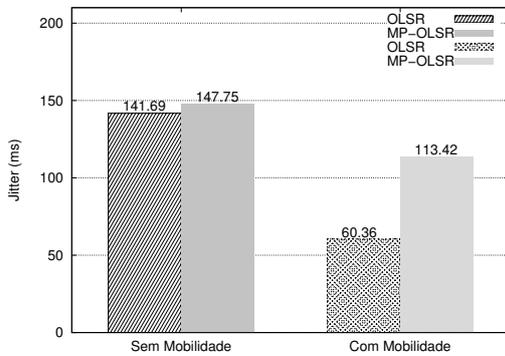


Figura 6. Jitter

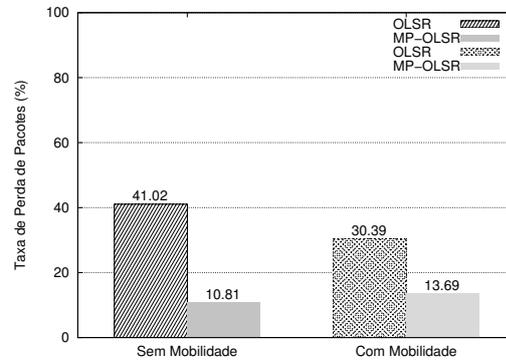


Figura 7. TPP

De forma geral, nota-se que os dois protocolos apresentaram pontos positivos e negativos. No caso do OLSR, observa-se que nos dois cenários a TPP foi considerada alta, 41,02% no cenário sem mobilidade e 30,39% no cenário com mobilidade, contra 10,81% e 13,39% do MP-OLSR. No entanto, o *jitter* obtido com o OLSR foi um pouco menor do que o alcançado pelo MP-OLSR, 141,69 ms no cenário estático e 60,36 ms no cenário móvel, contra 147,75 ms e 113,42 ms, respectivamente. Com base nestes resultados, percebe-se que o OLSR apresentou uma TPP cerca de quatro vezes maior que o MP-OLSR no cenário estático e aproximadamente três vezes maior no cenário móvel.

Para melhor entender, é necessário considerar a influência de interferências e analisar como o tráfego dos pacotes foi dimensionado através dos nós intermediários. Com o objetivo de ratificar a TPP dos protocolos analisados, as Figuras 8 e 9 ilustram os registros coletados da ferramenta *Wireshark*[Wireshark 2013] do cenário sem mobilidade.

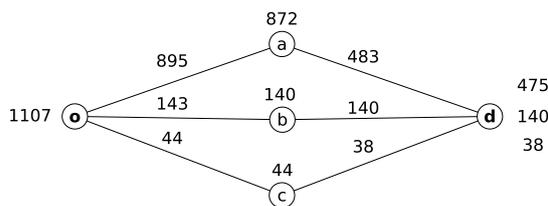


Figura 8. OLSR Estático

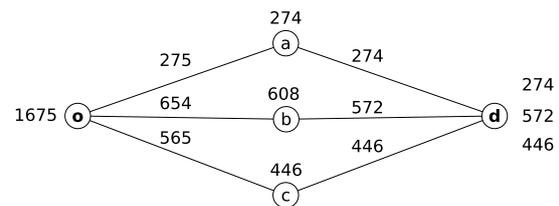


Figura 9. MP-OLSR Estático

Ao observar a Figura 8, é possível visualizar que o OLSR encaminhou a maioria dos pacotes através do nó intermediário *a*. Observa-se que o nó *o* emitiu 895 pacotes para *a*, onde 23 pacotes foram descartados, ao repassar para o nó *d*, nota-se que 389 pacotes foram descartados, representando somente neste caminho uma perda de 55%. Na comunicação por intermédio do nó *b*, foram perdidos apenas 3 pacotes. Enquanto que através do nó *c* perderam-se apenas 6 pacotes.

A Figura 9 apresenta a TPP em cada nó do cenário sem mobilidade com o protocolo MP-OLSR. Nota-se que a distribuição dos pacotes foi de fato realizada por multicaminhos. Por meio do nó *a* foram transmitidos 275 pacotes, com perda de apenas 1. No

caminho *b* foram perdidos 46 pacotes, 5,94%, enquanto que por intermédio do caminho *c* perderam-se 84 pacotes, 21,07%. Esta mesma análise foi realizada no cenário com baixa mobilidade. As Figuras 10 e 11 apresentam estes resultados.

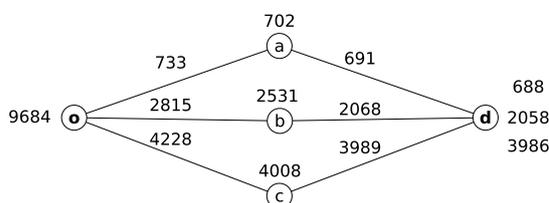


Figura 10. OLSR Móvel

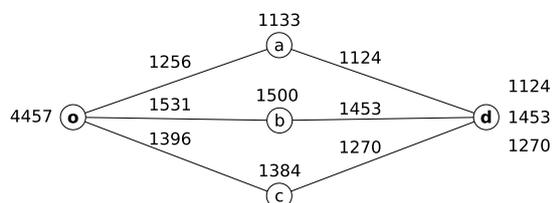


Figura 11. MP-OLSR Móvel

A Figura 10, mostra que através do nó *a* foram perdidos 45 pacotes, 6,14%, pelo OLSR. Através do nó *b* a quantidade de pacotes descartados foi 747, 26,54%. Enquanto que através do nó *c* foram 242 pacotes, 5,73%. Já a perda de pacotes do MP-OLSR, ilustrada na Figura 11, observa-se uma TPP de 10,51% por intermédio do nó *a*, 5,1% através do nó *b* e 9,03% através do nó *c*.

Os resultados mostraram uma alta TPP do OLSR. Contudo, esperava-se que a maioria dos pacotes emitidos fossem encaminhados através de uma única rota, resultando em uma TPP baixa. Uma possível explicação consiste na redução da métrica ETX de um caminho, quando este é usado para transferência de dados. Isso ocorre porque o tráfego de dados compete com os pacotes de controle utilizados para medir a qualidade dos enlaces.

Este fato explica o motivo pelo qual no cenário estático o OLSR obteve uma TPP mais elevada e a razão pela qual no MP-OLSR a TPP foi menor nos dois cenários. Como no cenário estático os nós estavam alinhados em paralelo, qualquer interferência mínima disparava a escolha de um novo melhor caminho. Verificou-se neste cenário cerca de 14 variações de MPRs em um intervalo de cinco minutos. Enquanto que devido ao emprego de multicaminhos, o MP-OLSR nos dois cenários utilizou os três caminhos simultaneamente, gerando uma TPP inferior. Ao comparar com o estudo de [Yi et al. 2011], os resultados obtidos foram semelhantes em termos de TPP.

6. Conclusão

Este trabalho apresentou uma análise experimental do desempenho de roteamento multicaminhos em um ambiente real de VANET. A média dos resultados da análise demonstram que o emprego de multicaminhos com o protocolo MP-OLSR apresenta uma TPP cerca de quatro vezes menor que o protocolo de caminho único OLSR no cenário estático e aproximadamente três vezes menor no cenário com baixa mobilidade. Contudo, foi constatado que o MP-OLSR apresenta um *jitter* mais elevado. Estes resultados demonstram a aplicabilidade de protocolos de roteamento multicaminhos para o encaminhamento de dados em situações de congestionamento nas VANETs.

Referências

- Bisht, A., Kumar, B., and Mishra, S. (2012). Efficiency evaluation of routing protocols for vehicular ad-hoc networks using city scenario. In *International Conference on Computer Communication and Informatics*, pages 1–7.

- Clausen, T. and Jacquet, P. (2003). Optimized link state routing protocol (olsr).
- Faezipour, M., Nourani, M., Saeed, A., and Addepalli, S. (2012). Progress and challenges in intelligent vehicle area networks. *Commun. ACM*, 55(2):90–100.
- Iperf (2013). <http://iperf.sourceforge.net>. Último Acesso: Fevereiro de 2013.
- Johnson, D. and Hancke, G. (2009). Comparison of two routing metrics in olsr on a grid based mesh network. *Ad Hoc Networks*, 7(2):374–387.
- Park, J.-S., Lee, U., and Gerla, M. (2010). Vehicular communications: emergency video streams and network coding. *J. Internet Services and Applications*, 1(1):57–68.
- Radu, D., Avram, C., Astilean, A., Parrein, B., and Yi, J. (2012). Acoustic noise pollution monitoring in an urban environment using a vanet network. In *IEEE International Conference on Automation Quality and Testing Robotics*, pages 244 –248.
- Sofra, N., Gkelias, A., and Leung, K. (2011). Route construction for long lifetime in vanets. *IEEE Transactions on Vehicular Technology*, 60(7):3450 – 3461.
- Spaho, E., Ikeda, M., Barolli, L., Xhafa, F., Younas, M., and Takizawa, M. (2012). Performance of olsr and dsdv protocols in a vanet scenario: Evaluation using cavernet and ns3. In *Seventh International Conference on Broadband, Wireless Computing, Communication and Applications.*, pages 108 –113.
- Toutouh, J., Garcia-Nieto, J., and Alba, E. (2012). Intelligent olsr routing protocol optimization for vanets. *IEEE Transactions on Vehicular Technology*, 61(4):1884 –1894.
- Wireshark (2013). <http://www.wireshark.org>. Último Acesso: Fevereiro de 2013.
- Yi, J., Adnane, A., David, S., and Parrein, B. (2011). Multipath optimized link state routing for mobile ad hoc networks. *Ad Hoc Networks*, 9(1):28 – 47.